

Université IBM i 2017

17 et 18 mai – IBM Client Center de Bois-Colombes

S36 - Analyser les performances SQL avec Visual Explain d'ACS

Jeudi 18 mai – 13h30 -15h00

Eric Saglier – Six-Axe
eric.saglier@six-axe.fr



Plan de la présentation

- Facteurs influençant l'optimisation des requêtes SQL.
 - Méthode d'accès.
 - Statistiques
 - Les différentes stratégies d'indexation.
 - Radix
 - EVI
 - Index avec clés dérivées
 - Index avec sélections

- L'outils d'aide Visual Explain
 - A quoi sert-ce composant ?
 - Comment fonctionne-t-il ?
 - Comment au travers de l'usage de celui-ci réussir à optimiser au mieux les requêtes SQL ?



SQL

- Langage d'accès aux bases de données relationnelles
- Langage naturel Facile
 - à écrire
 - à lire
 - à comprendre
- Fonctions relationnelles
- Langage assertionnel
 - **QUOI** et non **COMMENT**

SQL

- Langage assertionnel : **QUOI ?**
 - Donnez moi la liste des salariés dont l'âge est supérieure à la moyenne des âges du service dans lequel le salarié travaille.

Epno	Firstname	Midinit	Lastname	Workdept	Phone	Hiredate	Job	Edlevel
000010	Christine	I	Haas	A00	3978	1965-01-01	PRES	18
000060	Irving	F	Stern	D11	6423	1973-09-14	MANAGER	16
000110	Vincenzo	G	Lucchessi	A00	3490	1958-05-16	SALESREP	19
000130	Delores	M	Quintana	C01	4578	1971-07-28	ANALYST	16
000150	Bruce		Adamson	D11	4510	1972-02-12	DESIGNER	16
000180	Marilyn	S	Scoutten	D11	1682	1973-07-07	DESIGNER	17
000200	David		Brown	D11	4501	1966-03-03	DESIGNER	16

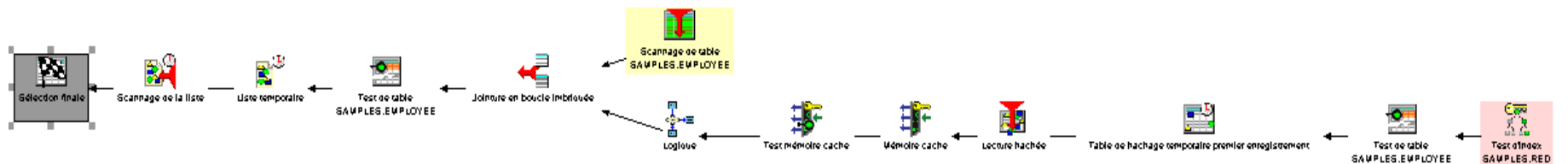
SQL

■ Langage assertionnel : **Comment ?**

– SQL

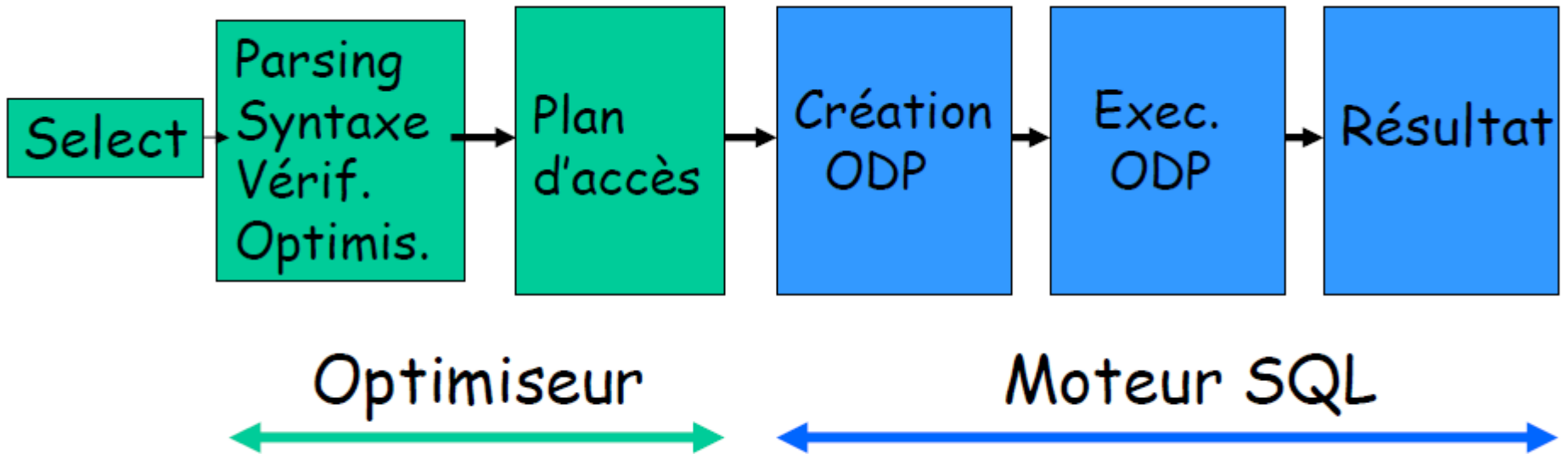
```
SELECT * FROM samples.employee e WHERE YEAR(CURDATE() -  
BIRTHDATE) > (SELECT AVG(YEAR(CURDATE() - BIRTHDATE))  
FROM samples.employee WHERE e.workdept = workdept)
```

– Stratégie mise en œuvre par l'optimiseur.



Requête SQL

- La vie d'une requête SQL

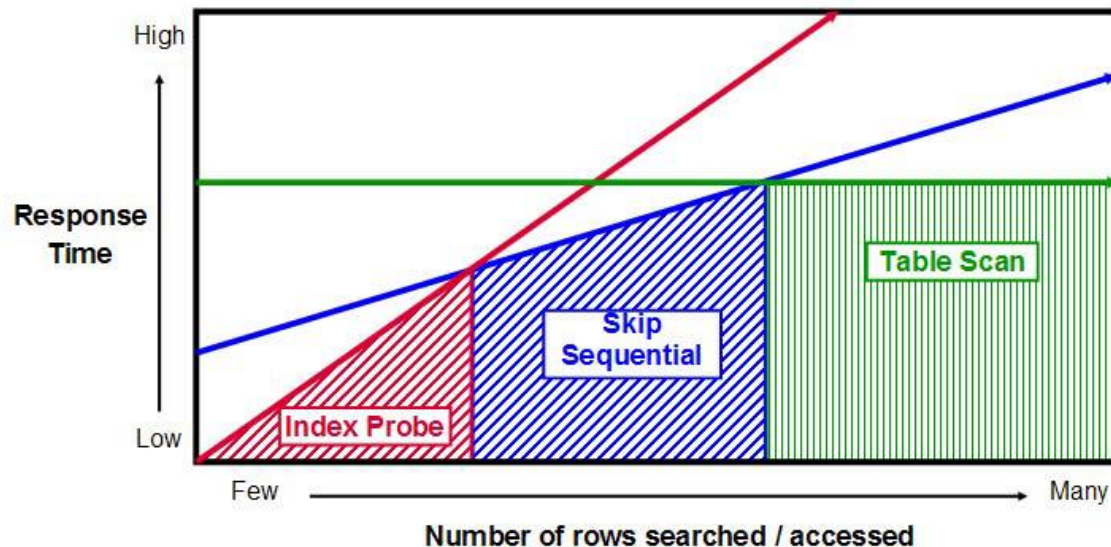


DB2 for i et l'usage des index

- DB2 for i s'appuie sur les index afin de fournir les méthodes et les statistiques permettant de fournir au plus vite les résultats.
- Au cours du processus d'optimisation, DB2 for i examine tous les index pertinents afin de réaliser la requête.
- Soit la requête suivante
 - SELECT COL1 where COL2 = 'XXX' order by COL3
 - Un index comportant COL1 est sans intérêt pour l'optimiseur (pas de statistiques ou de méthode d'exécution).
 - Un index comportant COL2 et/ou COL3 en clé principale pourra être envisagé afin d'activer un usage de type « Probe »

Méthode d'accès aux données

- En fonction de la **sélectivité**, DB2 pourra mettre en œuvre :
 - Un index « Probe »
 - Un index « Scan »
 - Un Table « Scan »



Les différentes méthodes d'accès aux données

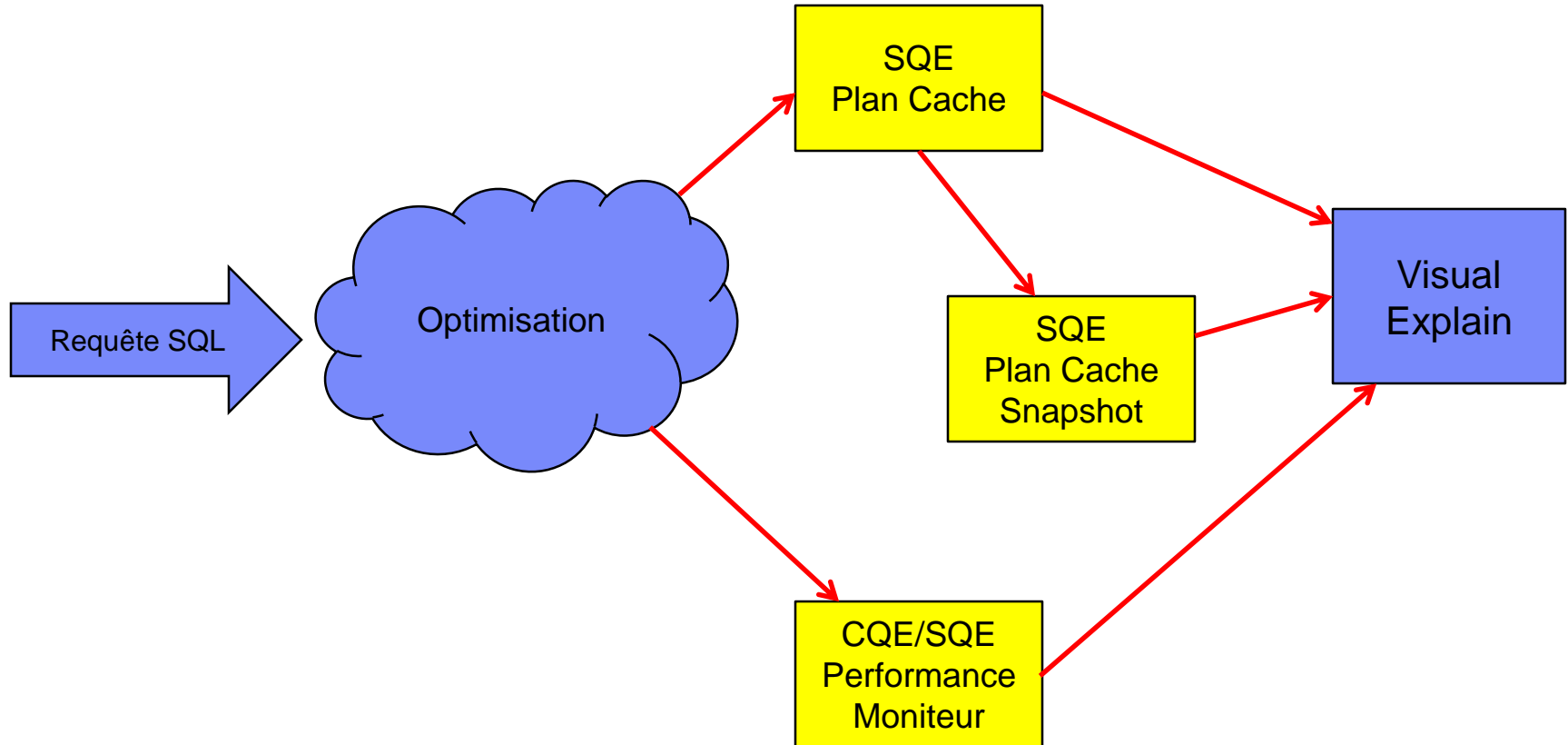
- la méthode d'accès la plus rapide pour une table variera en fonction de la sélectivité et du nombre d'occurrences par valeur distincte.
- Sur la base des 101 départements, la répartition dans la table des clients ne sera pas de 1/101 par département.
- C'est le rôle des statistiques que de fournir ces informations.

Rôles des statistiques

- Plus la sélectivité est importante, plus la présence d'un index est primordiale.
- L'objectif de l'optimiseur est de choisir une mise en œuvre éliminant les lignes qui ne sont pas intéressantes ou pas nécessaires. Pour ce faire :
 - L'optimiseur DB2 utilise les informations statistiques des index afin de « comprendre les données » stockées dans les tables sous-jacentes.
 - Cette compréhension des données permet à l'optimiseur trouver la méthode d'accès aux données la plus efficace.
 - Les index Radix et les EVI contiennent des informations sur le nombre de valeurs distinctes dans une colonne, ainsi que la distribution des dites valeurs.
 - La mise à jour des statistiques est définie via la valeur système QDBFSTCCOL (*ALL, *SYSTEM, *USER, *NONE)

Optimisation DB2 et outils d'analyse

- Outils d'analyse



Analyse d'une requête SQL inefficace

- Il faut rechercher et éliminer :
 - Les analyses complètes de table (Full table « Scan »).
 - Les analyses complètes d'index (Full index « Scan »).
 - Les constructions d'index temporaires.

les différentes stratégies d'indexation.


- Deux types d'index sont supportés par DB2 for i
 - Les index de type **Radix**
 - Les index de type **EVI** (Encoded Vector Index)
- Chaque type présente des spécificités et des avantages.
- Ces deux types de technologies se complètent parfaitement.
- Ces deux types d'index sont utilisés aussi bien pour la mise à jour des statistiques que pour l'exécution des requêtes.
- Les index sont utilisés de manières « Scanner » ou/et « Prober »

Lecture des index.

- Lecture « Scanned »
 - Lecture séquentielle de l'index.
 - Balayage de tous les postes d'index à la recherche des postes correspondants à la sélection.

- Lecture « Probed »
 - fourniture d'une clé
 - Parcours (rapide) dans l'arbre binaire
 - Accès au RRN (relative Record Number)

Scan/Probe

- Existence d'un index sur les colonnes (ITEM_NO, COLOR, SIZE)
 - ... WHERE COLOR = 'NOIR' and ITEM_NO = 003
 - ... WHERE SIZE = 'MEDIUM'
 - ... WHERE ITEM_NO =001 AND SIZE = 'LARGE'
- **Accès Probe** (Positionnement sur clé)
 - Clé(s) principale(s) + n clés contiguës
 - 1 = Valeur de ITEM_NO
 - 1 + 2 = Valeur de ITEM_NO, COLOR
 - 1 + 2 + 3 Valeur de ITEM_NO, COLOR,SIZE
- **Accès Scan** (Test d'index) 
 - N'importe quel ordre de clés
 - 2
 - 3
 - 2 + 3

ITEM_NO	COLOR	SIZE
001	BLEU	LARGE
002	ROUGE	SMALL
003	NOIR	SMALL
004	VERT	MEDIUM

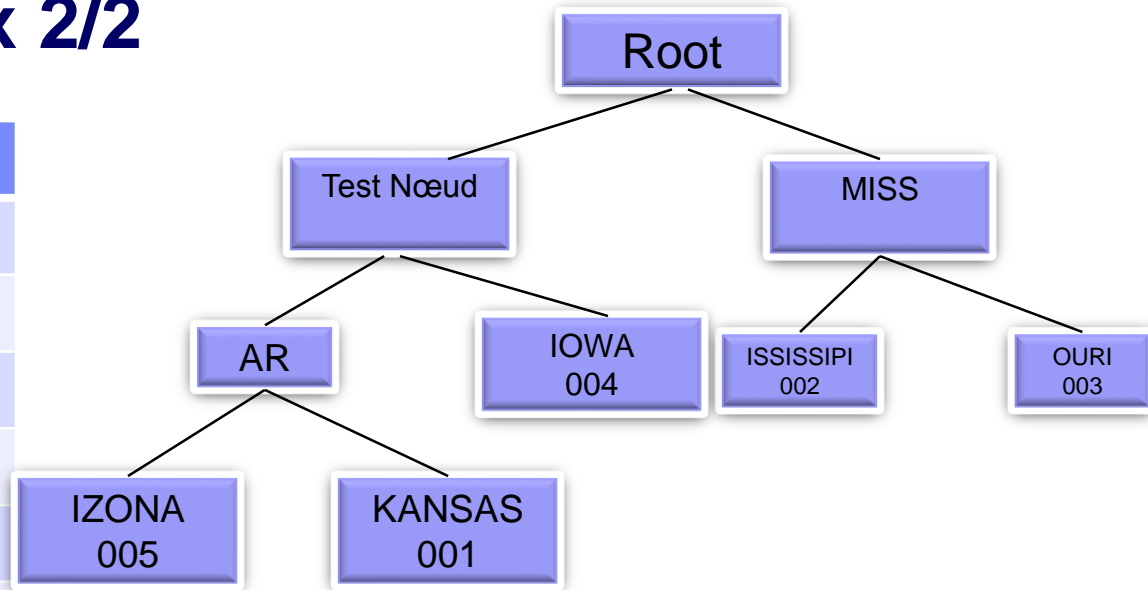
Index de type Radix 1/2

- Les index de type Radix utilisent une structure de type arbre binaire.
- Les valeurs de clés sont compressées logiquement
 - Les données communes sont stockées une seule fois,
 - Les données uniques sont stockées seules.
- Existence d'un algorithme propre de recherche au sein d'un arbre binaire
- Maintenance
 - L'arbre est automatiquement mis à jour en fonction des opérations réalisées au travers du DML (Data Manipulation Language), afin de maintenir une structure de données efficace.



Index de type Radix 2/2

Code	State
001	ARKSANSAS
002	MISSISSIPI
003	MISSOURI
004	IOWA
005	ARIZONA



■ Avantages

- Accès très rapide à une seule valeur de clé
- Accès rapide à un petit ensemble de clés (peu de cardinalité)
- Tri demandé

■ Désavantages

- Lignes récupérées dans l'ordre des valeurs de clés (pas de l'ordre physique) équivalent à des I/O random
- Aucune manière de prédire quelles seront les pages suivantes pour des valeurs de clés importantes

Index de type EVI

- Index conçu afin de délivrer des accès rapides dans les environnements de type « report analytique ».
 - Technologie issue de travaux de recherche IBM.
 - Permet de produire dynamiquement des listes de RRN.
 - Accès très rapide aux statistiques afin de faciliter les décisions de l'Optimiseur.
 - Ce n'est pas une structure de type « arbre »
 - Ne peut être créé que par l'intermédiaire d'une interface SQL, d'IBM Navigator for i ou de System i Navigator.

```
CREATE ENCODED VECTOR INDEX Mabib.Ixnom ON  
Mabib.Tablenom (Key1, Key2, ...)  
INCLUDE (SUM(Col2)) (Nouveauté V7R1)
```

Index de type EVI

Symbol Table				
Key Value	Code	Count	Include Sum()	Include Sum()
Arizona	1	5000	1500	2005
Arkansas	2	7300	3200	450
...				
Wisconsin	49	340	575	1200
Wyoming	50	2760	210	0

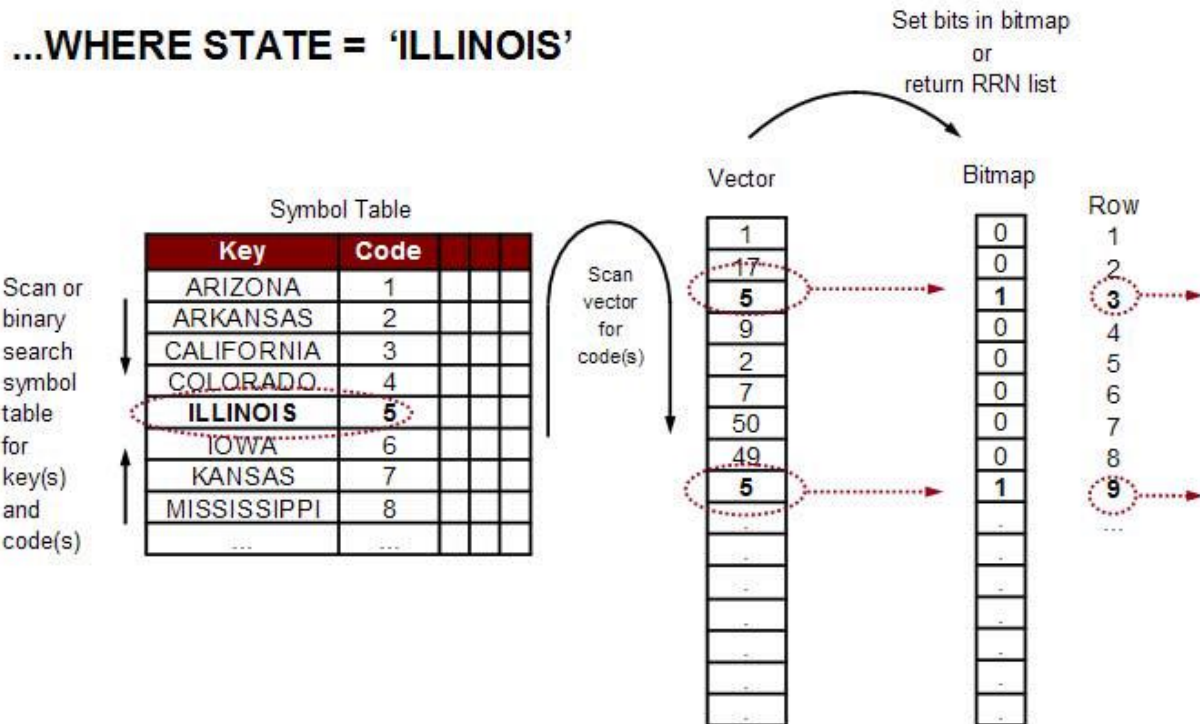
optional

Vector	RRN
1	1
17	2
5	3
9	4
2	5
7	6
50	7
49	8
5	9
...	...

- La table des symboles contient des informations pour chaque valeur distincte de clé
 - Chaque clé se voit affecter un code unique
 - Le code est sur 1,2 ou 4 bits (en fonction du nombre de valeurs distinctes)
- Nouveauté en V7R1 : possibilité de spécifier AVG, COUNT, COUNT_BIG, SUM, STDDEV, STDDEV_SAMP, VARIANCE, VARIANCE_SAMP
- En lieu et place d'un tableau de bits pour chaque valeur, utilisation d'un tableau de vecteurs

Utilisation d'un EVI

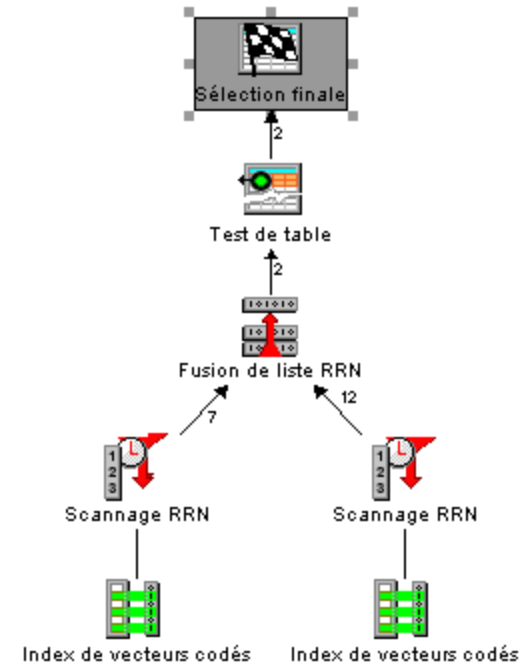
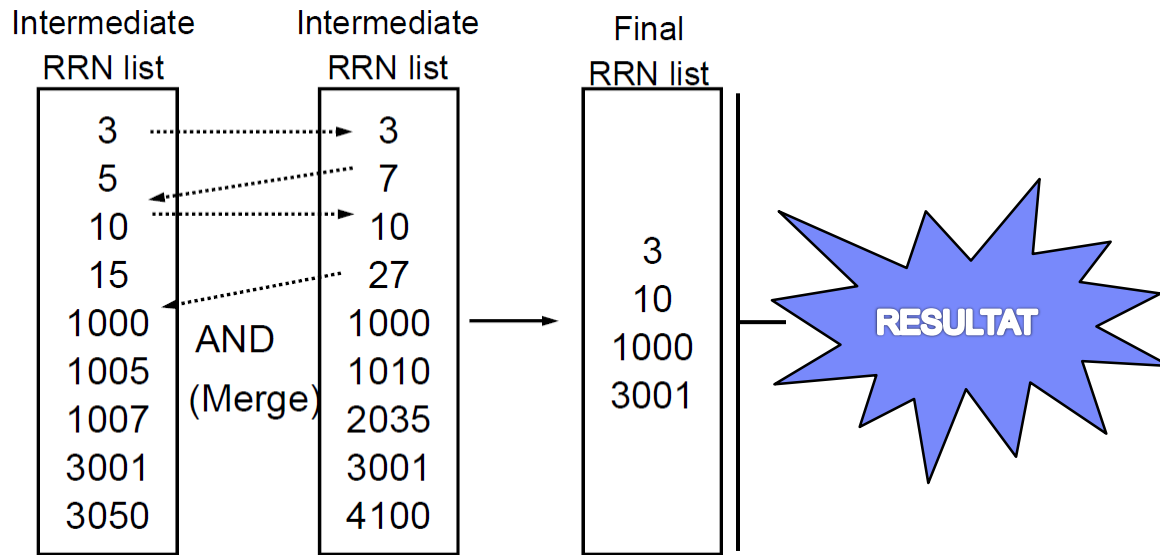
- Si l'optimiseur utilise un EVI
 - A partir de la table des vecteurs et pour les valeurs répondant aux critères de sélection. Le système bâtit une liste de RRN en positionnant un bit à *ON ou *OFF



Utilisation d'un EVI AND

- EVI et utilisation d'un AND

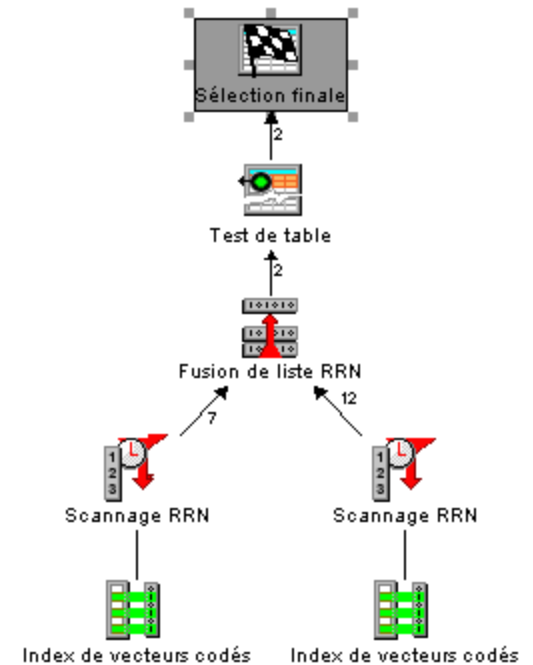
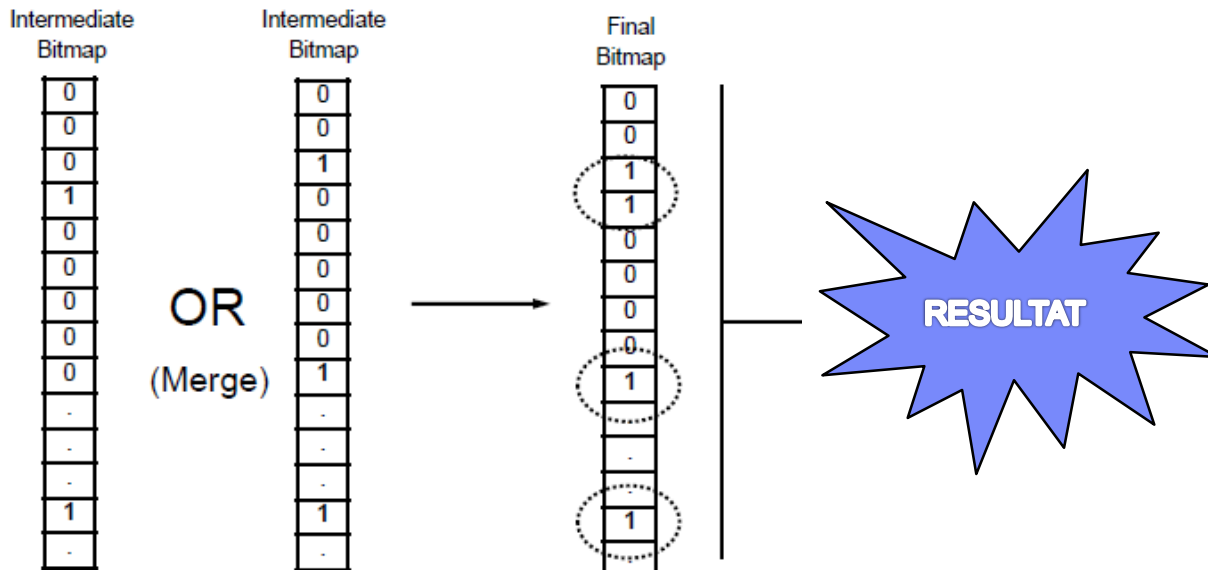
Select * from samples.employee where WORKDEPT in ('B01' , 'D11' , 'C01') **AND** JOB = 'MANAGER'



Utilisation d'un EVI OR

- EVI et utilisation d'un OR

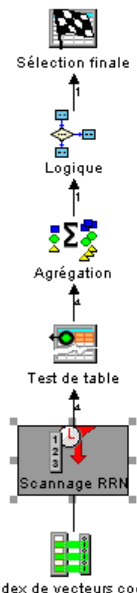
Select * from SAMPLES.EMPLOYEE where WORKDEPT = 'B01' **OR**
 JOB = 'MANAGER'



Utilisation d'un EVI et fonctions de colonnes

- EVI et les fonctions de colonnes (V7)
 - CREATE ENCODED VECTOR INDEX SAMPLES.EVI_SUM ON SAMPLES.EMPLOYEE (WORKDEPT)
 - INCLUDE (COUNT (*) , SUM (SALARY) , SUM (BONUS))
 - SELECT COUNT(*) FROM SAMPLES.EMPLOYEE WHERE WORKDEPT = 'C01'
 - SELECT COUNT(distinct WORKDEPT) FROM SAMPLES.EMPLOYEE
 - SELECT WORKDEPT, SUM(BONUS), SUM(SALARY) FROM SAMPLES.EMPLOYEE GROUP BY WORKDEPT

WORKDEPT	Count	Sum(Salary)	Sum(bonus)
A00	5	204250,00	4100,00
B01	1	800,00	41850,00
...
E21	6	2900,00	144520,00



Index avec clés dérivées

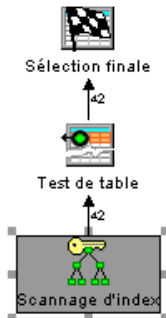
- Ce type d'index est apparu en V6R1
 - Permet de faire apparaître des expressions dans les colonnes clés.
 - Utilisé dans le cas de recherche en fonction de la casse.
 - Données extraites à partir de chaîne de caractères (SUBSTR, YEAR, MONTH,...).
 - Colonnes de groupage dérivées (GROUP BY YEAR(), ...)
 - Résultats de calculs ((PU + HT) , (QUANTITE * PU), ..)

 - CREATE INDEX SAMPLES.LASTNAME_UPPER ON SAMPLES.EMPLOYEE (UPPER(LASTNAME) as ULASTNAME ASC)
 - CREATE ENCODED VECTOR INDEX SAMPLES.YEARQTR ON SAMPLES.EMPLOYEE (YEAR(BIRTHDATE) as BIRTHYEAR ASC, QUARTER(BIRTHDATE) as BIRTHQ)

- A partir de la V7R1, ce type d'index est utilisable par l'optimiseur.

Index avec clés dérivées

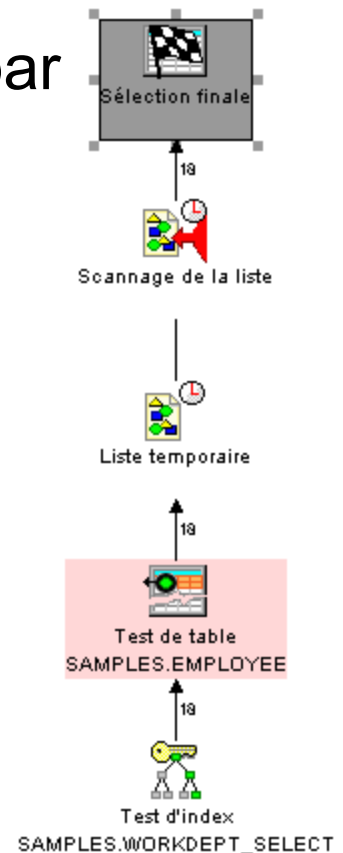
- CREATE INDEX SAMPLES.TOTALBONUS ON SAMPLES.EMPLOYEE ((SALARY + BONUS) * 13 as TOTALANNUEL)
- select firstnme from samples.employee order by ((salary + bonus) * 13)



Nom de table, nom de table de b...	
Nom de l'index utilisé	TOTALBONUS
Bibliothèque de l'index utilisé	SAMPLES
Membre de l'index utilisé	TOTALBONUS
Nom long de l'index utilisé	TOTALBONUS
Nom long de la bibliothèque de l'...	SAMPLES
Nom de la table interrogée	EMPLOYEE
Bibliothèque de la table interrogée	SAMPLES
Membre de la table interrogée	EMPLOYEE
Nom long de la table interrogée	EMPLOYEE
Nom long de la bibliothèque de l'...	SAMPLES
Accès par index seulement	Non
Informations durée estimée (he...	
Temps de traitement (ms)	.013
Temps cumulé (ms)	.013
Infos index supplémentaires	
Nombre d'entrées d'index	42
Taille de zone clé	8
Taille de page logique de l'index	65,536

Index avec sélection

- Ce type d'index apparait en V6R1 et permet de remplacer les fichiers logiques.
- A partir de la V7R1, ce type d'index est utilisable par l'optimiseur.
 - CREATE INDEX SAMPLES.WORKDEPT_SELECT ON SAMPLES.EMPLOYEE (WORKDEPT ASC)
WHERE WORKDEPT in ('D11', 'E11')
 - SELECT * FROM samples.employee WHERE WORKDEPT IN ('D11', 'E11') ORDER BY workdept



Choisir le bon type d'index

- Le choix du type d'index est fonction de la cardinalité des occurrences pour la valeur des colonnes de zones clés
 - **Haute cardinalité** : Nombre élevé de valeurs distinctes
 - **Faible cardinalité** : Nombre faible de valeurs distinctes.

- En général
 - Un index de type « **Radix** » est le meilleur choix à la condition que **la cardinalité soit élevée** et que le nombre **d'occurrences à traiter soit faible**.
 - Un index de type « **EVI** » est le meilleur choix pour accéder à un lot d'occurrences avec de **faibles cardinalités**.



Méthodologie d'indexation

- Suivant les paramètres de la requête, l'optimiseur mettra en œuvre les aspects suivants
 - Si présence de Sélection
 - Table « Scan »
 - Table « Probe »
 - Index « Scan »
 - Index « Probe »
 - Si présence de Jonction
 - Nested loop join with index (Jointure en boucle avec index)
 - Nested loop join with hashing (Jointure en boucle avec hachage)
 - Nested loop join with sort list (Jointure en boucle avec liste de tri)
 - Si présence de Group by
 - Grouping avec Index (Agrégation)
 - Grouping avec hachage
 - Si présence de Order by
 - Index
 - Tri

Stratégie d'indexation

- Deux approches pour créer les index :
 - Une approche proactive
 - Une approche réactive.

- L'approche proactive consiste à bâtir les index sur la base du modèle de données (MCD)
 - Mise en œuvre contraintes de clés Primaires, Uniques et Etrangères.
 - Communément, les colonnes utilisées dans les sélections ainsi que les colonnes dépendantes, en relation, en corrélation.
 - Communément, les colonnes utilisées dans les jonctions.
 - Communément, les colonnes utilisées dans le groupage

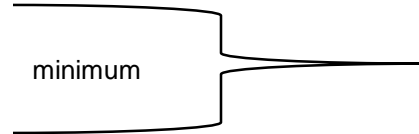
Stratégie d'indexation

- Méthode réactive. L'objectif d'une stratégie d'indexation est de donner à l'optimiseur de requêtes les deux éléments suivants :
 - Information sur les données contenues dans les table telles que :
 - le nombre de valeurs distinctes,
 - la répartition des valeurs de données,
 - le nombre moyen de valeurs en double.
 - Choix concernant les méthodes à utiliser pour traiter la requête.
Le choix dépend du résultat souhaité (Batch, Interactif, Optimize for all rows,...)

Stratégie d'indexation basique

■ Index de type « RADIX »

- Sélection de colonnes
- Jonction de colonnes
- Sélection de colonnes + colonnes de jonction
- Sélection de colonnes + colonnes de groupage
- Sélection de colonnes + colonnes de tri



■ Index de type « EVI »

- Sélection local avec AND / OR
- Jonction de colonnes
- Fonctions (Distinct, Count, SUM,..)

- Privilégier la sélectivité, mettre les colonnes en égalité en premier

Stratégie d'indexation

- Soit la requête suivante :

```
Select b.col1, b.col2, a.col1
```

```
From table1 a inner join table2 b on (a.join_col = b.join_col) (Jonction)
```

```
Where      b.col3 = ' quelques valeurs'          (Selection)
```

```
          and b.col2 = 9999
```

```
Group by   c.col1, b.col2, a.col1
```

```
Order by   b.col1
```

- Avec une requête de ce type les règles sont :
 - 2 index Radix sur les colonnes de **jonction** (a.join_col et b.join_col)
 - Sélection
 - 2 index Radix sur les colonnes de sélection (b.col3 et b.col2) **dès lors quelles sont utilisées dans d'autres requêtes comme critères de sélection.**
 - **Ou** bâtir deux index EVI sur les colonnes b.col3 et b.col2 dès lors que les colonnes de sélection ne sont pas uniques et que le nombre d'occurrences pour chacune est relativement faible.

Index de type Radix

- Dans un index l'ordre des colonnes est important
 - **Classer les colonnes d'égalité en premier** (permet d'activer la sélectivité)
 - Si tous les prédicats utilisés **sont en égalité**
 - **Colonnes de sélection + colonnes de jonction**
 - **Colonnes de sélection + colonnes de groupage**
 - **Colonnes de sélection + colonnes de tri**
 - **Mettre la colonne à sélectivité la plus élevée en premier.**

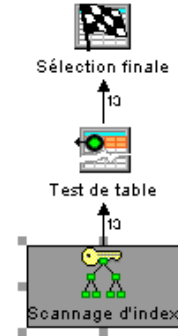
Select col1, col2, col3 from table where col4 = 2014 and col5 = 3 and col6 = 'T' order by col1, col2

L'index à créer comportera les colonnes de sélection (col4, col5, col6) suivies des colonnes de tri col1 et col2

L'ordre des colonnes de sélection devra être choisi en fonction de la sélectivité

Index parfait de type Radix/Index Only Access IOA

- Index Parfait : inclure dans l'index les colonnes souhaitées
Select nom, srv from employees order by nom
 - Si utilisation d'un Index sur nom uniquement.



- Création d'un index IOA « Only Access »

- Select nom, srv from ESALIB.EMPLOYES order by srv
CREATE INDEX ESALIB.INDEX_PARFAIT ON ESALIB.EMPLOYES (SRV ASC, NOM ASC)

- Récupération de l'ensemble des informations au sein de l'index => plus besoin de test de table

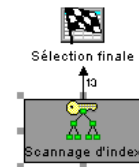


Illustration - Indexation de base

- SELECT partkey, shipmode FROM item_fact WHERE **orderkey**=1 ORDER BY partkey

- Index sur orderkey

- CREATE INDEX DB2IPFR/INDEX_BASE DB2IPFR/ITEM_FACT (**ORDERKEY** ASC)

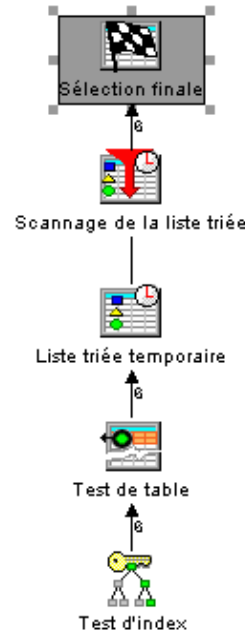
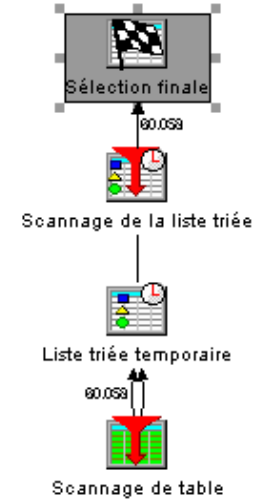
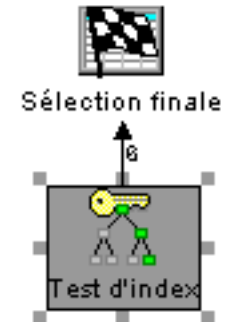


Illustration - Mise en œuvre index parfait Only Access

- Mieux encore, mettre en œuvre des Index parfaits de type « only access » en conséquence il n'y a plus d'accès à la table :
 - L'intégralité des colonnes de la requête fait partie des colonnes clés soit la requête :
 - `SELECT partkey, shipmode FROM item_fact WHERE orderkey=1 ORDER BY partkey`
 - L'index suivant contient toutes les colonnes :
 - `CREATE INDEX DB2IPFR.INDEX_IOA ON DB2IPFR.ITEM_FACT (ORDERKEY ASC, PARTKEY ASC, SHIPMODE ASC)`

- Durée estimée
 - Sans index : 3 950 ms
 - Index de base : 25 ms
 - Index parfait : 19 ms
 - Index parfait Only access : 1 ms



Optimisation DB2

- Lors de l'exécution d'une requête, l'optimiseur réalise les opérations suivantes :
 - Rassemblement des informations sur les métadonnées et les statistiques afin d'établir le coût d'exécution
 - Statistiques de sélectivité
 - Détermination des index disponibles pour chiffrer les coûts
 - Tri des index en fonction de leur pertinence
 - Détermination des attributs d'exécution pouvant affecter le coût
 - Génération d'un coût par défaut
 - Permet d'établir un plan d'accès associé avec le plan par défaut
 - Et ce pour chaque index
 - Recueil des renseignements nécessaires spécifiques à l'index
 - Construction d'un plan d'accès basé sur cet index
 - Chiffrage du coût de l'utilisation de cet index avec le plan d'accès
 - Comparaison du coût obtenu avec le coût le plus bas trouvé précédemment
 - Répétition de cette opération afin de trouver le coût le plus bas

Outil de conseil à la construction des index

- L'optimiseur va enregistrer les index conseillés dans un fichier. Le fichier **SYSIXADV de QSYS2**
- Ces conseils sont établis aussi bien pour le moteur SQE que pour le moteur CQE
- SQE
 - Conseils évolués
 - Index de type « Radix » et « EVI »
 - Basés sur la requête complète
 - Plusieurs index proposés pour la même requête
 - Quelques limitations
- CQE
 - Conseils basiques (Table scan et sélection uniquement)
 - Index « Radix » uniquement
 - Index temporaires
 - Visual Explain est susceptible de proposer de meilleurs index

Outil de conseil à la construction des index

- Fournit des informations sur
 - Sélections locales
 - Jonctions
 - Group by
 - Order by

- Propose des index sur
 - Sélection en égalité , suivi par inégalité (« Probe »)
 - Conseil limité avec des OR sur des colonnes différentes
 - Pas d'index dérivé
 - Pas d'EVI avec des fonctions d'agrégations

- Les conseils sont nombreux mais pas obligatoires...

Visual Explain

- Visual Explain permet d'afficher une représentation graphique du plan de la requête. Il permet de :
 - Visualiser les objets ainsi que les structures de données
 - Visualiser les méthodes et stratégies mises en œuvre
 - Présenter les informations associées à l'environnement d'exécution
 - Fournir des conseils sur les index à construire
 - Fournir des conseils sur les statistiques à collecter
- Cet outil est accessible au travers de l'interface System i Navigator et bien sur au sein d'IBM i Access Client Solution
- Il permet d'analyser
 - Le Plan Cache SQE
 - Les Snapshots du Plan Cache
 - Les moniteurs de performances base de données
 - Le détail d'un job

Objectifs de Visual Explain

- Visual Explain fournit des informations qui peuvent vous aider à déterminer si vous devez:
 - Réécrire ou modifier l'instruction SQL
 - Modifier les attributs de requête ou les paramètres de l'environnement
 - Créer de nouveaux index

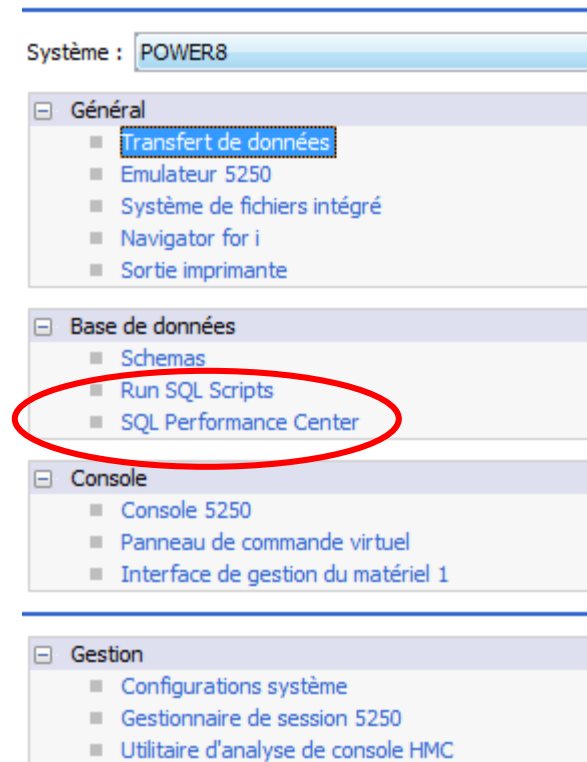
- Vous pouvez également utiliser Visual Explain afin de :
 - Voir les statistiques utilisées au moment de l'optimisation
 - Déterminez si un index a été utilisé pour accéder à une table
 - Déterminer les index à construire susceptibles d'améliorer les performance
 - Voir les effets de l'exécution des diverses options prises en comparant les versions avant et après du graphe de requête
 - Obtenir des informations sur chaque opération dans le graphique de la requête, y compris le coût.

Objectifs de Visual Explain

- Visual Explain fournit une représentation graphique de l'implémentation de l'optimiseur mis en œuvre dans le cadre de l'exécution d'une requête SQL.
- L'exécution de la requête est divisée en composants individuels présentés sous une forme iconique, chaque icône représente une étape mise en œuvre.
- Visual Explain affiche les informations sur les objets de base de données considérés et choisis par l'optimiseur de requête.
- Visual Explain permet de montrer le détail de l'environnement de travail ainsi que les niveaux de parallélisme de base de données utilisés pour traiter la requête.
- Visual Explain affiche le plan d'accès sous la forme de diagramme, en vous permettant de zoomer sur n'importe quelle partie du diagramme.

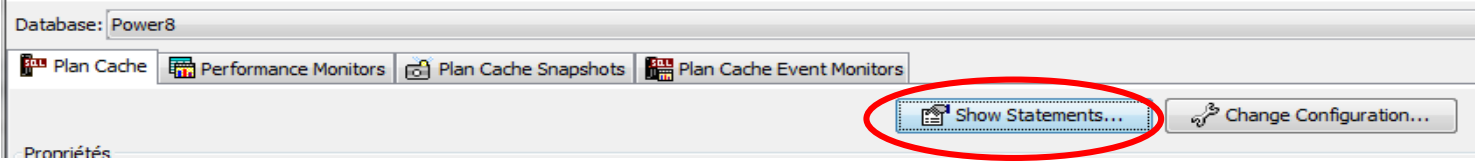
Visual Explain et i ACS

- L'accès à Visual Explain au sein d'i ACS est possible à partir de Base de données en sélectionnant soit :
 - Run SQL Scripts
 - SQL Performance Center

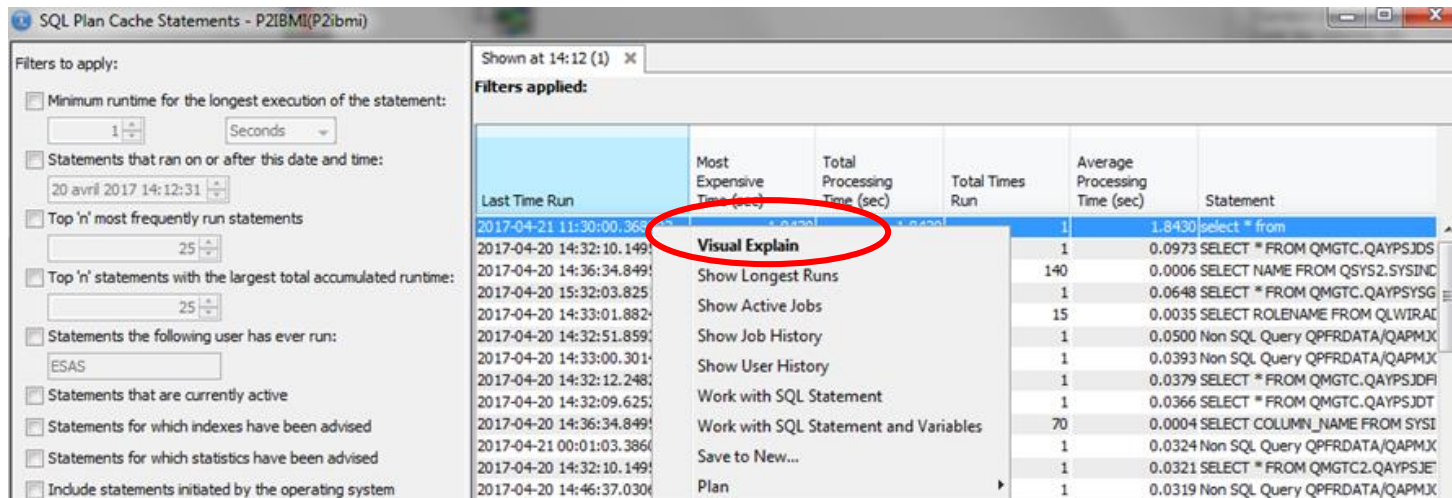


Visual Explain - SQL Performance center- Plan Cache

- Après avoir sélectionné SQL Performance Center et avoir sélectionné la vue Show Statements



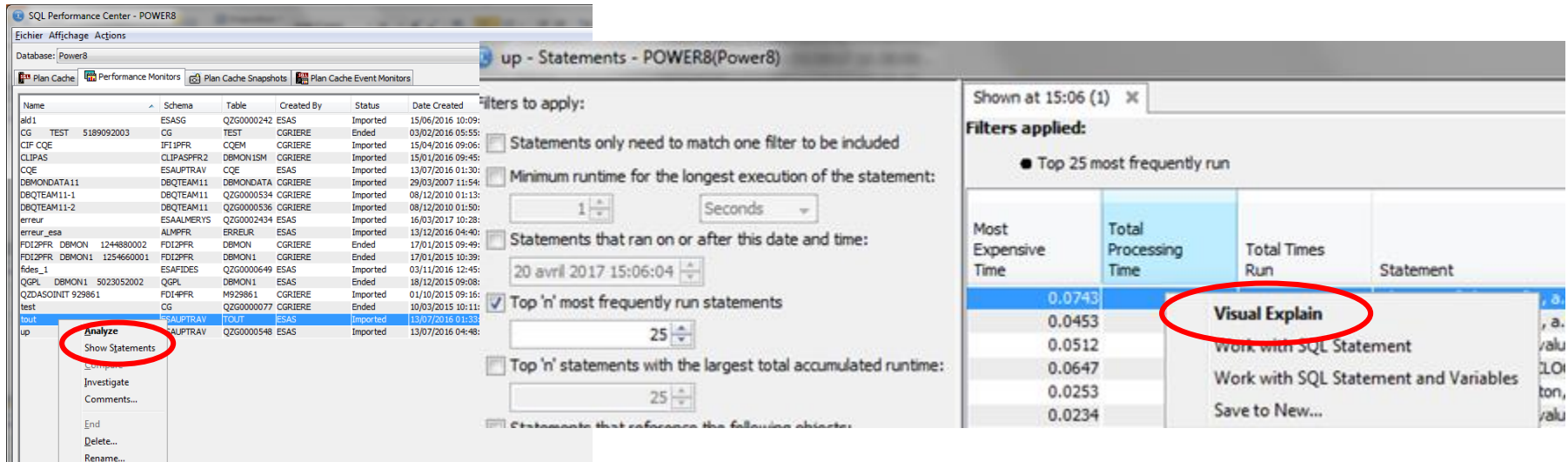
- Sélectionner une instruction SQL, utilisez le click bouton droit et choisir Visual Explain



Visual Explain - SQL Performance center

- Après avoir sélectionné un moniteur au sein de :
 - Performance Center,
 - Plan Cache Snapshots
 - Plan Cache Event moniteur

- Sélectionner une instruction SQL, utilisez le click bouton droit, puis choisir Visual Explain

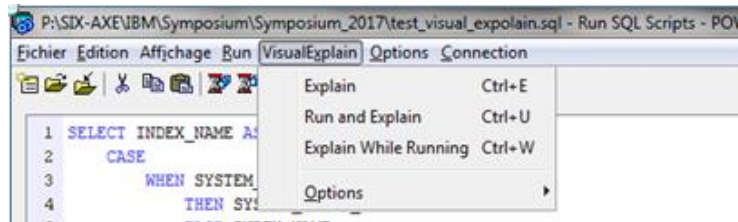


The screenshot shows the SQL Performance Center interface. On the left, a table lists various statements with columns for Name, Schema, Table, Created By, Status, and Date Created. A context menu is open over a statement, with the 'Analyze' option selected. On the right, the 'Visual Explain' dialog is shown, displaying filters applied (Top 25 most frequently run) and a table of statement performance metrics.

Most Expensive Time	Total Processing Time	Total Times Run	Statement
0.0743	Visual Explain		
0.0453			
0.0512			Work with SQL Statement
0.0647			Work with SQL Statement and Variables
0.0253			Save to New...
0.0234			

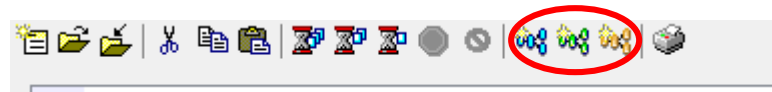
Lancement Visual Explain via Script SQL

- A partir de Script SQL il est possible de lancer Visual Explain et cela de trois manières :
 - Explain : En explicitation uniquement (sans exécution de la requête).
 - Run and Explain : En exécution de la requête, puis explicitation de la requête.
 - Explain while Running : En explicitation encours d'exécution.
- Ce lancement est possible :
 - Soit au travers de la barre de tâche



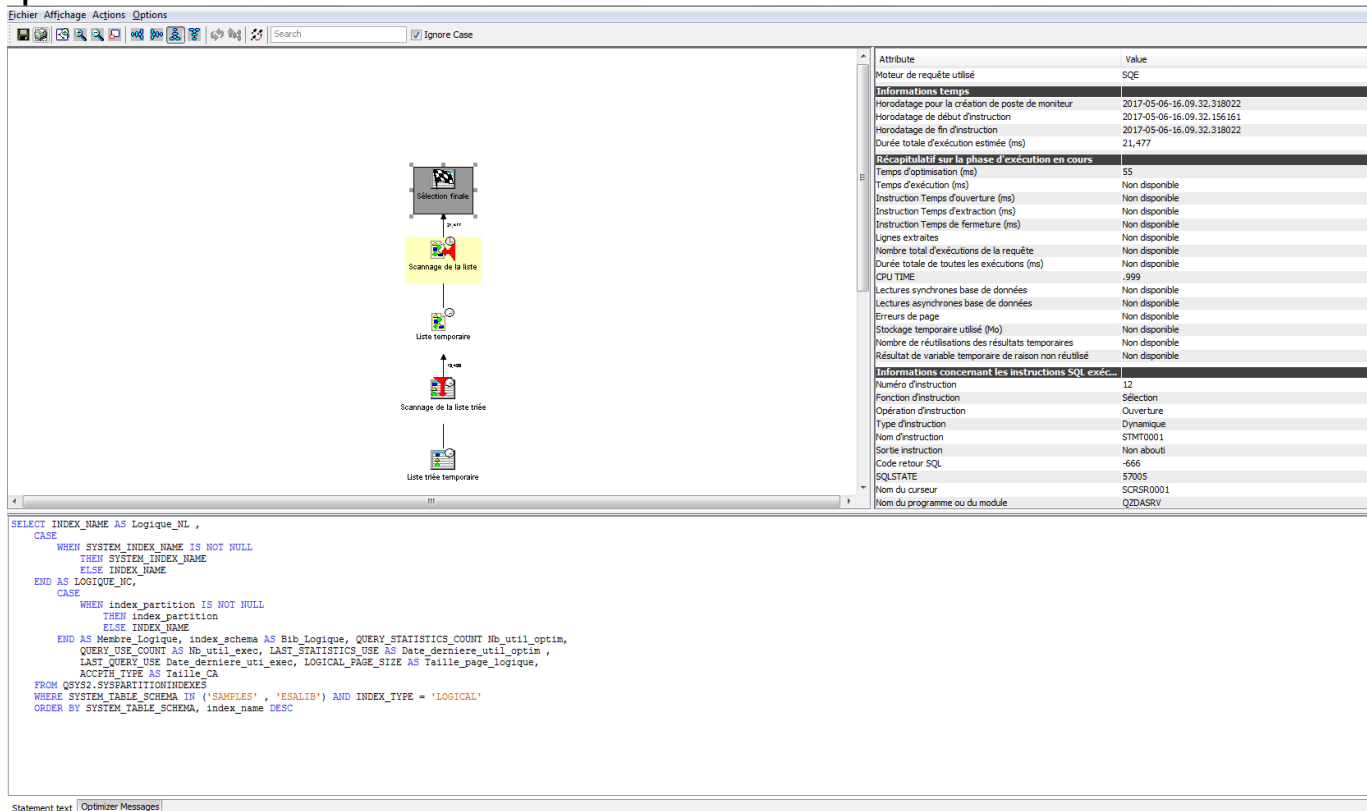
- soit par le biais des icones

- Bleu : Explain
- Vert : Run and Explain
- Orange : Explain while Running



Présentation générale de Visual Explain

- Affichage d'une fenêtre présentant trois cadres
 - Cadre en haut à gauche : Visualisation de la stratégie mise en œuvre
 - Cadre en haut à droite : Explicitation des étapes.
 - Cadre du bas suivant l'onglet sélectionné : Requête exécutée ou message de l'optimiseur.



The screenshot displays the Visual Explain tool interface. The top-left pane shows a query execution plan with steps: Sélection finale, Scannage de la liste, Liste temporaire, Scannage de la liste triée, and Liste triée temporaire. The top-right pane shows a table of execution statistics. The bottom pane shows the SQL query text.

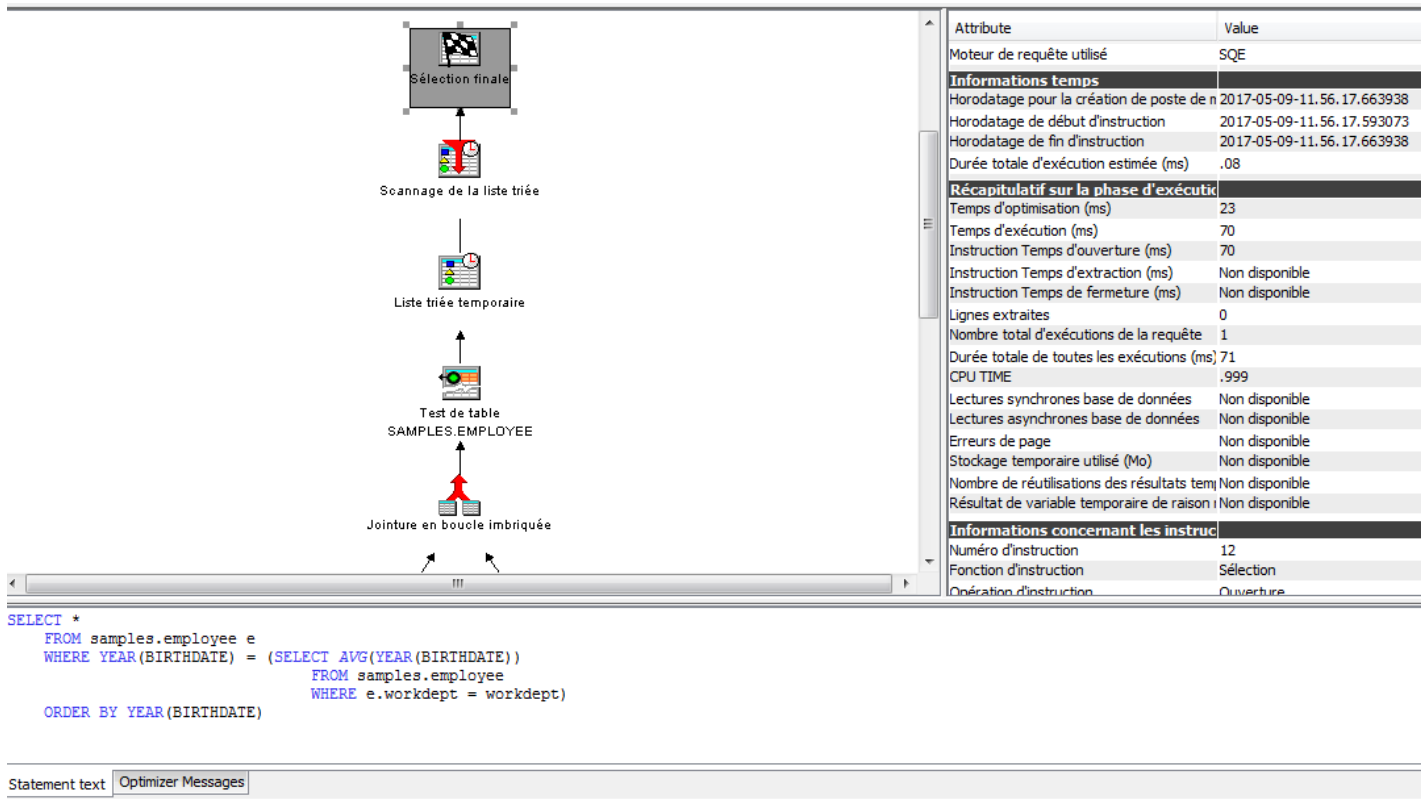
Attribute	Value
Moteur de requête utilisé	SQE
Informations temps	
Horodatage pour la création de poste de monitor	2017-05-06-16.09.32.318022
Horodatage de début d'instruction	2017-05-06-16.09.32.156161
Horodatage de fin d'instruction	2017-05-06-16.09.32.318022
Durée totale d'exécution estimée (ms)	21,477
Récapitulatif sur la phase d'exécution en cours	
Temps d'optimisation (ms)	55
Temps d'exécution (ms)	Non disponible
Instruction Temps d'ouverture (ms)	Non disponible
Instruction Temps d'extraction (ms)	Non disponible
Instruction Temps de fermeture (ms)	Non disponible
Lignes extraites	Non disponible
Nombre total d'exécutions de la requête	Non disponible
Durée totale de toutes les exécutions (ms)	Non disponible
CPU TIME	.599
Lectures synchrones base de données	Non disponible
Lectures asynchrones base de données	Non disponible
Erreurs de page	Non disponible
Stockage temporaire utilisé (Mo)	Non disponible
Nombre de réutilisations des résultats temporaires	Non disponible
Résultat de variable temporaire de raison non réutilisé	Non disponible
Informations concernant les instructions SQL exécutées	
Numéro d'instruction	12
Fonction d'instruction	Sélection
Opération d'instruction	Ouverture
Type d'instruction	Dynamique
Nom d'instruction	STM0001
Sortie instruction	Non about
Code retour SQL	-666
SQLSTATE	57005
Nom du curseur	SCRSR0001
Nom du programme ou du module	QZDASRV

```

SELECT INDEX_NAME AS Logique_NL ,
CASE
WHEN SYSTEM_INDEX_NAME IS NOT NULL
THEN SYSTEM_INDEX_NAME
ELSE INDEX_NAME
END AS Logique_NC,
CASE
WHEN index_partition IS NOT NULL
THEN index_partition
ELSE INDEX_NAME
END AS Membre_Logique, index_schema AS Bib_Logique, QUERY_STATISTICS_COUNT Nb_util_optim,
QUERY_USE_COUNT AS Nb_util_exec, LAST_STATISTICS_USE AS Date_derniere_util_optim ,
LAST_QUERY_USE Date_derniere_util_exec, LOGICAL_PAGE_SIZE AS Taille_page_logique,
ACCPH_TYPE AS Taille_CA
FROM QSYS2.SYSPARTITIONINDEXES
WHERE SYSTEM_TABLE_SCHEMA IN ('SAMPLES', 'ESALIB') AND INDEX_TYPE = 'LOGICAL'
ORDER BY SYSTEM_TABLE_SCHEMA, index_name DESC
    
```

Visual Explain - Cadre bas instruction SQL

- Les informations affichés dans le cadre du bas suivant l'onglet sélectionné seront les suivantes :
 - Affichage de l'instruction SQL



The Visual Explain interface displays the execution plan for the provided SQL query. The plan consists of the following steps:

- Jointure en boucle imbriquée (Nested Loop Join)
- Test de table SAMPLES.EMPLOYEE (Table Scan)
- Liste triée temporaire (Temporary Sorted List)
- Scannage de la liste triée (Scanning of the sorted list)
- Sélection finale (Final Selection)

The performance metrics table is as follows:

Attribute	Value
Moteur de requête utilisé	SQE
Informations temps	
Horodatage pour la création de poste de n	2017-05-09-11.56.17.663938
Horodatage de début d'instruction	2017-05-09-11.56.17.593073
Horodatage de fin d'instruction	2017-05-09-11.56.17.663938
Durée totale d'exécution estimée (ms)	.08
Récapitulatif sur la phase d'exécution	
Temps d'optimisation (ms)	23
Temps d'exécution (ms)	70
Instruction Temps d'ouverture (ms)	70
Instruction Temps d'extraction (ms)	Non disponible
Instruction Temps de fermeture (ms)	Non disponible
Lignes extraites	0
Nombre total d'exécutions de la requête	1
Durée totale de toutes les exécutions (ms)	71
CPU TIME	.999
Lectures synchrones base de données	Non disponible
Lectures asynchrones base de données	Non disponible
Erreurs de page	Non disponible
Stockage temporaire utilisé (Mo)	Non disponible
Nombre de réutilisations des résultats tem	Non disponible
Résultat de variable temporaire de raison	Non disponible
Informations concernant les instruc	
Numéro d'instruction	12
Fonction d'instruction	Sélection
Opération d'instruction	Ouverture

The SQL statement text is:

```
SELECT *
FROM samples.employee e
WHERE YEAR(BIRTHDATE) = (SELECT AVG(YEAR(BIRTHDATE))
FROM samples.employee
WHERE e.workdept = workdept)

ORDER BY YEAR(BIRTHDATE)
```




Visual Explain - Cadre bas Message de l'Optimiseur

- Les informations affichés dans le cadre du bas suivant l'onglet sélectionné seront les suivantes :
 - Affichage des messages de l'optimiseur si le menu Option Show Optimizer Messages est activé
 - Un double click sur un message affichera le détail du message.

The screenshot shows the Visual Explain interface. At the top, there's a table with columns 'Message ID' and 'Message text'. Below it, an 'Options' menu is open, with 'Show Optimizer Messages' checked. The 'Message text' pane is open to a message with ID 'CPI4326', showing detailed diagnostic information about a query execution plan and file access.

Message ID	Message text
CPI4339	Options de requête extraites du fichier QAQQINI dans la bibliothèque QTEMP. Que faire . . . : Options de
CPI4339	Options de requête extraites du fichier QAQQINI dans la bibliothèque QTEMP. Que faire . . . : Options de
CPI4323	Plan d'accès du processeur de requêtes reconstruit. Cause : Le plan d'accès a été reconstruit (code raison 0). La signification des différents codes rais...
CPI4339	Options de requête extraites du fichier QAQQINI dans la bibliothèque QTEMP. Que faire . . . : Options de requête extraites du fichier QAQQINI dans la bibliot...
CPI4339	Options de requête extraites du fichier QAQQINI dans la bibliothèque QTEMP. Que faire . . . : Options de requête extraites du fichier QAQQINI dans la bibliot...
CPI4344	Message de débogage de début **** de l'optimiseur pour la requête .Cause : Les messages de débogage de l'optimiseur qui suivent fournissent les inf...
CPI4325	Fichier résultat temporaire créé pour la requête .Cause : Un fichier temporaire a été créé pour la réception des résultats de la requête (code raison 7). ...
CPI4326	EMPLOYEE Fichier traité en position de jointure 1. Cause : Le chemin d'accès du membre EMPLOYEE du fichier EMPLOYEE de la bibliothèque SAMPLES a ...
CPI4326	EMPLOYEE Fichier traité en position de jointure 2. Cause : Le chemin d'accès du membre EMPLOYEE du fichier EMPLOYEE de la bibliothèque SAMPLES a ...
CPI4350	Tables de requêtes matérialisées prises en compte dans l'optimisation. Cause : L'optimiseur de requêtes a étudié l'utilisation des tables de requêtes mat...
CPI434B	Message de débogage de fin **** pour la requête .Cause : Ce message marque la fin des messages de débogage pour cette requête. Tous les messa...

Options

- Show Optimizer Messages
- Graph Detail
- Attributes Detail

Message text | **Optimizer Messages**

Message ID: CPI4326

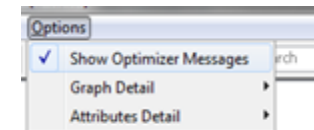
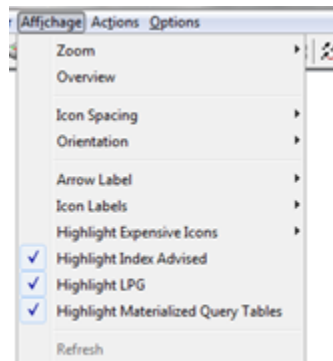
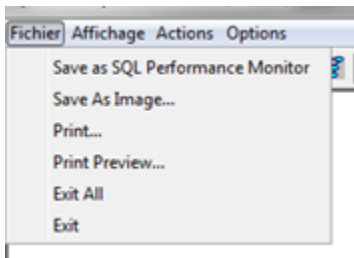
Message text:
 EMPLOYEE Fichier traité en position de jointure 1. Cause : Le chemin d'accès du membre EMPLOYEE du fichier EMPLOYEE de la bibliothèque SAMPLES a été utilisé pour accéder aux enregistrements du membre EMPLOYEE du fichier EMPLOYEE de la bibliothèque SAMPLES (code raison 1). Void la liste des codes raison et leur signification :
 1 - Sélection des enregistrements indiqués.
 2 - Exécution des critères de tri/groupage indiqués.
 3 - Sélection des enregistrements et critères de tri/groupage.
 4 - Exécution des critères de jointure indiqués.
 Si le fichier EMPLOYEE de la bibliothèque SAMPLES est un fichier logique, c'est le membre EMPLOYEE du fichier physique EMPLOYEE de la bibliothèque SAMPLES qui est en position de jointure 2.
 Si le nom de fichier du chemin d'accès commence par *TEMPX, il s'agit d'un chemin d'accès temporaire construit sur le fichier EMPLOYEE.
 Si le nom du fichier commence par *N ou *QUERY, il s'agit d'un fichier temporaire.
 L'accès par index seulement a été utilisé pour ce fichier dans le cadre de la requête : *NO.
 Pour l'accès par index seulement, la valeur *YES indique que toutes les zones utilisées à partir de ce fichier pour la requête peuvent être trouvées dans le chemin d'accès du fichier EMPLOYEE. La valeur *NO indique que l'accès par index seulement n'a pas pu être effectué pour ce chemin d'accès.
 L'accès par index seulement offre en général de meilleures performances, puisque toutes les données peuvent être extraites à partir du chemin d'accès et qu'il n'est pas nécessaire de paginer l'espace de données dans la mémoire active.
 Que faire . . . : En règle générale, pour forcer le traitement d'un fichier en position de jointure 1, indiquez une zone de tri uniquement pour ce fichier.
 Pour obtenir un tri, l'indication de zones de tri à l'instruction ORDER BY pour plus d'un fichier entraîne la création d'un fichier temporaire et permet l'optimisation du tri de tous les fichiers en jointure. Aucun fichier n'est forcé en première position.
 L'accès par index seulement ne peut être effectué sur un chemin d'accès que si toutes les zones utilisées dans la requête pour ce fichier sont également des zones dés de ce chemin d'accès.
 Pour des conseils supplémentaires sur l'optimisation de l'ordre de jointure d'une requête et de l'accès par le seul index, reportez-vous à l'ensemble de rubrique DB2 for i - Database Performance and Query Optimization dans la catégorie Database de l'IBM i Information Center.

Barre de menu de Visual Explain

- La barre de menu de Visual Explain se présente ainsi :



- Les menus Fichiers, Affichage, Actions et Options permettent d'afficher les options suivantes :











Barre d'icônes Visual Explain


- La barre d'icônes de Visual Explain se présente ainsi :

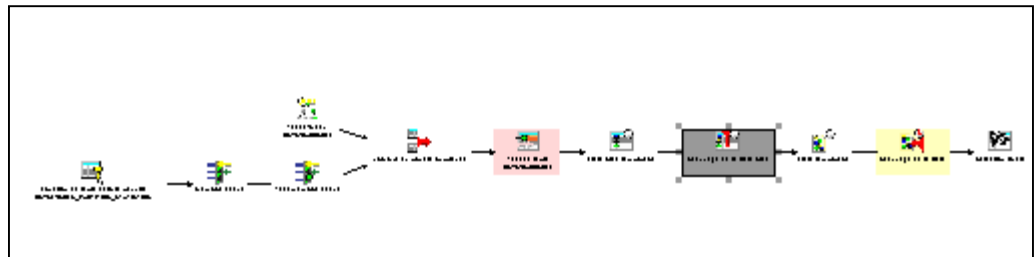
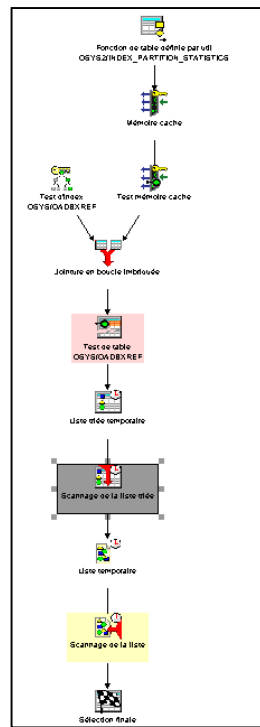
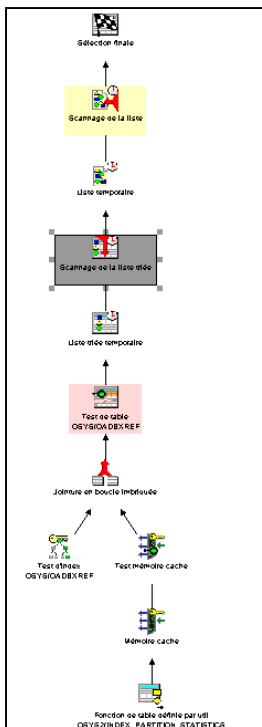


- Les icônes permettent les actions suivantes :

-  – Sauvegarde sous la forme de Moniteur de Performance.
-  – D'imprimer le graphique affiché
-  – De zoomer le graphique affiché (4 boutons)
-  – D'orienter le graphique (Gauche, Droite, Bas , Haut)
-  – De rafraichir le graphique si l'exécution est en mode Explain while Running
-  – D'envisager un nouveau plan d'accès si exécution en mode Explain while Running.
-  – D'afficher les index conseillés.
-  – D'effectuer une recherche sur le cadre de gauche (en ignorant ou non la casse).

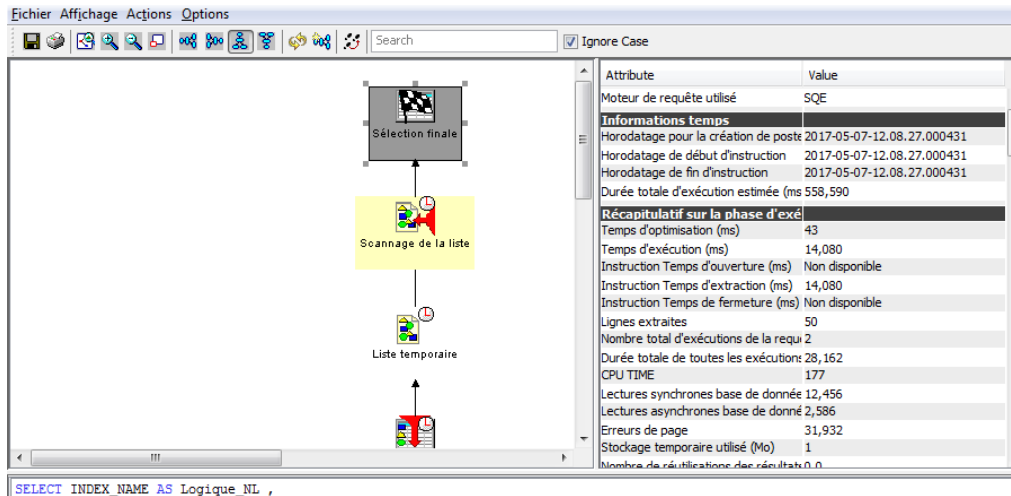
Visual Explain – Présentation du Graphique

- Le graphique présenté dans le cadre de droite peut être manipulé :
 - Par l'intermédiaire de la molette de défilement il peut être zoomé.
 - Par le biais des icones suivantes  son orientation peut être modifiée.




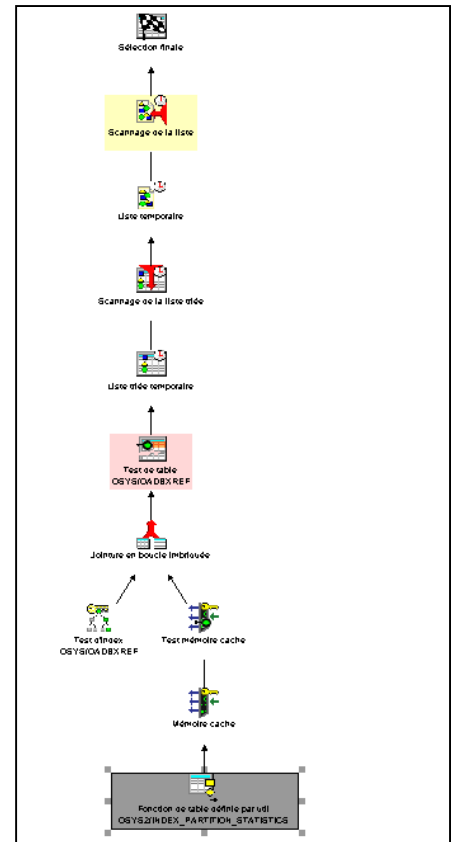
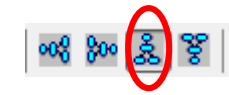
Visual Explain – Présentation

- Le cadre gauche présente le Graphique d'implémentation de la requête.
 - C'est la représentation graphique de la mise en œuvre de l'instruction SQL ainsi que des méthodes utilisées afin accéder à la base de données.
 - Les flèches indiquent l'ordre des étapes. Chaque nœud du graphique a une icône qui représente une opération ou les valeurs renvoyées d'une opération.
- Le cadre de droite de l'affichage présente les attributs et les valeurs de la requête.
 - L'affichage correspond à l'élément qui a été sélectionné sur le graphique, sont affichés les attributs et les valeurs de la requête correspondant à l'icône sélectionné.



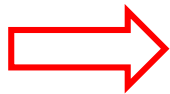
Visual Explain – Analyse du Graphique

- Le graphique s’analyse de bas en haut.
 - Il est conseillé de le présenter en utilisant l’icone 
 - Les étapes de la requête exécutée s’analysent de bas en haut, jusqu’à la sélection dite finale.
 - Dès lors que l’on sélectionne une icone, le cadre de gauche est mis à jour afin d’explicitier les actions mise en œuvre par l’optimiseur.



Visual Explain - Visualisation de l'Environnement d'exécution

- Après avoir sélectionner l'icone Sélection et avoir fait un click bouton droit :
 - Il est possible d'accéder aux caractéristiques de l'environnement d'exécution.



Attribute	Value
Nom du système	POWER8
Nom du travail	QZDASOINIT
Utilisateur du travail	QUSER
Numéro du travail	743881
Utilisateur	ESAS
Taille du pool de mémoire	2.822E10
Format de date	ISO
Séparateur de date	-
Format d'heure	ISO
Séparateur d'heure	.
Symbole décimal	.

Visual Explain – Sélection Finale 1/2

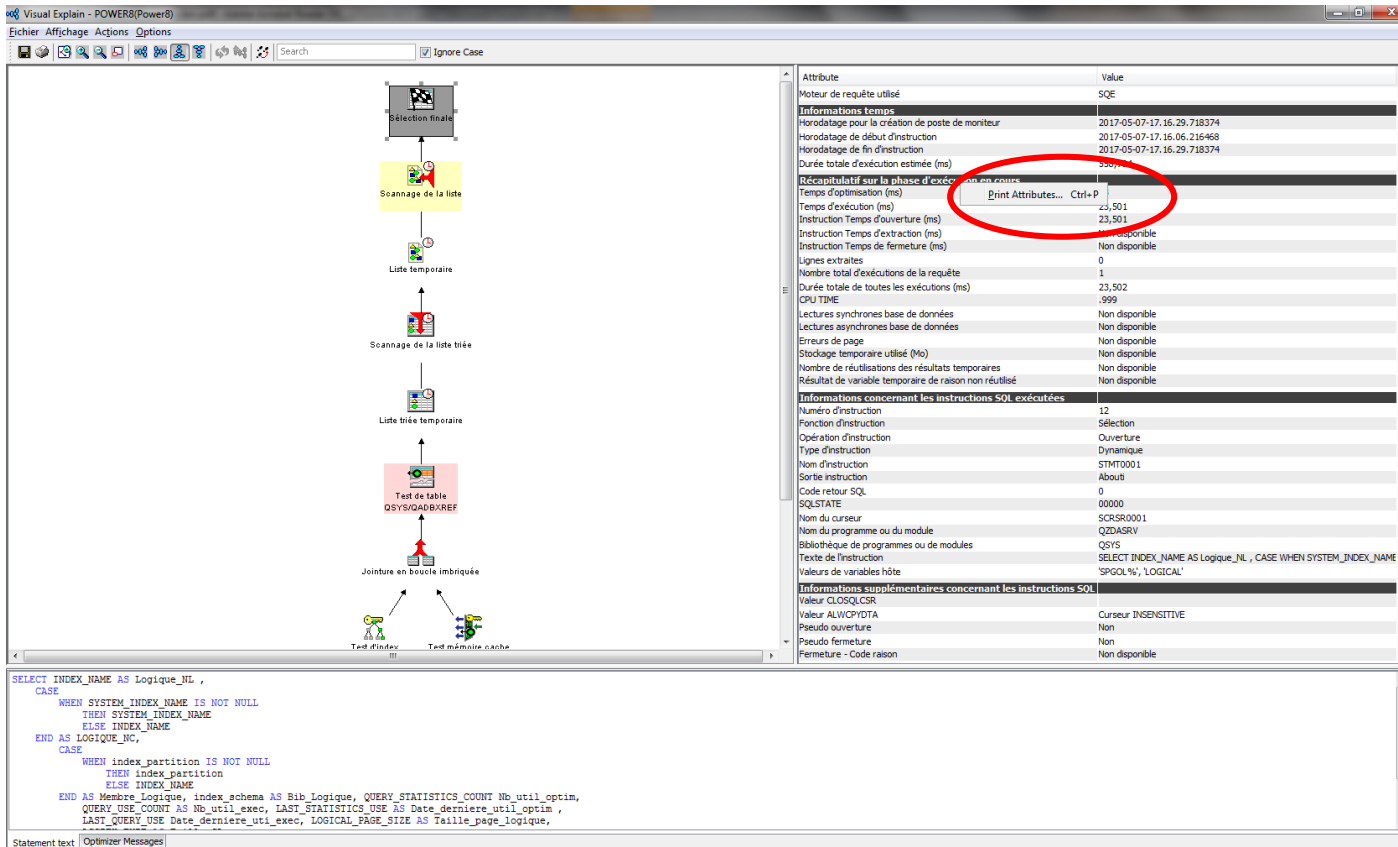


- Amener le curseur sur cet Icone permettra de voir dans le cadre de droite les information suivantes :
 - Le moteur utilisé.
 - Les informations sur la durée.
 - Des récapitulatifs sur la phase d'exécution en cours.
 - Les informations concernant les instructions SQL exécutées.
 - Les informations supplémentaires concernant les instructions.
 - Les informations supplémentaires concernant les instructions SQL.
 - Les informations communes à la plupart des lignes du moniteur.
 - Les options du fichier QAQQINI.
 - Les informations d'environnement pour les instructions SQL.

Visual Explain – Sélection Finale 1/3



- Ces informations se présentent ainsi, en utilisant le bouton droit de la souris il est possible de les imprimer.



The screenshot shows the Visual Explain - POWER8 interface. On the left, a flowchart illustrates the execution process, with the 'Sélection finale' step highlighted in yellow. On the right, a table provides detailed performance metrics for the selected step. A red circle highlights the 'Print Attributes...' button in the table.

Attribute	Value
Moteur de requête utilisé	SQE
Informations Temps	
Horodatage pour la création de poste de moniteur	2017-05-07-17.16.29.718374
Horodatage de début d'instruction	2017-05-07-17.15.06.216468
Horodatage de fin d'instruction	2017-05-07-17.16.29.718374
Durée totale d'exécution estimée (ms)	33077
Récapitulatif sur la phase d'exécution en cours	
Temps d'optimisation (ms)	Print Attributes... Ctrl+P
Temps d'exécution (ms)	23,501
Instruction Temps d'ouverture (ms)	23,501
Instruction Temps d'extraction (ms)	Non disponible
Instruction Temps de fermeture (ms)	Non disponible
Lignes extraites	0
Nombre total d'exécutions de la requête	1
Durée totale de toutes les exécutions (ms)	23,502
CPU TIME	999
Lectures synchrones base de données	Non disponible
Lectures asynchrones base de données	Non disponible
Erreurs de page	Non disponible
Stockage temporaire utilisé (%)	Non disponible
Nombre de réutilisations des résultats temporaires	Non disponible
Résultat de variable temporaire de raison non réutilisé	Non disponible
Informations concernant les instructions SQL exécutées	
Numero d'instruction	12
Fonction d'instruction	Sélection
Opération d'instruction	Ouverture
Type d'instruction	Dynamique
Nom d'instruction	STMT0001
Sortie instruction	About
Code retour SQL	0
SQLSTATE	00000
Nom du curseur	SCRSR0001
Nom du programme ou du module	QZDASRV
Bibliothèque de programmes ou de modules	QSYS
Texte de l'instruction	SELECT INDEX_NAME AS Logique_NL , CASE WHEN SYSTEM_INDEX_NAME
Valeurs de variables hôte	'SPGOL%', 'LOGICAL'
Informations supplémentaires concernant les instructions SQL	
Valeur CURSOR	
Valeur ALWCOPYDTA	Curseur INSENSITIVE
Pseudo ouverture	Non
Pseudo fermeture	Non
Fermeture - Code raison	Non disponible

```

SELECT INDEX_NAME AS Logique_NL ,
CASE
WHEN SYSTEM_INDEX_NAME IS NOT NULL
THEN SYSTEM_INDEX_NAME
ELSE INDEX_NAME
END AS LOGIQUE_NC,
CASE
WHEN index_partition IS NOT NULL
THEN index_partition
ELSE INDEX_NAME
END AS Membre_Logique, index_schema AS Bib_Logique, QUERY_STATISTICS COUNT Nb_util_optim,
QUERY_USE_COUNT AS Nb_util_exec, LAST_STATISTICS USE AS Date_derniere_util_optim ,
LAST_QUERY_USE Date_derniere_util_exec, LOGICAL_PAGE_SIZE AS Taille_page_Logique,
    
```



Visual Explain – Sélection Finale 3/3

■ Extrait de l'impression

Sélection finale

Attribute	Value
Moteur de requête utilisé	SQL
Informations temps	
Horodatage pour la création de poste de moniteur	2017-05-07-17.16.29.718374
Horodatage de début d'instruction	2017-05-07-17.16.06.216468
Horodatage de fin d'instruction	2017-05-07-17.16.29.718374
Durée totale d'exécution estimée (ms)	556,704
État: réutilisé sur la phase d'exécution en cours	
Temps d'optimisation (ms)	43
Temps d'exécution (ms)	23,501
Instruction Temps d'ouverture (ms)	23,501
Instruction Temps d'extraction (ms)	Nondisponible
Instruction Temps de fermeture (ms)	Nondisponible
Lignes extraites	0
Nombre total d'exécutions de la requête	1
Durée totale de toutes les exécutions (ms)	23,502
CPU TIME	.999
Lectures synchrones base de données	Nondisponible
Lectures asynchrones base de données	Nondisponible
Erreurs de page	Nondisponible
Stockage temporaire utilisé (Mo)	Nondisponible
Nombre de réutilisations des résultats temporaires	Nondisponible
Résultat de variable temporaire de raison non réutilisé	Nondisponible
Informations concernant les instructions SQL exécutées	
Numéro d'instruction	12
Fonction d'instruction	Sélection
Opération d'instruction	Ouverture
Type d'instruction	Dynamique
Nom d'instruction	STMT0001
Sortie d'instruction	About
Code retour SQL	0
SQLSTATE	00000
Nom du curseur	SCRSR0001
Nom du programme ou du module	QZDASRV
Bibliothèque de programmes ou de modules	QSYS
Texte de l'instruction	SELECT INDEX_NAME AS Logique_NL, CASE WHEN SYSTEM_INDE
Valeurs de variables hôte	'SPGOL%', 'LOGICAL'
Informations supplémentaires concernant les instructions	
Valeur CLOSQCSR	
Valeur ALWCPYDTA	Curseur INSENSITIVE
Pseudo ouverture	Non
Pseudo fermeture	Non
Fermeture - Code raison	Nondisponible
Mise en oeuvre ODP	Réutilisable
Replanification dynamique - Code raison	Plan d'accès non reconstruit
Horodatage lors de la création du plan	2017-05-06-17.09.16.052658
Code raison conversion de données	Non applicable
Groupe actif	ALWBLK(*ALLREAD)
Durée de prép	Oui
Instruction explicitable par un EXPLAIN	Oui
Convention d'appellation	Système
Type de traitement dynamique	Instruction préparée en local
Chemin SQL	QSYS1_QSYS2_QQLPSYS_QJUSRSYS_SIXAXE_QIWS
Résolution des accès simultanés utilisée	WAIT FOR OUTCOME
Numéro de port IP	8,471

Sélection finale

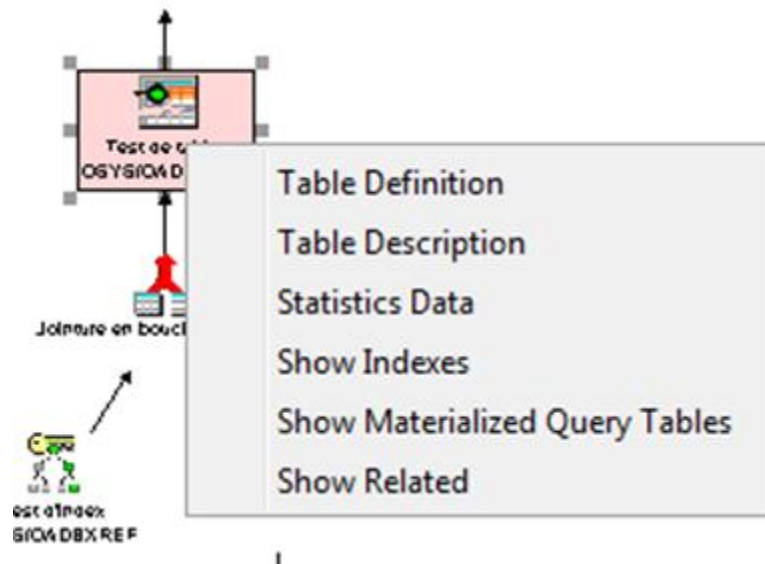
Attribute	Value
CCSID des données XML	1,208
AQP utilisé dans le plan d'accès	Non
Itération du plan d'accès AQP	1
Contrôle d'accès	Néant
Informations communes à la plupart des lignes du moniteur	
Nom du système	POWER8
Nom du travail	QZDASOINIT
Utilisateur du travail	QUSER
Numéro du travail	743247
Utilisateur	ESAS
Nombre de requêtes uniques	23
Actualiser le comptage	0
Options INI	
Bibliothèque d'options de requête	QTEMP
Table d'options de requête	QAQJINI
Commande de jointure forcée	Non
Réoptimisation de plan d'accès	Non
Nom de la base de données relationnelle	POWER8
Valeur ALWCPYDTA	Curseur INSENSITIVE
Définition de degré parallèle	*OPTIMIZE 100
Objectif d'optimisation	*FIRSTIO
Jointure en étoile	Non applicable
Limite mémoire de la fonction de régulation de requête, en seconde	2,147E9
Limite mémoire de la fonction de régulation de requête	2,147E9
Affichage des messages de débogage	Non
Conversion de marque de paramètre	Oui
Décalages redondants de LIKE ignorés	Optimisation
Admettre optimisation de longueur variable	Par défaut
Ignorer index dérivé	Par défaut
Normalisation données	Non
Mettre en mémoire cache les résultats temporaires	Système
Mode d'arrondi: DECFLOAT	Nondisponible
Graphique Visual Explain	De base
Utilisation de la table de requêtes matérialisée	Néant
Age de régénération de la table de requêtes matérialisée	0
Informations d'environnement pour instruction SQL	
Taille du pool de mémoire	2,821E10
MEMORY POOL ID	*BASE
Partage mémoire disponible (octets)	7,235E8
Moyenne utilisation active	39
Nombre maximal d'unités d'exécution actives admises dans pool	1,001
Moyenne activité dans le pool	39
Nombre de processeurs	2 (16)
Groupe de charge de travail spécifié	Non
Unités de traitement	2
Format de date	ISO
Séparateur de date	-
Format d'heure	ISO
Séparateur d'heure	.
Symbole décimal	.
Bibliothèque de séquence de tri	Néant
Table de séquence de tri	Néant
ID langue	FRA
Définition de degré parallèle	*OPTIMIZE 100
Nombre maximal de tâches	Nondisponible
Option suspension ROLLBACK	La position du curseur peut demeurer inconnue
Résolution des accès simultanés indiquée	WAIT FOR OUTCOME

Sélection finale

Attribute	Value
Temps d'attente fichier	31
Admettre les index temporaires	Oui
Traitement interactif	Non
Options de positionnement	Tout
Optimisation pour N lignes	30
Extraction N premières lignes	Toutes les lignes
Niveau contrôle de validation	WITH CS
Degré en cours	Nondisponible
Utilisateur session	Nondisponible
Utilisateur système	Nondisponible
Mode débogage	Nondisponible
Code comptabilité client	Windows?SQL=false;admin_user=true
Identificateur d'utilisateur client	esqglr
Nom d'application client	IBM Access Client Solutions - Run SQL Scripts
Nom de poste de travail client	SAP0W209.SIX-AXE.LOC
Identificateur de programme client	File(C:/Program%20Files%20(x86))/1_Access_Solution/acsbundle
Utilisateur en cours	Nondisponible
Optimisation d'ES à chaud demandée	Par défaut
Optimisation d'ES à chaud utilisée	Non
Objectif d'optimisation utilisé	Par défaut
Avertissements DECFLOAT	Non
Autoriser AQP	Oui
Embrasser les erreurs	Non
Comparaison des procédures de zone	Equal only
Autoriser les modifications de valeurs de tableau	Non
Option d'analyse XML implicite en cours	Nondisponible
QIO H4H4	956348D
Moteur de requête utilisé	SQL

Visual Explain – Information sur une Table

- En cas de sélection de l'icône représentant une table, en utilisant le bouton droit de la souris le menu suivant s'affichera permettant d'accéder aux informations suivantes :
- Définition de la table
- Description de la table
- Données statistiques
- Index
- MQT
- Objets liés.



Visual Explain – Définition de la table

- Le choix de la définition de la table permet de visualiser les informations suivantes :
 - Les caractéristique de la table
 - La liste des colonnes
 - Les contraintes de clés
 - Les contraintes de clés étrangères
 - La définition de la MQT (si MQT)
 - La liste des partitions

Table Columns Key Constraints Foreign Key Constraints

Name: EMPLOYEE

Schema: SAMPLES

System name: EMPLOYEE

Preferred storage media is solid-state drive

Keep in memory

Volatile data

Row access control

Column access control

Name	Type	Key Columns
Q_SAMPLES_EMPLO...	Primary key	EMPNO

Column Name	System Name	Data Type	Length	Null...	Generated Value	Default
EMPNO	EMPNO	CHARACTER	6	No		No defai
FIRSTNME	FIRSTNME	VARCHAR	12	No		No defai
MIDINIT	MIDINIT	CHARACTER	1	No		No defai
LASTNAME	LASTNAME	VARCHAR	15	No		No defai
WORKDEPT	WORKDEPT	CHARACTER	3	Yes		No defai
PHONENO	PHONENO	CHARACTER	4	Yes		No defai
HIREDATE	HIREDATE	DATE		Yes		No defai
JOB	JOB	CHARACTER	8	Yes		No defai
		LLINT		No		No defai
		RACTER	1	Yes		No defai
		E		Yes		No defai

Name	Key Columns	Parent Table	Parent Key Constraint
RED	WORKDEPT	SAMPLES/DEPAR...	Q_SAMPLES_DEPARTME.

Name	Check Condition
NUMBER	PHONENO >= '0000' AND PHONENO <= '9999'

Table Columns Key Constraints Foreign Key Constraints Check Constraints Materialized Query Partitioning

Partitioning type

Not partitioned

Partition by hash

Number of partitions:

Partition by range

Visual Explain – Description de la table

- Le choix de la description de la table permet de visualiser les informations suivantes :
 - Les informations générale.
 - Le nombre maximum d’enregistrements.
 - Les caractéristiques du chemin d’accès.
 - Les informations d’utilisations.
 - L’activité sur la table.
 - Des informations détaillées.

General	Allocation	Access Path	Usage	Activity	Details
<input checked="" type="checkbox"/> No maximum rows					
Initial number of rows:	<input type="text" value="0"/>				
Increment number of rows:	<input type="text" value="0"/>				
Maximum number of increments:	<input type="text" value="0"/>				
Current number of increments:	0				
Row capacity:	4294967288				
<input type="checkbox"/> Allocate storage					
<input type="checkbox"/> Contiguous storage					
Rows to force a write:	None				

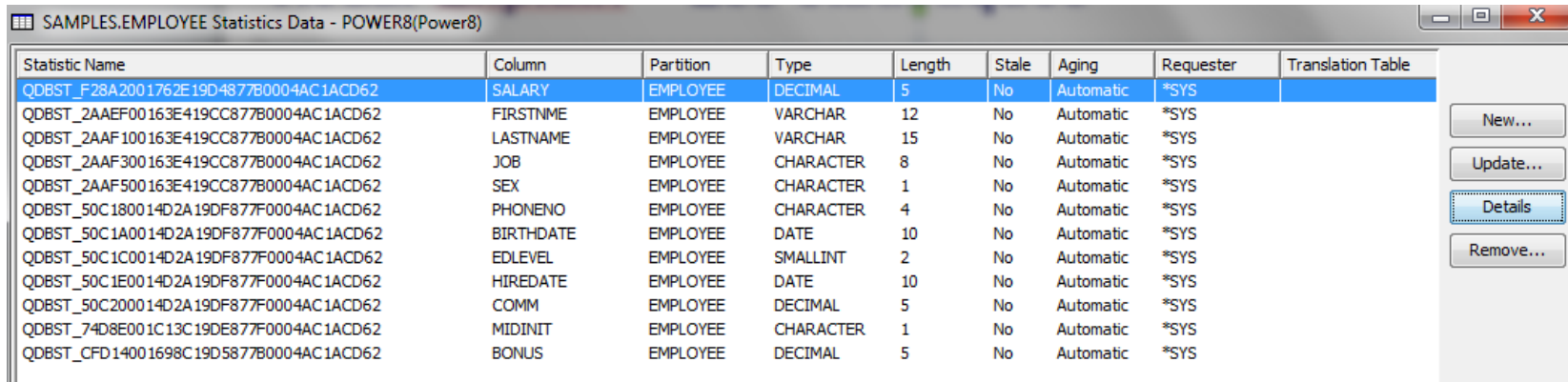
General	Allocation	Access Path	Usage	Activity
Since last restart				
Insert operations:	0			
Update operations:	0			
Delete operations:	0			
Logical reads:	0			
Physical reads:	0			
Sequential reads:	0			

General	Allocation	Access Path	Usage	Activity
Creation date:	11/01/16 19:51:07			
Last used:	08/05/17			
Days used count:	25			
Date count reset:	None			
Last change:	08/05/17 14:24:28			
Last source change:	None			
Last save:	06/05/17 23:21:43			
Last restore:	None			
Expiration date:	None			
Type:	Table			
<input type="checkbox"/> Share open data path				
Partition level identifier:	1160111195107			

General	Allocation	Access Path	Usage	Activity	Details
Type:	Table				
Description:	Tables des employés				
Data size:	76 KB				
Varying length data size:	0 bytes				
Current number of rows:	42				
Deleted rows:	0				
Rows in varying length storage:	0				
Maximum percent deleted rows allowed:	None				
<input checked="" type="checkbox"/> Reuse deleted rows					
Volatile:	General	Allocation	Access Path	Usage	Activity
Current size:	152 KB				
Maximum size:	1 TB				
Maximum key length:	6				
Page size:	Key length				
Valid:	Yes				
Implicit sharing:	Yes				
Held:	No				
Journalled:	No				
Maintenance:	Do not wait				
Recovery:	After IPL				
<input type="checkbox"/> Force index changes					
Last build:	19/12/16 15:12:33				
Estimated rebuild time:	1				
Logical page size:	65536				
Index logical reads:	0				
Delayed maintenance keys:	0				
Unique partial key values					
	0				0
	0				0

Visual Explain – Les statistiques 1/2

- Le choix de l'accès aux statistiques de la table permet de visualiser les statistiques et de pouvoir :
 - Les visualiser.
 - Les mettre à jour.
 - Les supprimer.
 - Les bloquer.



Statistic Name	Column	Partition	Type	Length	Stale	Aging	Requester	Translation Table
QDBST_F28A2001762E19D4877B0004AC1ACD62	SALARY	EMPLOYEE	DECIMAL	5	No	Automatic	*SYS	
QDBST_2AAEF00163E419CC877B0004AC1ACD62	FIRSTNAME	EMPLOYEE	VARCHAR	12	No	Automatic	*SYS	
QDBST_2AAF100163E419CC877B0004AC1ACD62	LASTNAME	EMPLOYEE	VARCHAR	15	No	Automatic	*SYS	
QDBST_2AAF300163E419CC877B0004AC1ACD62	JOB	EMPLOYEE	CHARACTER	8	No	Automatic	*SYS	
QDBST_2AAF500163E419CC877B0004AC1ACD62	SEX	EMPLOYEE	CHARACTER	1	No	Automatic	*SYS	
QDBST_50C180014D2A19DF877F0004AC1ACD62	PHONENO	EMPLOYEE	CHARACTER	4	No	Automatic	*SYS	
QDBST_50C1A0014D2A19DF877F0004AC1ACD62	BIRTHDATE	EMPLOYEE	DATE	10	No	Automatic	*SYS	
QDBST_50C1C0014D2A19DF877F0004AC1ACD62	EDLEVEL	EMPLOYEE	SMALLINT	2	No	Automatic	*SYS	
QDBST_50C1E0014D2A19DF877F0004AC1ACD62	HIREDATE	EMPLOYEE	DATE	10	No	Automatic	*SYS	
QDBST_50C200014D2A19DF877F0004AC1ACD62	COMM	EMPLOYEE	DECIMAL	5	No	Automatic	*SYS	
QDBST_74D8E001C13C19DE877F0004AC1ACD62	MIDINIT	EMPLOYEE	CHARACTER	1	No	Automatic	*SYS	
QDBST_CFD14001698C19D5877B0004AC1ACD62	BONUS	EMPLOYEE	DECIMAL	5	No	Automatic	*SYS	

Visual Explain – Les statistiques 2/2

- La visualisation des statistiques se présentes ainsi

Statistic Data Details

General | Estimated Value Ranges | Estimated Most Common Values

Statistic name: QDBST_F28A2001762E19D4877B0004AC1ACD62

Statistic last collected: 19/12/16 15:07:36

Requester: *SYS

Statistic created: 27/04/16 15:30:35

Creator: *SYS

Estimate of cardinality: 32

Column nullable: Yes

Number of nulls: 0

Number of rows when collected: 42

Current rows in table: 42

Inserts, updates, and deletes when collected: 85

Current inserts, updates and deletes: 88

Translation: No translation

Translation table:

Total space used for all statistics: 44 KB

Age statistics manually

Statistic Data Details

General | Estimated Value Ranges | Estimated Most Common Values

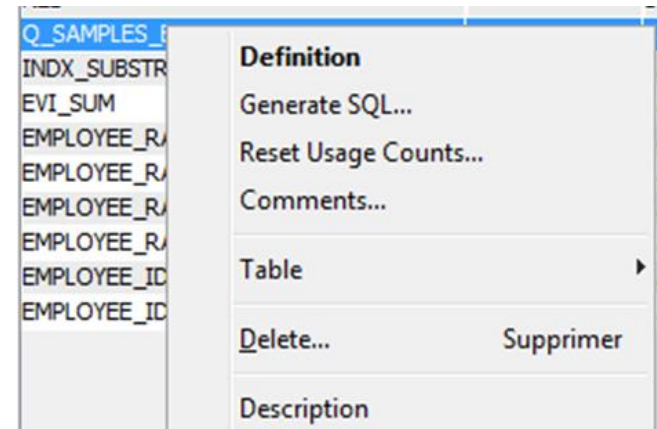
Lower Value	Upper Value	Count
	15340.00	1
15340.00	15900.00	2
15900.00	17250.00	1
17250.00	17750.00	1
17750.00	18270.00	1
18270.00	19180.00	1
19180.00	19950.00	1
19950.00	20450.00	1
20450.00	21340.00	1

General | Estimated Value Ranges | Estimated Most Common Values

Value	Count
29840.00	2
15900.00	2
28420.00	2
28760.00	2
29250.00	2
24680.00	2
46500.00	2
25370.00	2
23840.00	2
26250.00	2
22250.00	1
23800.00	1

Visuel Explain – Index

- Le choix de l'accès aux index de la table permet de visualiser la liste de tous les index pointant sur la table :
- Pour chaque index en faisant un click droit sur l'index, il est possible d'accéder aux informations et opérations suivantes :
 - La définition de l'index
 - De générer le DDL de l'index
 - De remettre à blanc les compteurs d'usage
 - D'accéder à un menu Table
 - De les supprimer.
 - D'accéder à leur description.



Visual Explain – Index Description

- Informations accessibles

SAMPLES/WORKDEPT_SELECT - POWER8(Power8)

Index | Key | WHERE clause | Options

Name: WORKDEPT_SELECT

Schema: SAMPLES

System name: WORKD00001

Table schema: SAMPLES

Table name: EMPLOYEE

Index type: Not unique

Preferred storage media is solid-state drive

SAMPLES/WORKDEPT_SELECT - POWER8(Power8)

Index | Key | WHERE clause | Options

Order	Name	System Name
Ascending	WORKDEPT	WORKDEPT

SAMPLES/WORKDEPT_SELECT - POWER8(Power8)

Index | Key | WHERE clause | Options

Subset of rows included in the index:

WORKDEPT IN ('D11', 'E11')

SAMPLES/WORKDEPT_SELECT - POWER8(Power8)

Index | Key | WHERE clause | Options

Number of distinct values: 0

Page size: Key length

Sort sequence table: None (Sort by hexadecimal value)

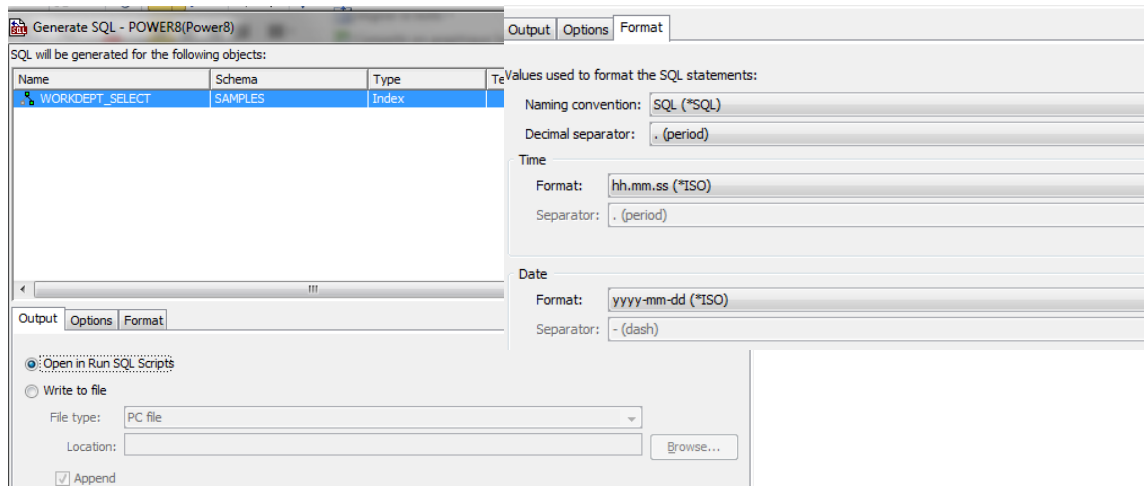
Additional non-key data

Include aggregate functions

Aggregates:

Visual Explain – Index Génération SQL

- Le choix de la génération de l'ordre SQL affichera la boîte de dialogue suivante :

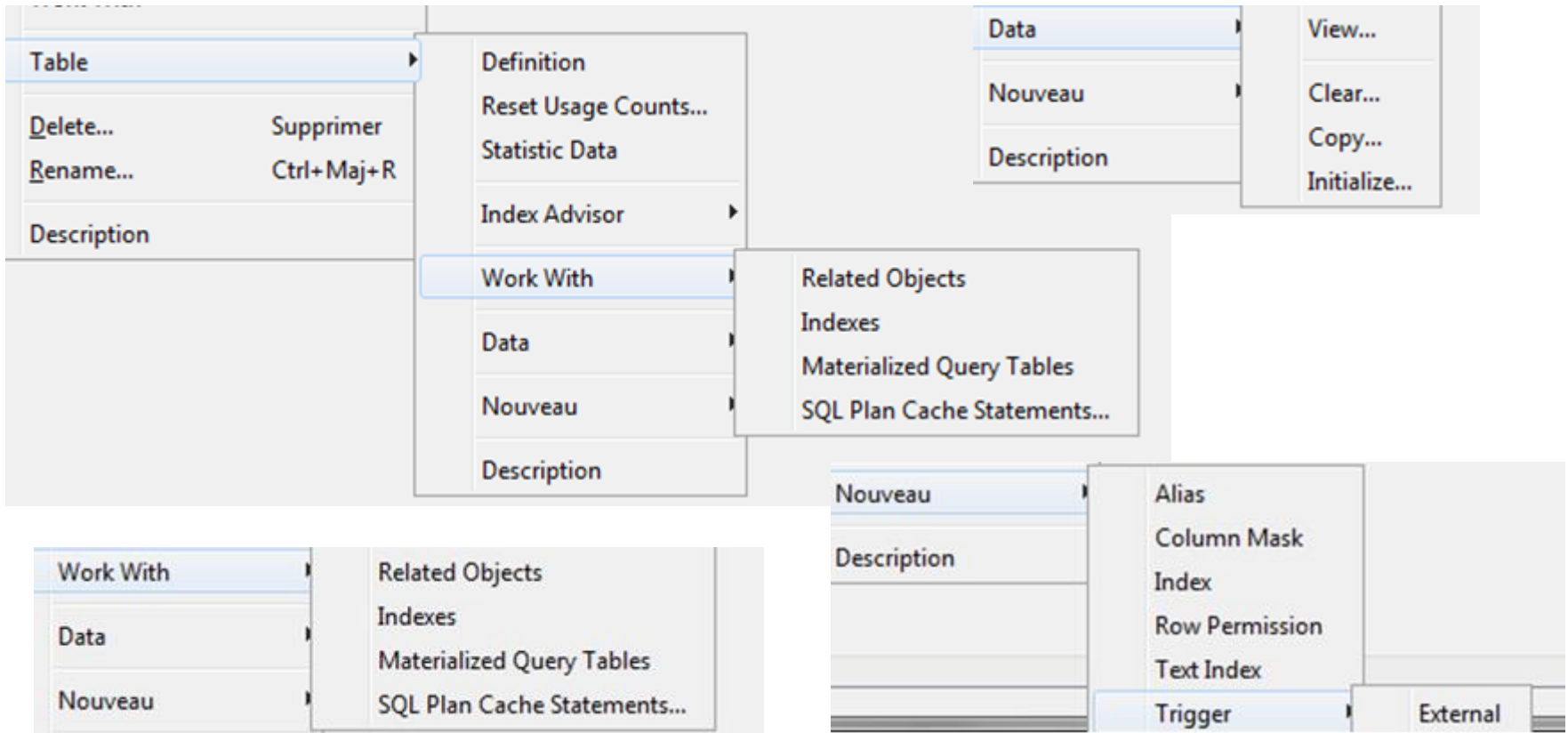


La validation affichera Script SQL

```
-- Générer SQL
-- Version :                V7R2M0 140418
-- Générée le :             08/05/17 15:14:43
-- Base données relation :  POWER8
-- Option normes :         DB2 for i
CREATE INDEX SAMPLES.WORKDEPT_SELECT
  FOR SYSTEM NAME WORKD00001 ON SAMPLES.EMPLOYEE(WORKDEPT ASC)
  WHERE WORKDEPT IN ('D11' , 'E11')
  RCDFMT WORKD00001;
```

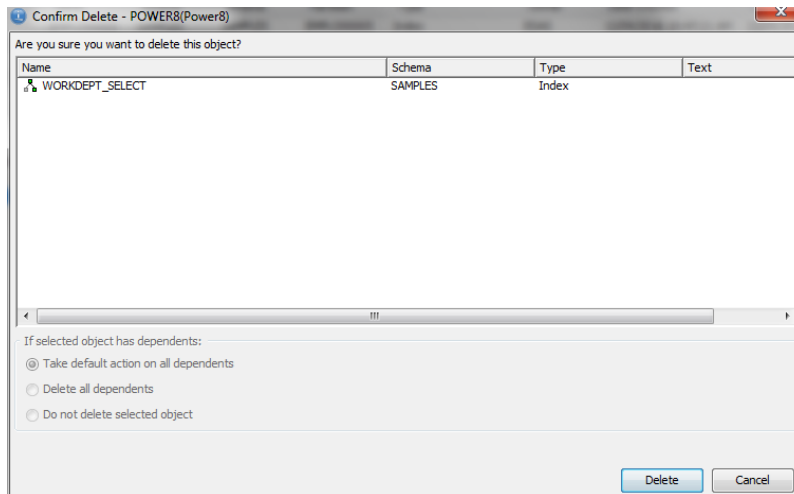
Visual Explain – Index Table

- En regard d'un index, en sélectionnant Table, les sous-menus suivants s'afficheront :

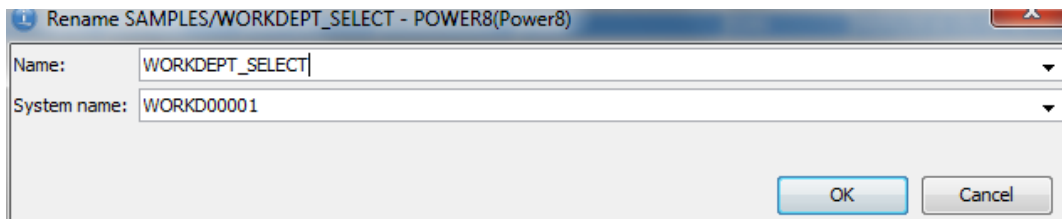


Visual Explain – Index Suppression/Rename

- La sélection en regard d'un index de Suppression affichera la boîte de dialogue suivante :



- La sélection de Rename affiche une boîte de dialogue permettant de modifier le nom ainsi que le nom système



Visual Explain – MQT

- Si la table sélectionnée est utilisée par des MQT (Materialize Query Table), l'information suivante est affichée :

Fichier Affichage Actions

MQTs for SAMPLES/EMPLOYEE

Name	System Name	Schema	System Schema	Partition	Owner	Enabled	Date Created	Last Refreshed
SUM_SALAIRE	SUM_S00001	SAMPLES	SAMPLES	SUM_S00001	ESAS	Oui	08/05/2017 02:12:36 PM	08/05/2017 02:12:36 PM
SUM_WRKDEPT	SUM_W00001	SAMPLES	SAMPLES	SUM_W00001	ESAS	Oui	20/09/2016 10:17:45 AM	20/09/2016 10:18:09 AM

- En regard de chaque MQT il est possible d'avoir accès aux informations suivantes :

Advised Indexes

Condense Advised Indexes

Clear All Advised Indexes...

MQTs for SAMPLES/EMPLOYEE

Name	System Name	Schema	System Schema	Partition	Owner
SUM_SALAIRE	SUM_S00001	SAMPLES	SAMPLES	SUM_S00001	ESAS
SUM_WRKDEPT	SUM_W00001	SAMPLES	SAMPLES	SUM_W00001	ESAS

Definition

Generate SQL...

Reset Usage Counts...

Statistic Data

Comments...

Index Advisor >

Work With >

Data

Delete... Supprimer

Rename... Ctrl+Maj+R

Nouveau

Description

View...

Clear...

Copy...

Initialize...

Refresh Table

Related Objects

Indexes

SQL Plan Cache Statements...



Visual Explain – Objets Liés 1/2

- La sélection en regard d'une table permettra d'accéder à l'ensemble des objets en relation avec cette table sous la forme suivante :

Fichier Affichage Actions						
Objects Related to SAMPLES/EMPLOYEE						
Name	System Name	Schema	Type	Constrained Table	Owner	Text
EMP	EMP	SAMPLES	Alias (Table)		ESAS	
NUMBER		SAMPLES	Check Constraint	SAMPLES.EMPLO...		
Q_SAMPLES_EMP_PHOTO_EMPNO_00002		SAMPLES	Foreign Key Constraint	SAMPLES.EMP_P...		
Q_SAMPLES_EMP_RESUME_EMPNO_00002		SAMPLES	Foreign Key Constraint	SAMPLES.EMP_R...		
Q_SAMPLES_PROJECT_RESPEMP_00001		SAMPLES	Foreign Key Constraint	SAMPLES.PROJECT		
RDE		SAMPLES	Foreign Key Constraint	SAMPLES.DEPAR...		
RED		SAMPLES	Foreign Key Constraint	SAMPLES.EMPLO...		
EMPLOYEE_IDX_TMP	EMPLO00005	SAMPLES	Index		ESAS	
EMPLOYEE_IDX_TMP1	EMPLO00006	SAMPLES	Index		ESAS	
EMPLOYEE_RADIX_INDEX_1	EMPLO00001	SAMPLES	Index		ESAS	
EMPLOYEE_RADIX_INDEX_2	EMPLO00002	SAMPLES	Index		ESAS	
EMPLOYEE_RADIX_INDEX_3	EMPLO00003	SAMPLES	Index		ESAS	
EMPLOYEE_RADIX_INDEX_4	EMPLO00004	SAMPLES	Index		ESAS	
EVI_SUM		SAMPLES	Index		ESAS	
INDX_SUBSTR	INDX_00001	SAMPLES	Index		ESAS	
WORKDEPT_SELECT	WORKD00001	SAMPLES	Index		ESAS	
XEMP1	XEMP1	SAMPLES	Index		ESAS	
XEMP2	XEMP2	SAMPLES	Index		ESAS	
QSQJRN	QSQJRN	SAMPLES	Journal		ESAS	Collection créée par SQL
Q_SAMPLES_EMPLOYEE_EMPNO_00001		SAMPLES	Primary Key Constraint	SAMPLES.EMPLO...		
SUM_SALAIRE	SUM_S00001	SAMPLES	Table (Materialized Query)		ESAS	
SUM_WRKDEPT	SUM_W00001	SAMPLES	Table (Materialized Query)		ESAS	
VASTRDE2	VASTRDE2	SAMPLES	View		ESAS	
VDEPMG1	VDEPMG1	SAMPLES	View		ESAS	
VEMP	VEMP	SAMPLES	View		ESAS	
VEMPDPT1	VEMPDPT1	SAMPLES	View		ESAS	
VEMPLP	VEMPLP	SAMPLES	View		ESAS	
VPHONE	VPHONE	SAMPLES	View		ESAS	
VPROJRE1	VPROJRE1	SAMPLES	View		ESAS	
VSTAFAC2	VSTAFAC2	SAMPLES	View		ESAS	
VUE_ESA	VUE_ESA	SAMPLES	View		ESAS	



Visual Explain – Objets Liés 2/2

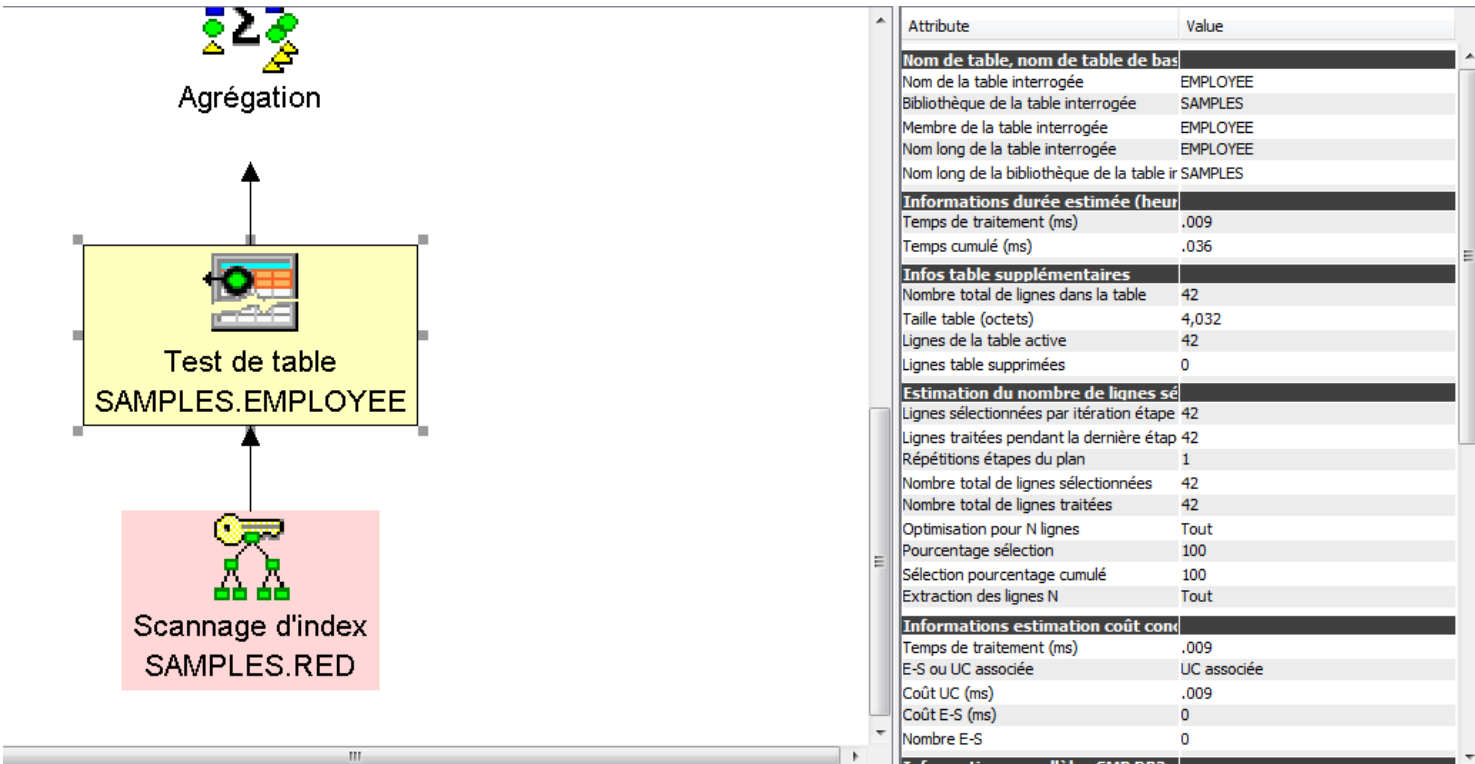
- La sélection d'un élément de cette liste permet de déclencher des actions soit par click bouton droit sur l'élément sélectionné, soit par le biais du menu Actions (les actions possibles dépendent du type d'objet)

Name	Schema	Type	Constrained Table	Owner	Text
EMP	SAMPLES	Alias (Table)		ESAS	
NUMBER	SAMPLES	Check Constraint	SAMPLES.EMPLO...		
Q_SAMPLES_EMP_P	SAMPLES	Foreign Key Constraint	SAMPLES.EMP_P...		
Q_SAMPLES_EMP_R	SAMPLES	Foreign Key Constraint	SAMPLES.EMP_R...		
Q_SAMPLES_PROJE	SAMPLES	Foreign Key Constraint	SAMPLES.PROJECT		
RDE	SAMPLES	Foreign Key Constraint	SAMPLES.DEPAR...		
RED	SAMPLES	Foreign Key Constraint	SAMPLES.EMPLO...		
EMPLOYEE_IDX_TMI	05 SAMPLES	Index		ESAS	
EMPLOYEE_IDX_TMI	06 SAMPLES	Index		ESAS	
EMPLOYEE_RADIX_I	01 SAMPLES	Index		ESAS	
EMPLOYEE_RADIX_I	02 SAMPLES	Index		ESAS	
EMPLOYEE_RADIX_INDEX_3	EMPLO00003	SAMPLES	Index	ESAS	
EMPLOYEE_RADIX_INDEX_4	EMPLO00004	SAMPLES	Index	ESAS	
EVI_SUM	EVI_SUM	SAMPLES	Index	ESAS	
INDX_SUBSTR	INDX_00001	SAMPLES	Index	ESAS	
WORKDEPT_SELECT	WORKD00001	SAMPLES	Index	ESAS	
XEMP1	XEMP1	SAMPLES	Index	ESAS	
XEMP2	XEMP2	SAMPLES	Index	ESAS	
QSQRJRN	QSQRJRN	SAMPLES	Journal	ESAS	Collection créée par SQL
Q_SAMPLES_EMPLOYEE_EMPNO_00001	SAMPLES	Primary Key Constraint	SAMPLES.EMPLO...		
SUM_SALAIRE	SUM_S00001	SAMPLES	Table (Materialized Query)	ESAS	
SUM_WRKDEPT	SUM_W00001	SAMPLES	Table (Materialized Query)	ESAS	
VASTRDE2	VASTRDE2	SAMPLES	View	ESAS	
VDEPMG1	VDEPMG1	SAMPLES	View	ESAS	
VEMP	VEMP	SAMPLES	View	ESAS	

Terminé : 31 lignes extraites

Visual Explain – Information du cadre de gauche

- Dès lors qu'une icône est sélectionnée, dans le cadre de gauche les informations sur les actions menées par l'optimiseur sont affichées, il est possible de les imprimer.



The screenshot shows the Visual Explain interface. On the left, a flow diagram illustrates the execution plan:

- Scannage d'index SAMPLES.RED** (Index Scan) is the starting point, represented by a key icon.
- An arrow points to **Test de table SAMPLES.EMPLOYEE** (Table Test), represented by a monitor icon.
- Another arrow points to **Agrégation** (Aggregation), represented by a funnel icon.

On the right, a table displays the details for the selected operation:

Attribute	Value
Nom de table, nom de table de base	
Nom de la table interrogée	EMPLOYEE
Bibliothèque de la table interrogée	SAMPLES
Membre de la table interrogée	EMPLOYEE
Nom long de la table interrogée	EMPLOYEE
Nom long de la bibliothèque de la table ir	SAMPLES
Informations durée estimée (heur	
Temps de traitement (ms)	.009
Temps cumulé (ms)	.036
Infos table supplémentaires	
Nombre total de lignes dans la table	42
Taille table (octets)	4,032
Lignes de la table active	42
Lignes table supprimées	0
Estimation du nombre de lignes sé	
Lignes sélectionnées par itération étape	42
Lignes traitées pendant la dernière étape	42
Répétitions étapes du plan	1
Nombre total de lignes sélectionnées	42
Nombre total de lignes traitées	42
Optimisation pour N lignes	Tout
Pourcentage sélection	100
Sélection pourcentage cumulé	100
Extraction des lignes N	Tout
Informations estimation coût con	
Temps de traitement (ms)	.009
E-S ou UC associée	UC associée
Coût UC (ms)	.009
Coût E-S (ms)	0
Nombre E-S	0

Visual Explain – Impression du cadre de Gauche

- Informations obtenues en demandant l'impression via Click bouton droit

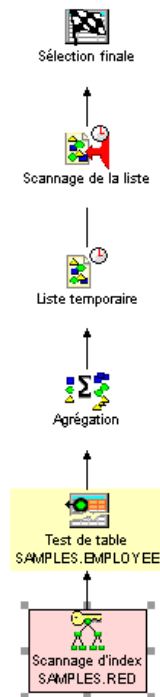
Attribute	Value
Nom de table, nom de base	
Nom de l'index utilisé	Print Attributes... Ctrl+P
Bibliothèque de l'index utilisé	SAMPLES

Attribute	Value
Nom de table, nom de base	
Nom de la table interrogée	EMPLOYEE
Bibliothèque de la table interrogée	SAMPLES
Membre de la table interrogée	EMPLOYEE
Nom long de la table interrogée	EMPLOYEE
Nom long de la bibliothèque de la table	SAMPLES
Informations durée estimée (heures)	
Temps de traitement (ms)	.009
Temps cumulé (ms)	.036
Infos table supplémentaires	
Nombre total de lignes dans la table	42
Taille table (octets)	4,032
Lignes de la table active	42
Lignes table supprimées	0
Estimation du nombre de lignes sélectionnées	
Lignes sélectionnées par itération étape	42
Lignes traitées pendant la dernière étape	42
Répétitions étapes du plan	1
Nombre total de lignes sélectionnées	42
Nombre total de lignes traitées	42
Optimisation pour N lignes	Tout
Pourcentage sélection	100
Sélection pourcentage cumulé	100
Extraction des lignes N	Tout
Informations estimation coût conc	
Temps de traitement (ms)	.009
E-S ou UC associée	UC associée
Coût UC (ms)	.009
Coût E-S (ms)	0
Nombre E-S	0
Informations parallèles SMP DB2	
Degré parallèle utilisé	1
Degré parallèle demandé	1
Degré parallèle max	1
Degré parallèle cumulé max	1
Degré E-S max	21
Récapitulatif sur la phase d'exécution	
Lignes actuelles sélectionnées par itération	Inconnu

Attribute	Value
Lignes actuelles traitées par itération étape	Inconnu
Itérations actuelles étapes du plan	Inconnu
Nombre total actuel de lignes sélectionnées	Inconnu
Nombre total de lignes traitées	Inconnu
Informations relatives au plan app	
Contient prédicat	Oui
Défilement possible	Oui
Nom du plan	TableProbe
Type des étapes du plan	Logic
Nom des étapes du plan	Node_32
Liste des index optimisés	SAMPLES/RED 0
Texte de l'instruction	SELECTEMPLOYEE_1.WORKDEPT, EMPLOYEE_1.BONUS, EMPLOYEE_1.SALARY FROM SAMPLES.EMPLOYEE EMPLOYEE_1 WHERE DseIdField(EMPLOYEE_1)=DseIdField(EMPLOYEE)

Visual Explain – Analyse d'une requête

- Soit la requête suivante
 - SELECT WORKDEPT, SUM(BONUS), SUM(SALARY) FROM SAMPLES.EMPLOYEE GROUP BY WORKDEPT HAVING SUM(bonus) > 5200
- La représentation graphique sera la suivante



Visual Explain – Analyse d’une requête

■ Impression de l’étape Scan d’index

Attribute	Value
Nom de table, nom de table de base	
Nom de l'index utilisé	EMPLOYEE
Bibliothèque de l'index utilisé	SAMPLES
Membre de l'index utilisé	EMPLOYEE
Nom long de l'index utilisé	RED
Nom long de la bibliothèque de l'index utilis	SAMPLES
Nom de la table interrogée	EMPLOYEE
Bibliothèque de la table interrogée	SAMPLES
Membre de la table interrogée	EMPLOYEE
Nom long de la table interrogée	EMPLOYEE
Nom long de la bibliothèque de la table in	SAMPLES
Accès par l'index seulement	Non
Informations durée estimée (heure)	
Temps de traitement (ms)	.016
Temps cumulé (ms)	.016
Infos index supplémentaires	
Nombre d'entrées d'index	42
Taille de zone clé	4
Taille de page logique de l'index	65,536
Longueur de clé variable	Non
Remplissage de toutes les zones	Non
Unique	Non
Contient une sélection fractionnée	Non
Les clés indéfinies sont fausses	Oui
Type d'index	Foreign Key Constraint
Liste des clés de l'index	Ascending,EMPLOYEE_1.WORKDEPT
Estimation du nombre de lignes sé	
Lignes sélectionnées par itération étape c	42
Lignes traitées pendant la dernière étape	42
Répétitions étapes du plan	1
Nombre total de lignes sélectionnées	42
Nombre total de lignes traitées	42
Optimisation pour N lignes	Tout
Pourcentage sélection	100
Sélection pourcentage cumulé	100
Extraction des lignes N	Tout
Informations estimation coût conc	
Temps de traitement (ms)	.016
E-S ou UC associée	UC associée

Attribute	Value
Coût UC (ms)	.016
Coût E-S (ms)	0
Nombre E-S	1
Temps de lecture E-S moyen	6.153
I/O COST REDUCTION	Oui
Relation de préchargement	Non
Mémoire utilisé (octets)	65,536
Mémoire contrainte	Non
Mémoire cumulée contrainte	Non
Informations parallèles SMP DB2	
Degré parallèle utilisé	1
Degré parallèle demandé	1
Degré parallèle max	1
Degré parallèle cumulé max	1
Degré E-S max	1
Récapitulatif sur la phase d'exécuti	
Lignes actuelles sélectionnées par itératic	Inconnu
Lignes actuelles traitées par itération étaj	Inconnu
Itérations actuelles étapes du plan	Inconnu
Nombre total actuel de lignes sélectionné	Inconnu
Nombre total de lignes traitées	Inconnu
Informations relatives au plan appl	
Contient classement	Oui
Défilement possible	Oui
Nom du plan	Index Scan
Type des étapes du plan	Logic
Code raison	Index pour groupage ou classement
Nom des étapes du plan	Node_35
Texte de l'instruction	SELECT DseIdField(EMPLOYEE), EMPLOY

Visual Explain – Analyse d’une requête

- L’impression de l’étape Test de Table

Attribute	Value
Nom de table, nom de table de bas	
Nom de la table interrogée	EMPLOYEE
Bibliothèque de la table interrogée	SAMPLES
Membre de la table interrogée	EMPLOYEE
Nom long de la table interrogée	EMPLOYEE
Nom long de la bibliothèque de la table ir	SAMPLES
Informations durée estimée (heur	
Temps de traitement (ms)	.009
Tempscumulé(ms)	.036
Infos table supplémentaires	
Nombre total de lignes dans la table	42
Taille table (octets)	4,032
Lignes de la table active	42
Lignes table supprimées	0
Estimation du nombre de lignes sé	
Lignes sélectionnées par itération étape	42
Lignes traitées pendant la dernière étape	42
Répétitions étapes du plan	1
Nombre total de lignes sélectionnées	42
Nombre total de lignes traitées	42
Optimisation pour N lignes	Tout
Pourcentage sélection	100
Sélection pourcentage cumulé	100
Extraction des lignes N	Tout
Informations estimation coût conc	
Temps de traitement (ms)	.009
E-S ou UC associée	UC associée
Coût UC (ms)	.009
Coût E-S (ms)	0
Nombre E-S	0
Informations parallèles SMP DB2	
Degré parallèle utilisé	1
Degré parallèle demandé	1
Degré parallèle max	1
Degré parallèle cumulé max	1
Degré E-S max	21
Récapitulatif sur la phase d'exéc	
Lignes actuelles sélectionnées par itérati	Inconnu

Attribute	Value
Lignes actuelles traitées par itération éti	Inconnu
Itérations actuelles étapes du plan	Inconnu
Nombre total actuel de lignes sélectionn	Inconnu
Nombre total de lignes traitées	Inconnu
Informations relatives au plan app	
Contient prédicat	Oui
Défilement possible	Oui
Nom du plan	TableProbe
Type des étapes du plan	Logic
Nom des étapes du plan	Node_32
Liste des index optimisés	SAMPLES/RED 0
Texte de l'instruction	SELECTEMPLOYEE_1.WORKDEPT, EMPLOYEE_1.BONUS, EMPLOYEE_1.SALARY FROM SAMPLES.EMPLOYEE EMPLOYEE_1 WHERE DseIdField(EMPLOYEE_1)=DseIdField(E MPLOYEE)

Visual Explain – Analyse d’une requête

■ L’impression de l’étape Agrégation

Agrégation

Attribute	Value
Informations durée estimée (heure)	
Temps de traitement (ms)	.007
Temps cumulé (ms)	.043
Estimation du nombre de lignes sé	
Lignes sélectionnées par itération étape c	2.666
Lignes traitées pendant la dernière étape 8	
Répétitions étapes du plan	1
Nombre total de lignes sélectionnées	2.666
Nombre total de lignes traitées	8
Optimisation pour N lignes	30
Pourcentage sélection	100
Sélection pourcentage cumulé	33.333
Extraction des lignes N	Tout
Informations estimation coût conc	
Temps de traitement (ms)	.007
E-S ou UC associée	UC associée
Coût UC (ms)	.007
Coût E-S (ms)	0
Nombre E-S	0
Données d’agrégation distinctes	
Agrégations	
Colonnes SUM	Sum(EMPLOYEE_1.BONUS), Sum(EMPL
Colonnes COUNT	
Colonnes MAX	
Colonnes MIN	
Informations parallèles SMP DB2	
Degré parallèle utilisé	1
Degré parallèle demandé	1
Degré parallèle max	1
Degré parallèle cumulé max	1
Degré E-S max	1
Récapitulatif sur la phase d’exécuti	
Lignes actuelles sélectionnées par itérati	Inconnu
Lignes actuelles traitées par itération étaj	Inconnu
Itérations actuelles étapes du plan	Inconnu
Nombre total actuel de lignes sélectionné	Inconnu
Nombre total de lignes traitées	Inconnu
Informations relatives au plan appl	

Agrégation

Attribute	Value
Contient regroupement	Oui
Contient prédicat	Oui
Contient agrégat	Oui
Défilement possible	Oui
Nom du plan	Aggregation
Type des étapes du plan	Logic
Nom des étapes du plan	Node_79
Texte de l’instruction	SELECTEMPLOYEE_1.WORKDEPT, Sum(EMPLOYEE_1.BONUS), Sum(EMPLOYEE_1.SALARY) FROM Node_32 GROUPBY EMPLOYEE_1.WORKDEPT HAVING Sum(EMPLOYEE_1.BONUS)> 5200.00

Visual Explain – Analyse d’une requête

- L’impression de l’étape Liste Temporaire

Liste temporaire

Attribute	Value
Informations durée estimée (heure)	
Tempscumulé(ms)	.044
Informations liste temporaire	
Taille liste totale estimée (octets)	4,096
Lignes dans liste	2.666
Taille des données autres que les clés (oc 38)	
Informations estimation coût conce	
Durée alimentation estimée	7.746E-4
E-S ou UC associée	UC associée
Coût UC (ms)	7.746E-4
Coût E-S (ms)	0
Nombre E-S	0
Temps de lecture E-S moyen	5.001
Informations parallèles SMP DB2	
Degré parallèle utilisé	1
Degré parallèle demandé	1
Degré parallèle max	1
Récapitulatif sur la phase d'exécuti	
Nombre réel d'entrées	Inconnu
Nombre réel de clés	Inconnu
Taille réelle en octets	Inconnu
Informations relatives au plan appl	
Texte de l'instruction	CREATE TEMPORARY LIST QdsPt(Node_79)(SELECT EMPLOYEE_1.WORKDEPT, Sum(EMPLOYEE_1.BONUS), Sum(EMPLOYEE_1.SALARY) FROM Node_32 GROUPBY EMPLOYEE_1.WORKDEPT HAVING Sum(EMPLOYEE_1.BONUS)>5200.00)

Visual Explain – Analyse d’une requête

- L’impression de l’étape Scannage de la liste

Scannage de la liste

Attribute	Value
Informations durée estimée (heure)	
Temps de traitement (ms)	2.106E-4
Temps cumulé (ms)	.044
Estimation du nombre de lignes sé	
Lignes sélectionnées par itération étape c	2.666
Lignes traitées pendant la dernière étape	3
Répétitions étapes du plan	1
Nombre total de lignes sélectionnées	2.666
Nombre total de lignes traitées	3
Optimisation pour N lignes	30
Pourcentage sélection	100
Sélection pourcentage cumulé	100
Extraction des lignes N	Tout
Informations estimation coût conce	
Temps de traitement (ms)	2.106E-4
E-S ou UC associée	UC associée
Coût UC (ms)	2.106E-4
Coût E-S (ms)	0
Nombre E-S	0
Informations parallèles SMP DB2	
Degré parallèle utilisé	1
Degré parallèle demandé	1
Degré parallèle max	1
Degré parallèle cumulé max	1
Degré E-S max	1
Récapitulatif sur la phase d’exécuti	
Lignes actuelles sélectionnées par itératic	Inconnu
Lignes actuelles traitées par itération éta	Inconnu
Itérations actuelles étapes du plan	Inconnu
Nombre total actuel de lignes sélectionné	Inconnu
Nombre total de lignes traitées	Inconnu
Informations relatives au plan appl	
Défilement possible	Oui
Nom du plan	Scan Temporary
Type des étapes du plan	Loaic
Nom des étapes du plan	Node_80

Scannage de la liste

Attribute	Value
Texte de l’instruction	SELECTEMPLOYEE_1.WORKDEPT, Sum(EMPLOYEE_1.BONUS), Sum(EMPLOYEE_1.SALARY) FROM QdsPtl(Node_79)

Visual Explain – Analyse d’une requête

- L’impression de l’étape Sélection Finale 1/2

Sélection finale

Attribute	Value
Moteur de requête utilisé	SQE
Informations temps	
Horodatage pour la création de poste de	2017-05-08-16.09.01.423907
Horodatage de début d'instruction	2017-05-08-16.09.01.419878
Horodatage de fin d'instruction	2017-05-08-16.09.01.423907
Durée totale d'exécution estimée (ms)	.044
Récapitulatif sur la phase d'exécution	
Temps d'optimisation(ms)	4
Temps d'exécution (ms)	4
Instruction Temps d'ouverture (ms)	4
Instruction Temps d'extraction (ms)	Nondisponible
Instruction Temps de fermeture (ms)	Nondisponible
Lignes extraites	0
Nombre total d'exécutions de la requête	1
Durée totale de toutes les exécutions (ms)	5
CPU TIME	.999
Lectures synchrones base de données	Nondisponible
Lectures asynchrones base de données	Nondisponible
Erreurs de page	Nondisponible
Stockage temporaire utilisé (Mo)	Nondisponible
Nombre de réutilisations des résultats ten	Nondisponible
Résultat de variable temporaire de raison	Nondisponible
Informations concernant les instructions	
Numéro d'instruction	18
Fonction d'instruction	Sélection
Opération d'instruction	Ouverture
Type d'instruction	Dynamique
Nom d'instruction	STMT0002
Sortie instruction	Abouti
Code retour SQL	0
SQLSTATE	00000
Nom du curseur	SCRSR0002
Nom du programme ou du module	QZDASRV
Bibliothèque de programmes ou de modules	QSYS
Texte de l'instruction	SELECT WORKDEPT, SUM(BONUS), SUM
Valeurs de variables hôte	5200
Liste des tables de requêtes matérialisées	SAMPLES/SUM_S00001 12, SAMPLES/SU
Informations supplémentaires con	
Valeur CLOSQLCSR	

Sélection finale

Attribute	Value
Valeur ALWCPYDTA	Curseur INSENSITIVE
Pseudo ouverture	Non
Pseudo fermeture	Non
Fermeture - Code raison	Nondisponible
Mise en oeuvre ODP	Réutilisable
Replanification dynamique - Code raison	Plan d'accès non reconstruit
Horodatage lors de la création du plan	2017-05-08-16.08.44.810593
Code raison conversion de données	Non applicable
Groupage activé	ALWBLK(*ALLREAD)
Durée de prép	Oui
Instruction explicitable par un EXPLAIN	Oui
Convention d'appellation	SQL
Type de traitement dynamique	Instruction préparée en local
Chemin SQL	"QSYS", "QSYS2", "SYSPROC", "SYSIBMAD
Résolution des accès simultanés utilisée	WAIT FOR OUTCOME
Numéro de port IP	8,471
Adresse IP du client	10.10.4.14
Type d'adresse IP	1
CCSID des données XML	1,208
AQP utilisé dans le plan d'accès	Non
Itération du plan d'accès AQP	1
Contrôle d'accès	Néant
Informations communes à la plupart	
Nom du système	POWER8
Nom du travail	QZDASOINIT
Utilisateur du travail	QUSER
Numéro du travail	744085
Utilisateur	ESAS
Nombre de requêtes uniques	857
Actualiser le comptage	0
Options INI	
Bibliothèque d'options de requête	QTEMP
Table d'options de requête	QAQQINI
Commande de jointure forcée	Non
Réoptimisation de plan d'accès	Non
Nom de la base de données relationnelle	POWER8
Valeur ALWCPYDTA	Curseur INSENSITIVE
Définition de degré parallèle	*OPTIMIZE 100
Objectif d'optimisation	*FIRSTIO

Visual Explain – Analyse d’une requête

- L’impression de l’étape Sélection Finale 2/2

Sélection finale

Attribute	Value
Jointure en étoile	Non applicable
Limite mémoire de la fonction de régulatix	2.147E9
Limite mémoire de la fonction de régulatix	2.147E9
Affichage des messages de débogage	Non
Conversion de marque de paramètre	Oui
Décalages redondants de LIKE ignorés	Optimisation
Admettre optimisation de longueur variab	Par défaut
Ignorer index dérivé	Par défaut
Normalisation données	Non
Mettre en mémoire cache les résultats ter	Système
Mode d'arrondi DECFLOAT	Nondisponible
Graphique Visual Explain	De base
Utilisation de la table de requêtes matéria	Tout
Age de régénération de la table de requê	0
Informations environnement pour l	
Taille du pool de mémoire	2.822E10
MEMORY POOL ID	*BASE
Partage mémoire disponible (octets)	6.135E8
Moyenne utilisation active	46
Nombre maximal d'unités d'exécution acti	1,001
Moyenne activité dans le pool	46
Nombre de processeurs	2 (16)
Groupe de charge de travail spécifié	Non
Unités de traitement	2
Format de date	ISO
Séparateur de date	-
Format d'heure	ISO
Séparateur d'heure	-
Symbole décimal	.
Bibliothèque de séquence de tri	Néant
Table de séquence de tri	Néant
ID langue	FRA
Définition de degré parallèle	*OPTIMIZE 100
Nombre maximal de tâches	Nondisponible
Option suspension ROLLBACK	La position du curseur peut demeurer ino
Résolution des accès simultanés indiquée	WAIT FOR OUTCOME
Temps d'attente fichier	31
Admettre les index temporaires	Oui
Traitement interactif	Non

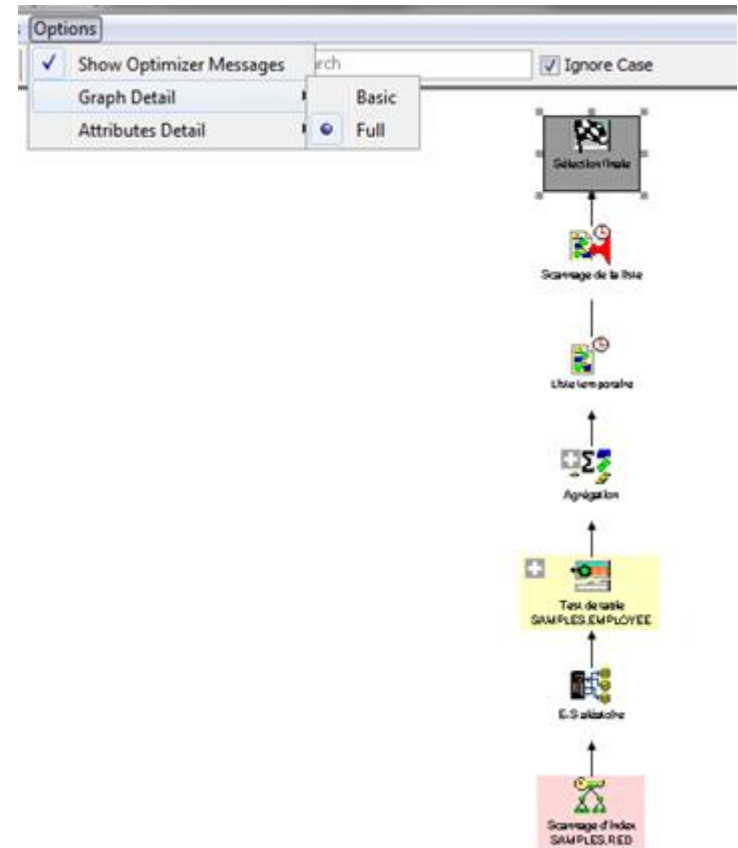
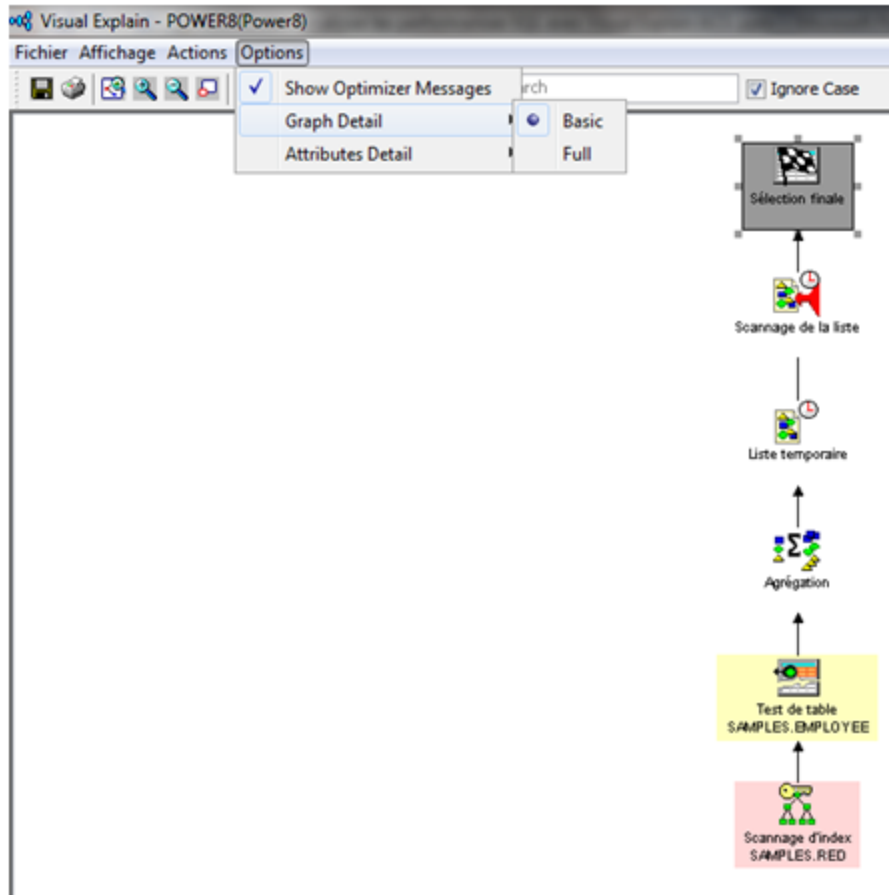
Sélection finale

Attribute	Value
Options de positionnement	Tout
Optimisation pour N lignes	30
Extraction N premières lignes	Toutes les lignes
Niveau contrôle de validation	WITH CS
Degré en cours	Nondisponible
Utilisateur session	Nondisponible
Utilisateur système	Nondisponible
Mode débogage	Nondisponible
Code comptabilité client	Windows7;SSL=false;admin_user=true
Identificateur d'utilisateur client	esaglier
Nom d'application client	IBM I Access Client Solutions - Run SQL Sc
Nom de poste de travail client	SAPOW209.SDX-AXE.LOC
Identificateur de programme client	file:/C:/Program%20Files%20(x86)/I_Ac
Utilisateur en cours	Nondisponible
Optimisation d'E/S à chaud demandée	Par défaut
Optimisation d'E/S à chaud utilisée	Non
Objectif d'optimisation utilisé	Par défaut
Avertissements DECFLOAT	Non
Autoriser AQP	Oui
Interclasser les erreurs	Non
Comparaison des procédures de zone	Equal only
Autoriser les modifications de valeurs de l	Non
Option d'analyse XML implicite en cours	Nondisponible
QRO HASH	3D86BB2E
Moteur de requête utilisé	SQE



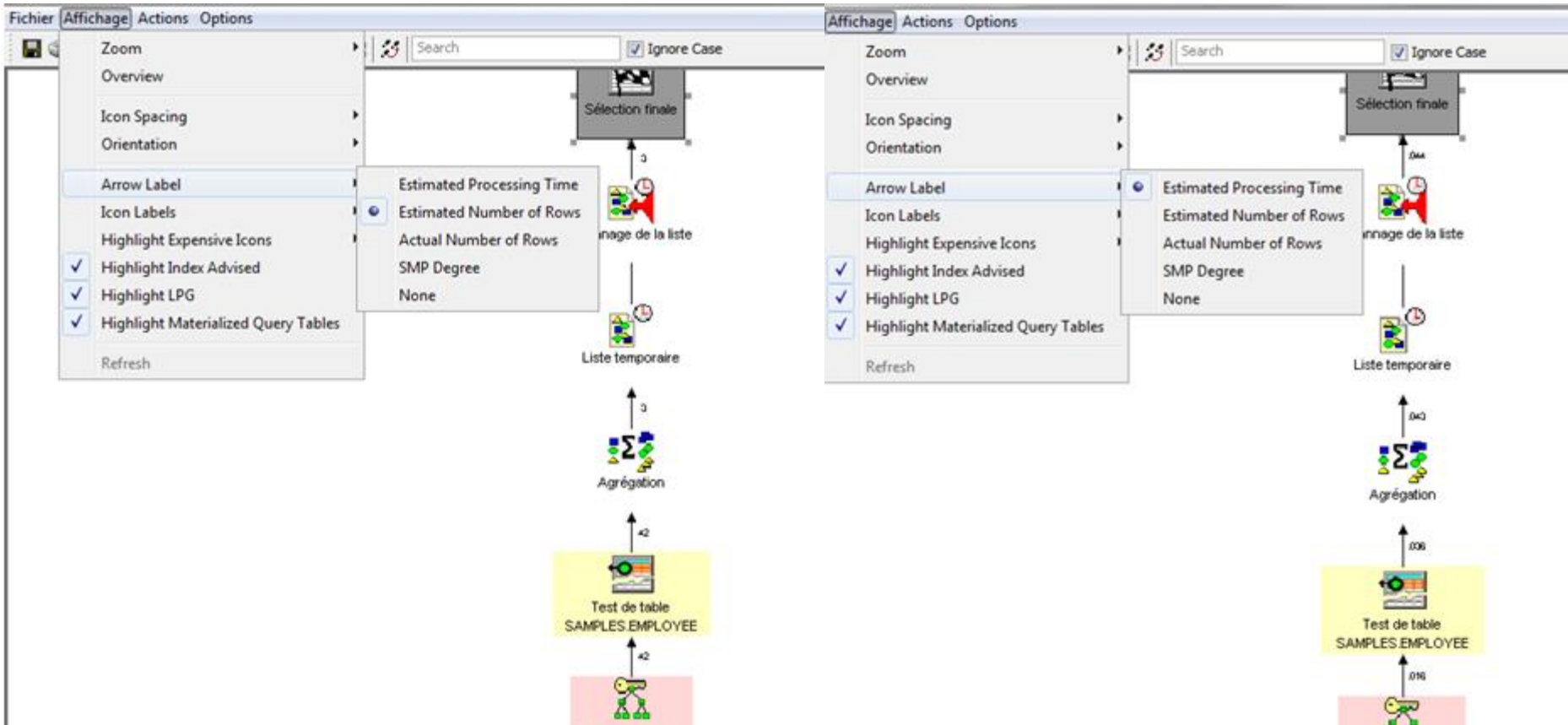
Visual Explain – Option du graphique affiché

- Il est possible de modifier les informations affichées sur le graphique au travers du menu Options :



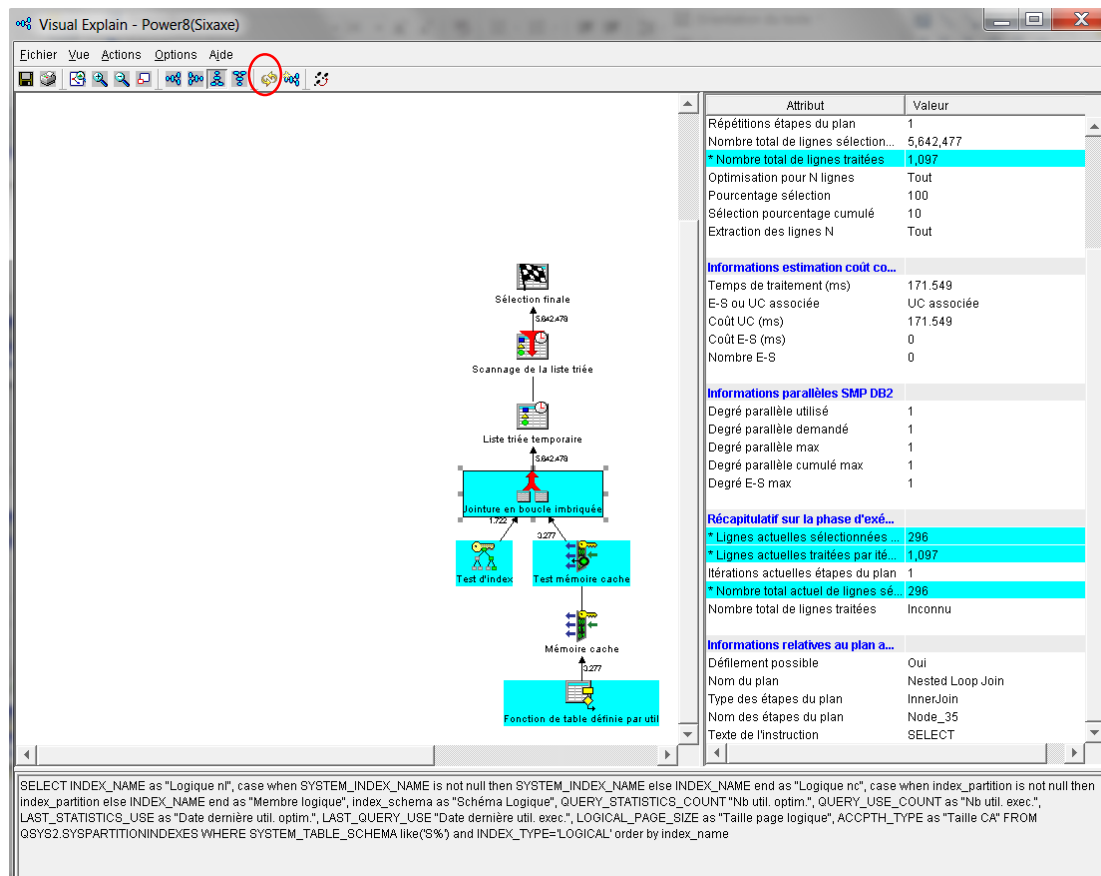
Visual Explain – Choix des informations affichées

- Il est également possible de choisir les informations liées au label des flèches



Visual Explain Suivi temps réel d'une exécution 1/2

- Si l'on lance une requête en mode Explain While Running il est possible d'utiliser une icone afin voir avancer l'exécution.



The screenshot shows the Visual Explain interface for a query. The main window displays a query execution plan with the following steps (from bottom to top):

- Fonction de table définie par util
- Mémoire cache
- Test d'index
- Test mémoire cache
- Jointure en boucle imbriquée
- Liste triée temporaire
- Scannage de la liste triée
- Sélection finale

The statistics table on the right provides the following data:

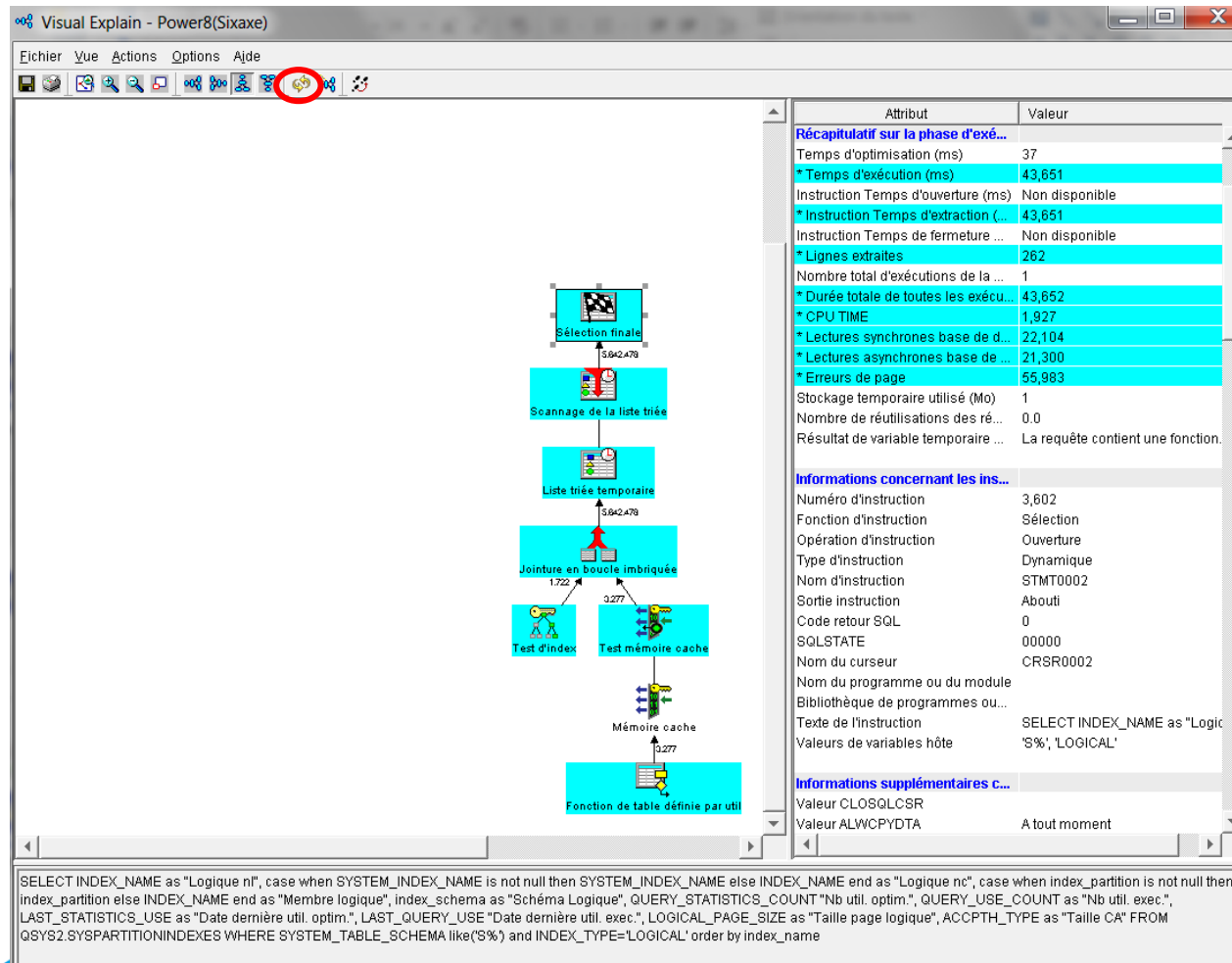
Attribut	Valeur
Répétitions étapes du plan	1
Nombre total de lignes sélection...	5,642,477
* Nombre total de lignes traitées	1,097
Optimisation pour N lignes	Tout
Pourcentage sélection	100
Sélection pourcentage cumulé	10
Extraction des lignes N	Tout
Informations estimation coût co...	
Temps de traitement (ms)	171.549
E-S ou UC associée	UC associée
Coût UC (ms)	171.549
Coût E-S (ms)	0
Nombre E-S	0
Informations parallèles SMP DB2	
Degré parallèle utilisé	1
Degré parallèle demandé	1
Degré parallèle max	1
Degré parallèle cumulé max	1
Degré E-S max	1
Récapitulatif sur la phase d'exé...	
* Lignes actuelles sélectionnées	296
* Lignes actuelles traitées par ité...	1,097
Itérations actuelles étapes du plan	1
* Nombre total actuel de lignes sé...	296
Nombre total de lignes traitées	Inconnu
Informations relatives au plan a...	
Défilement possible	Oui
Nom du plan	Nested Loop Join
Type des étapes du plan	InnerJoin
Nom des étapes du plan	Node_35
Texte de l'instruction	SELECT

The SQL query at the bottom of the window is:

```
SELECT INDEX_NAME as "Logique n", case when SYSTEM_INDEX_NAME is not null then SYSTEM_INDEX_NAME else INDEX_NAME end as "Logique nc", case when index_partition is not null then index_partition else INDEX_NAME end as "Membre logique", index_schema as "Schéma Logique", QUERY_STATISTICS_COUNT "Nb util. optim.", QUERY_USE_COUNT as "Nb util. exec.", LAST_STATISTICS_USE as "Date dernière util. optim.", LAST_QUERY_USE "Date dernière util. exec.", LOGICAL_PAGE_SIZE as "Taille page logique", ACCPTH_TYPE as "Taille CA" FROM QSYS2.SYSPARTITIONINDEXES WHERE SYSTEM_TABLE_SCHEMA like(S%) and INDEX_TYPE='LOGICAL' order by index_name
```

Visual Explain Suivi temps réel d'une exécution 2/2

- Si l'on lance une requête en mode Explain While Running il est possible d'utiliser une icone afin voir avancer l'exécution.



The screenshot shows the Visual Explain interface for a query. The main window displays a query execution plan with the following steps (from bottom to top):

- Fonction de table définie par util (3,277)
- Mémoire cache (3,277)
- Test d'index (1,722) and Test mémoire cache (3,277)
- Jointure en boucle imbriquée (3,042,478)
- Liste triée temporaire (3,042,478)
- Scannage de la liste triée (3,042,478)
- Sélection finale (3,042,478)

The right-hand pane shows a summary table with the following data:

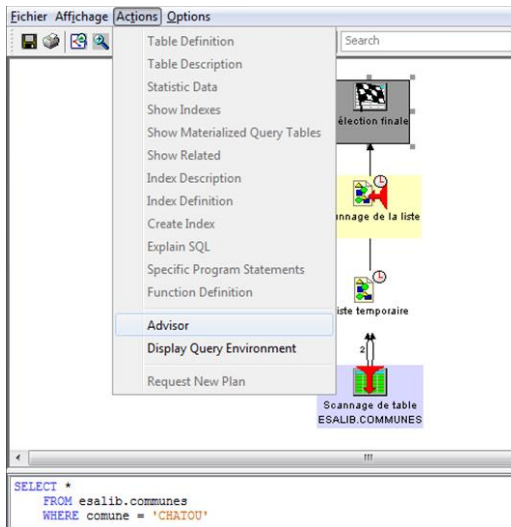
Attribut	Valeur
Récapitulatif sur la phase d'exé...	
Temps d'optimisation (ms)	37
* Temps d'exécution (ms)	43,651
Instruction Temps d'ouverture (ms)	Non disponible
* Instruction Temps d'extraction (...)	43,651
Instruction Temps de fermeture ...	Non disponible
* Lignes extraites	262
Nombre total d'exécutions de la ...	1
* Durée totale de toutes les exécu...	43,652
* CPU TIME	1,927
* Lectures synchrones base de d...	22,104
* Lectures asynchrones base de ...	21,300
* Erreurs de page	55,983
Stockage temporaire utilisé (Mo)	1
Nombre de réutilisations des ré...	0.0
Résultat de variable temporaire ...	La requête contient une fonction.
Informations concernant les ins...	
Numéro d'instruction	3,602
Fonction d'instruction	Sélection
Opération d'instruction	Ouverture
Type d'instruction	Dynamique
Nom d'instruction	STMT0002
Sortie instruction	Abouti
Code retour SQL	0
SQLSTATE	00000
Nom du curseur	CRSR0002
Nom du programme ou du module	
Bibliothèque de programmes ou...	
Texte de l'instruction	SELECT INDEX_NAME as "Logique
Valeurs de variables hôte	'S%', 'LOGICAL'
Informations supplémentaires C...	
Valeur CLOQLCSR	
Valeur ALWCOPYDTA	A tout moment

At the bottom of the window, the SQL query is visible:

```
SELECT INDEX_NAME as "Logique n", case when SYSTEM_INDEX_NAME is not null then SYSTEM_INDEX_NAME else INDEX_NAME end as "Logique nc", case when index_partition is not null then index_partition else INDEX_NAME end as "Membre logique", index_schema as "Schéma Logique", QUERY_STATISTICS_COUNT "Nb util. optim.", QUERY_USE_COUNT as "Nb util. exec.", LAST_STATISTICS_USE as "Date dernière util. optim.", LAST_QUERY_USE "Date dernière util. exec.", LOGICAL_PAGE_SIZE as "Taille page logique", ACCPTH_TYPE as "Taille CA" FROM QSYS2.SYSPARTITIONINDEXES WHERE SYSTEM_TABLE_SCHEMA like('S%') and INDEX_TYPE='LOGICAL' order by index_name
```

Visual Explain – Fonction Advisor

- La fonction Advisor permet d'afficher la liste des index préconisés. Cette information peut être obtenue de deux manières différentes :
 - Soit par le biais du menu Action en sélectionnant la fonction Advisor.



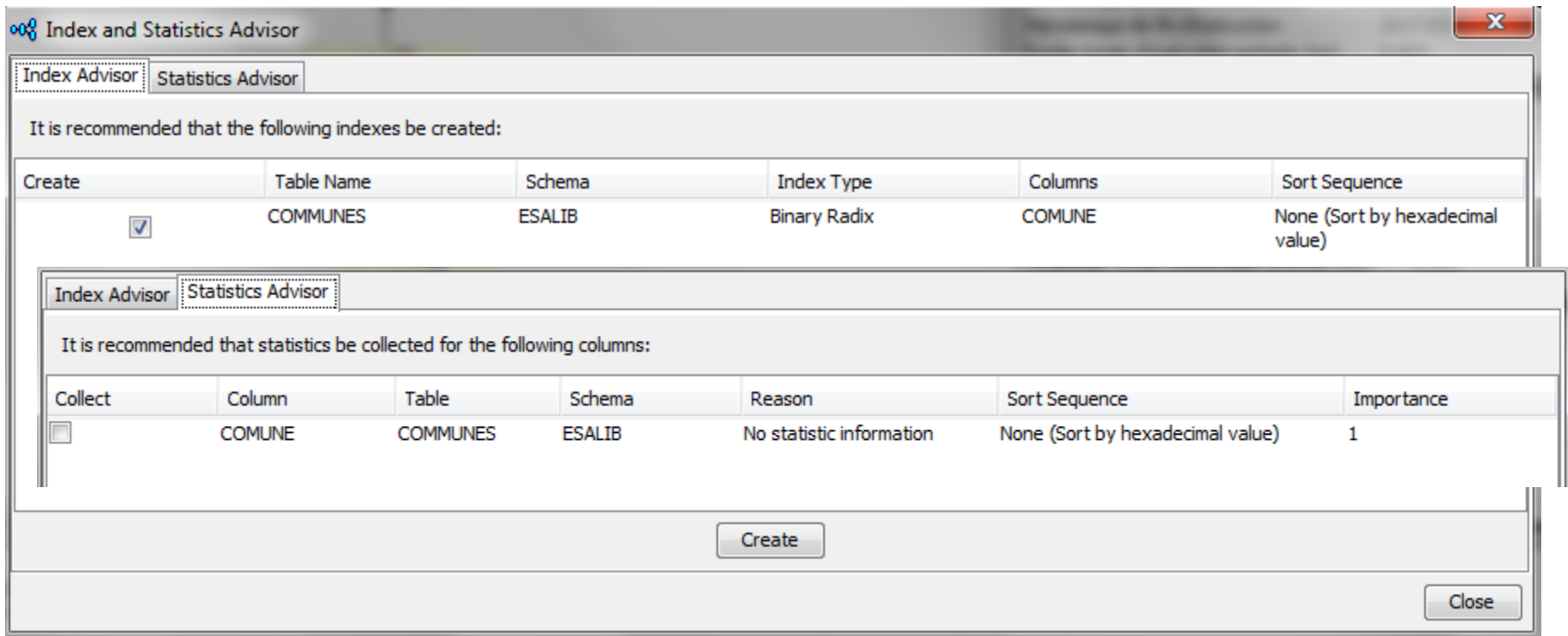
- Soit en sélectionnant l'icone



Visual Explain – Fonction Advisor

- Le fait de lancer, Affichera la boîte de dialogue suivante :

Cette boîte de dialogue permet de créer les index conseillés et de mettre à jour les statistiques si elles le nécessitent.



Index and Statistics Advisor

Index Advisor | Statistics Advisor

It is recommended that the following indexes be created:

Create	Table Name	Schema	Index Type	Columns	Sort Sequence
<input checked="" type="checkbox"/>	COMMUNES	ESALIB	Binary Radix	COMUNE	None (Sort by hexadecimal value)

Index Advisor | Statistics Advisor

It is recommended that statistics be collected for the following columns:

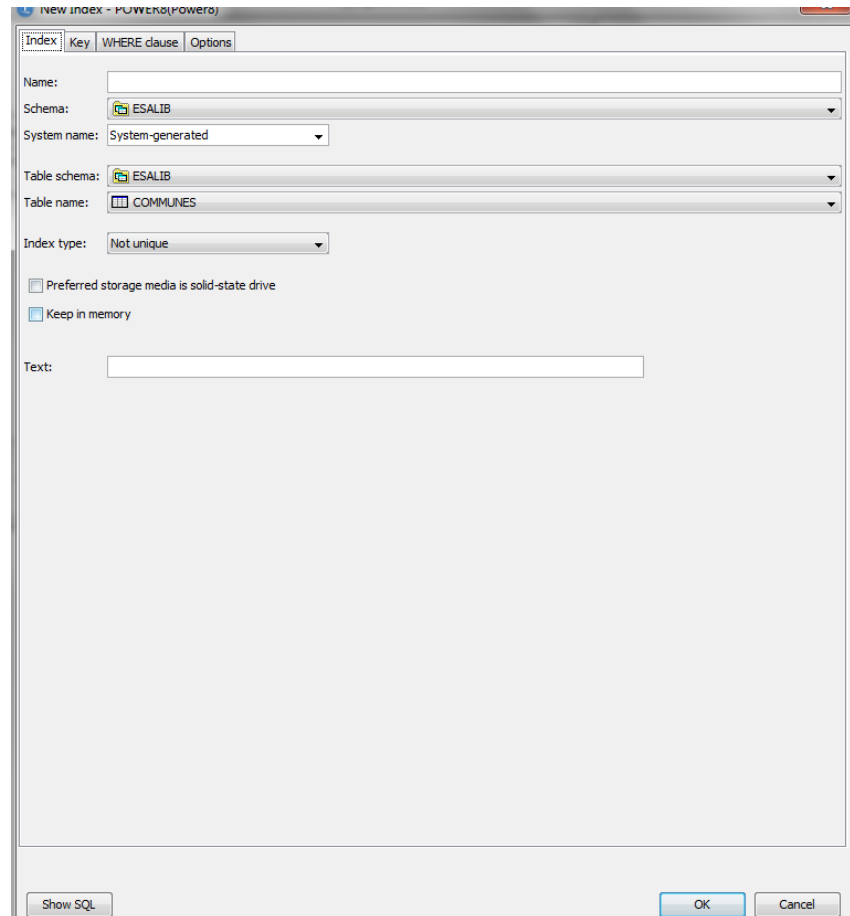
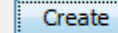
Collect	Column	Table	Schema	Reason	Sort Sequence	Importance
<input type="checkbox"/>	COMUNE	COMMUNES	ESALIB	No statistic information	None (Sort by hexadecimal value)	1

Create

Close

Visual Explain – Création d'index

- La fait de sélectionner le bouton Création
 - Affichera la boîte de dialogue suivante :

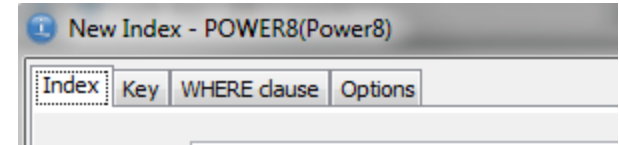
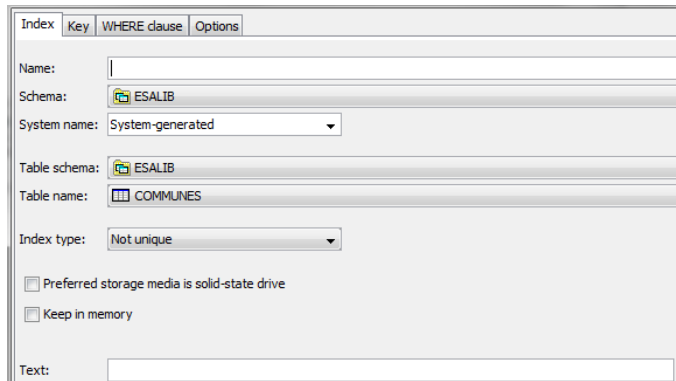


The image shows a 'New index' dialog box in a database management system. The dialog has a title bar 'New index - POWER9 (Power9)'. It contains several fields and options:

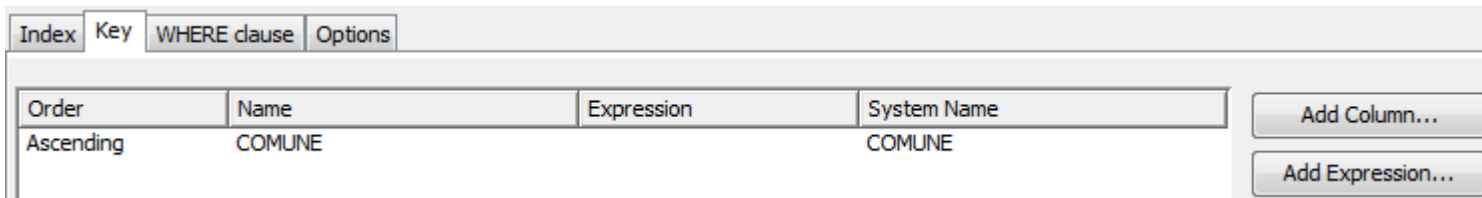
- Index:** A tabbed interface with 'Index' selected. Other tabs are 'Key', 'WHERE clause', and 'Options'.
- Name:** An empty text input field.
- Schema:** A dropdown menu showing 'ESALIB'.
- System name:** A dropdown menu showing 'System-generated'.
- Table schema:** A dropdown menu showing 'ESALIB'.
- Table name:** A dropdown menu showing 'COMMUNES'.
- Index type:** A dropdown menu showing 'Not unique'.
- Options:** Two checkboxes: 'Preferred storage media is solid-state drive' (unchecked) and 'Keep in memory' (checked).
- Text:** An empty text input field.
- Buttons:** 'Show SQL', 'OK', and 'Cancel' at the bottom.

Visual Explain – Création d'index

- Renseigner les onglets :
 - Index (renseigner les zones obligatoires)

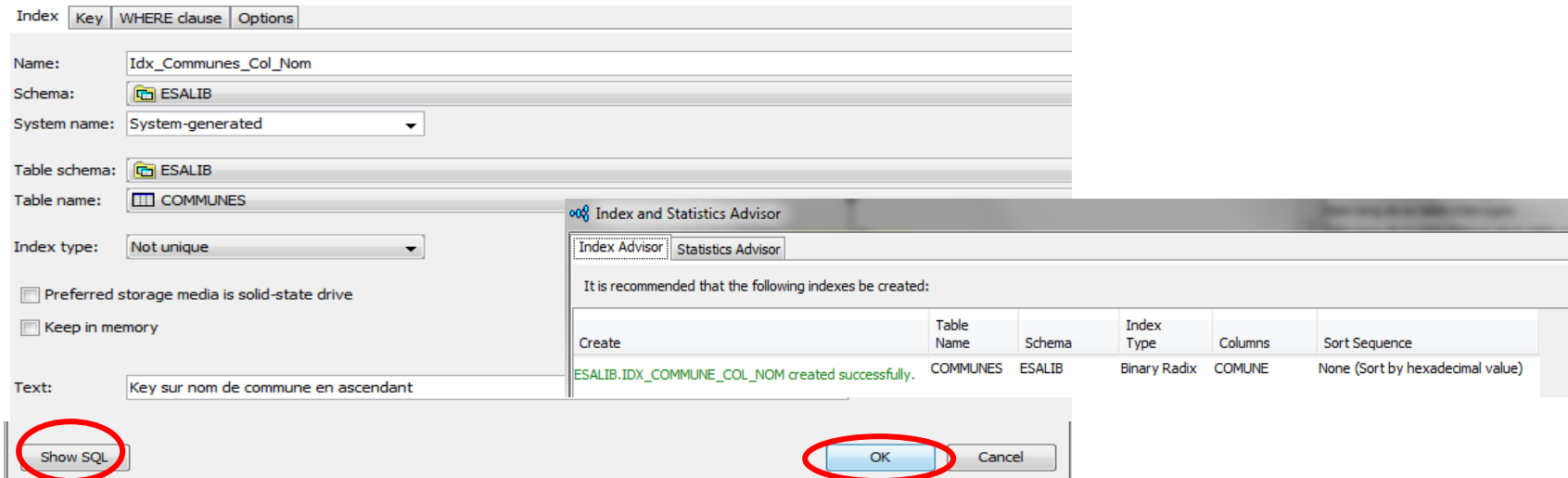
- Clés
 - Les Colonnes sont pré renseignées



Order	Name	Expression	System Name
Ascending	COMUNE		COMUNE

Visual Explain – Création d'index

- Après avoir renseigné les différents onglets
 - Il suffit de valider par le bouton OK



Index | Key | WHERE clause | Options

Name: Idx_Communes_Col_Nom

Schema: ESALIB

System name: System-generated

Table schema: ESALIB

Table name: COMMUNES

Index type: Not unique

Preferred storage media is solid-state drive

Keep in memory

Text: Key sur nom de commune en ascendant

Show SQL

OK Cancel

Index and Statistics Advisor

Index Advisor | Statistics Advisor

It is recommended that the following indexes be created:

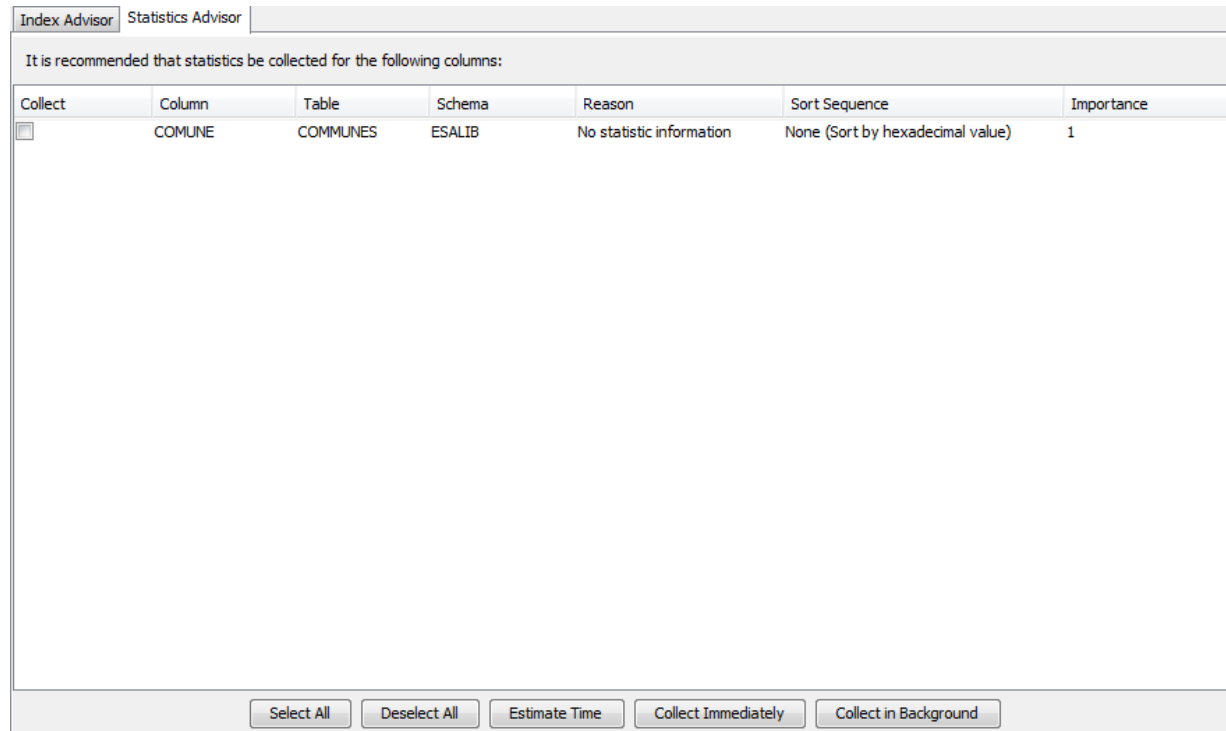
Create	Table Name	Schema	Index Type	Columns	Sort Sequence
ESALIB.IDX_COMMUNE_COL_NOM created successfully.	COMMUNES	ESALIB	Binary Radix	COMMUNE	None (Sort by hexadecimal value)

- Ou de sélectionner SQL et Script Sql s'affichera avec la requête Sql suivante :

```
/*  
Creating ESALIB.IDX_COMMUNE_COL_NOM [Index]  
When creating this index the database connection should have a sort sequence of *HEX.  
*/  
CREATE INDEX ESALIB.IDX_COMMUNE_COL_NOM ON ESALIB.COMMUNES (COMMUNE ASC) UNIT ANY KEEP IN MEMORY NO;  
  
/* Setting label text for ESALIB.IDX_COMMUNE_COL_NOM */  
LABEL ON INDEX ESALIB.IDX_COMMUNE_COL_NOM IS 'Key sur nom de commune en ascendant';
```

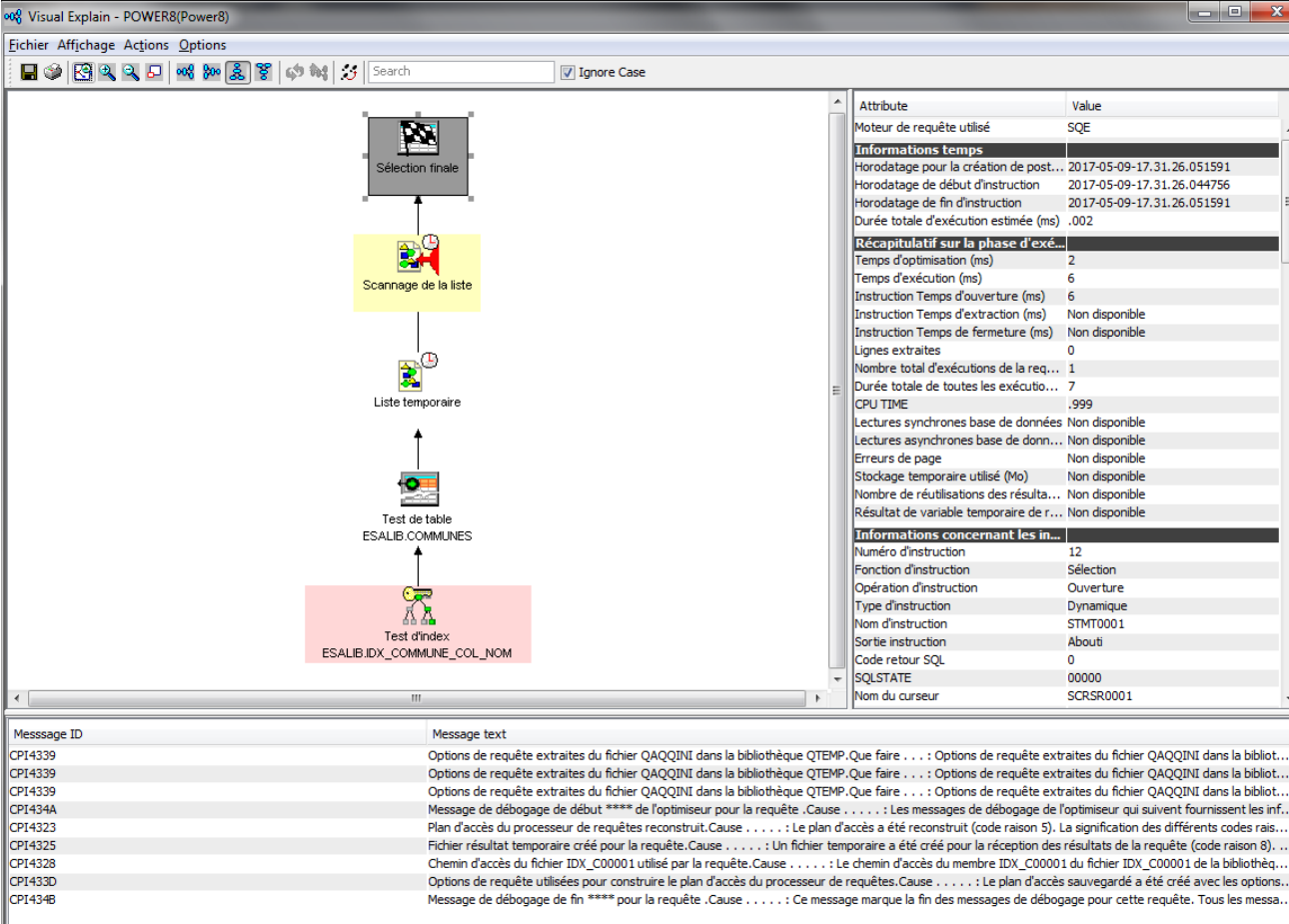
Visual Explain – Création Mise à jour des statistiques

- Après avoir sélectionner l'onglet Statistics Advisor, la boîte dialogue suivante s'affiche :
 - Il suffit de sélectionner les colonnes pour lesquelles les mise à jour sont proposées et valider en sélectionnant le bouton de votre choix



Visual Explain – Advisor

- A l'issue de la création de l'index le plan d'accès devient le suivant :



The screenshot displays the Visual Explain interface for a query. The main window shows a vertical flow of execution steps:

- Sélection finale** (Final Selection)
- Scannage de la liste** (List Scanning)
- Liste temporaire** (Temporary List)
- Test de table ESALIB.COMMUNES** (Table Test)
- Test d'index ESALIB.IDX_COMMUNE_COL_NOM** (Index Test)

On the right side, there is a table of performance statistics:

Attribute	Value
Moteur de requête utilisé	SQE
Informations temps	
Horodatage pour la création de post...	2017-05-09-17.31.26.051591
Horodatage de début d'instruction	2017-05-09-17.31.26.044756
Horodatage de fin d'instruction	2017-05-09-17.31.26.051591
Durée totale d'exécution estimée (ms)	.002
Récapitulatif sur la phase d'exé...	
Temps d'optimisation (ms)	2
Temps d'exécution (ms)	6
Instruction Temps d'ouverture (ms)	6
Instruction Temps d'extraction (ms)	Non disponible
Instruction Temps de fermeture (ms)	Non disponible
Lignes extraites	0
Nombre total d'exécutions de la req...	1
Durée totale de toutes les exécuto...	7
CPU TIME	.999
Lectures synchrones base de données	Non disponible
Lectures asynchrones base de donn...	Non disponible
Erreurs de page	Non disponible
Stockage temporaire utilisé (Mo)	Non disponible
Nombre de réutilisations des résulta...	Non disponible
Résultat de variable temporaire de r...	Non disponible
Informations concernant les in...	
Numéro d'instruction	12
Fonction d'instruction	Sélection
Opération d'instruction	Ouverture
Type d'instruction	Dynamique
Nom d'instruction	STM0001
Sortie instruction	Abouti
Code retour SQL	0
SQLSTATE	00000
Nom du curseur	SCRSR0001

At the bottom, a message log shows various system messages, including:

- Options de requête extraites du fichier QAQQINI dans la bibliothèque QTEMP.
- Plan d'accès du processeur de requêtes reconstruit.
- Fichier résultat temporaire créé pour la requête.
- Chemin d'accès du fichier IDX_C00001 utilisé par la requête.
- Options de requête utilisées pour construire le plan d'accès du processeur de requêtes.
- Message de débogage de fin **** pour la requête.



Visual Explain – les Icones



- The *Final Select* icon displays the original text and summary information about how the query was implemented.



- The *Table* icon indicates that a table was accessed. See the Select icon for more details.



- The *Table Scan* icon indicates that all rows in the table were paged in, and selection criteria was applied against each row. Only those rows that meet the selection criteria were retrieved. To obtain the result in a particular sequence, you must specify the ORDER BY clause.



- The *Table Probe* icon indicates that data from the table needed to be processed and selected for this query. The table is probed using a key derived from the ordinal number or relative record number associated with each row in the table. The ordinal number is used to calculate what pages of data need to be retrieved and brought into main memory to satisfy this probe request. The ordinal number used for the probe operation was provided by another data access method previously processed for this query.



- The *Index* icon indicates that an index object was used to process this query.



Visual Explain – les Icones



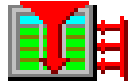
- The *Index Scan* icon indicates that the entire index will be scanned, which causes all of the entries in the index to be paged into main memory to be processed. Any selection criteria, whose predicates match the key columns of the index, can then be applied against the index entries. Only those key entries that match the specified key selection will be used to select rows from the corresponding table data.



- The *Index Probe* icon indicates that the selection criteria whose predicates matched the leading key columns of the index were used to probe directly into the index. The probe selection minimizes the number of key entries that must be processed and paged into main memory. Additional key selection can be applied against the non-leading key columns of the index to further reduce the number of selected key entries. Only those key entries that match the specified probe and key selection will be used to select rows from the corresponding table data.



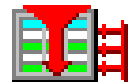
Visual Explain – les Icones



- The *Parallel Table Scan* icon indicates that a table scan access method was used and multiple tasks were used to fill the rows in parallel. The table was partitioned, and each task was given a portion of the table to use.



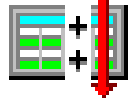
- The *Skip Sequential Table Scan* icon indicates that a bitmap was used to determine which rows would be selected. No CPU processing was done on non-selected rows, and I/O was minimized by bringing in only those pages that contained rows to be selected. This icon usually is related to the Dynamic Bitmap or Bitmap Merge icons.



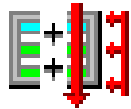
- The *Skip Sequential Parallel Scan* icon indicates that a skip sequential table scan access method was used and multiple tasks were used to fill the rows in parallel. The table was partitioned, and each task was given a portion of the table to use.



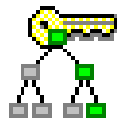
Visual Explain – les Icones



- The *Derived Column Selection* icon indicates that a column in the row selected had to be mapped or derived before selection criteria could be applied against the row. Derived column selection is the slowest selection method.



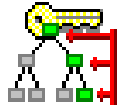
- The *Parallel Derived Column Selection* icon indicates that derived field selection was performed, and the processing was accomplished using multiple tasks. The table was partitioned, and each task was given a portion of the table to use.



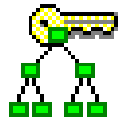
- The *Index Key Positioning* icon indicates that only entries of the index that match a specified range of key values were “paged in”. The range of key values was determined by the selection criteria whose predicates matched the key columns of the index. Only selected key entries were used to select rows from the corresponding table data.



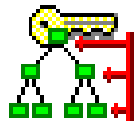
Visual Explain – les Icones



- The *Parallel Index Key Positioning* icon indicates that multiple tasks were used to perform the key positioning in parallel. The range of key values was determined by the selection criteria, whose predicates matched the key columns of the index. Only selected key entries were used to select rows from the corresponding table data.



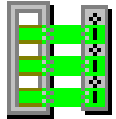
- The *Index Key Selection* icon indicates that all entries of the index were paged in. Any selection criteria, whose predicates match the key columns of the index, was applied against the index entries. Only selected key entries were used to select rows from the table data.



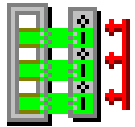
- The *Parallel Index Key Selection* icon indicates that multiple tasks were used to perform key selection in parallel. The table was partitioned, and each task was given a portion of the table to use.



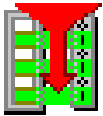
Visual Explain – les Icones



- The *Encoded-vector Index* icon indicates that access was provided to a database file by assigning codes to distinct key values, and then representing these values in an array (vector). Because of their compact size and relative simplicity, encoded-vector indexes (EVIs) provide for faster scans.



- The *Parallel Encoded-vector Index* icon indicates that multiple tasks were used to perform the EVI selection in parallel. This allows for faster scans that can be more easily processed in parallel.



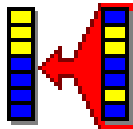
- The *Encoded-vector Index Scan* icon indicates that the entire EVI will be scanned causing all of the distinct values represented in the index to be processed. Any selection criteria, whose predicates match the key columns of the EVI, can then be applied against the distinct values represented in the index. Only those distinct values that match the specified key selection will then be used to process the vector and generate either a temporary row number list or temporary row number bitmap.



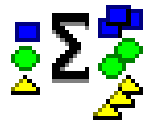
Visual Explain – les Icones



- The *Encoded-vector Index Probe* icon indicates that the selection criteria whose predicates matched the leading key columns of the EVI were used to probe directly into the distinct values represented in the index. Only those distinct values that match the specified probe selection will then be used to process the vector and generate either a temporary row number list or temporary row number bitmap, can be use with include and Group By.



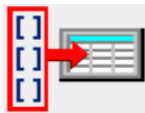
- The *Sort Sequence* icon indicates that selected rows were sorted using a sort algorithm.



- The *Grouping* icon indicates that selected rows were grouped or summarized. Therefore, duplicate rows within a group were eliminated.



- The *Nested Loop Join* icon indicates that queried tables were joined together using a nested loop join implementation. Values from the primary file were joined to the secondary file by using an index whose key columns matched the specified join columns. This icon is usually after the method icons used on the underlying tables (that is, index scan-key selection and index scan-key positioning).



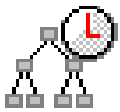
- The *Unnest Array* icon indicates that the Function UNNEST use an Array in the procedure.



Visual Explain – les Icones



- The *Hash Join* icon indicates that a temporary hash table was created. The tables queried were joined together using a hash join implementation where a hash table was created for each secondary table. Therefore, matching values were hashed to the same hash table entry.



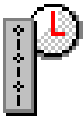
- The *Temporary Index* icon indicates that a temporary index was created, because the query either requires an index and one does not exist, or the creation of an index will improve performance of the query.



- The *Temporary Hash Table* icon indicates that a temporary hash table was created to perform hash processing.



- The *Temporary Table* icon indicates that a temporary table was required to either contain the intermediate results of the query, or the queried table could not be queried as it currently exists and a temporary table was created to replace it.



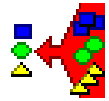
- The *Dynamic Bitmap* icon indicates that a bitmap was dynamically generated from an existing index. It was then used to determine which rows were to be retrieved from the table. To improve performance, dynamic bitmaps can be used in conjunction with a table scan access method for skip sequential or with either the index key position or key selection.



- The *Bitmap Merge* icon indicates that multiple bitmaps were merged or combined to form a final bitmap. The merging of the bitmaps simulates boolean logic (AND/OR selection).



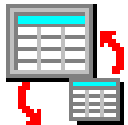
Visual Explain – les Icones



- The *DISTINCT* icon indicates that duplicate rows in the result were prevented. You can specify that you do not want any duplicates by using the *DISTINCT* keyword, followed by the selected column names.



- The *UNION Merge* icon indicates that the results of multiple subselects were merged or combined into a single result.



- The *Subquery Merge* icon indicates that the nested *SELECT* was processed for each row (*WHERE* clause) or group of rows (*HAVING* clause) selected in the outer level *SELECT*. This is also referred to as a *correlated subquery*.



- The *Hash Table Scan* icon indicates that the entire temporary hash table will be scanned and all of the entries contained with the hash table will be processed. A hash table scan is generally considered when optimizer is considering a plan that requires the data values to be collated together but the sequence of the data is not required. The use of a hash table scan allows the optimizer to generate a plan that can take advantage of any non-join selection while creating the temporary hash table.



Visual Explain – les Icones



- The *Hash Table Probe* icon indicates that the selection criteria that match the key columns used to create the temporary hash table will be probed to find all of the matching values stored within the hash table. A hash table probe is generally considered when determining the implementation for a secondary table of a join. The use of a hash table probe allows the optimizer to generate a plan that can take advantage of any non-join selection while creating the temporary hash table. An additional benefit of using a hash table probe is that the data structure of the temporary hash table usually causes the table data to remain resident within main memory after creation, reducing paging on the subsequent probe operation.



- The *Temporary Distinct Hash Table* icon indicates that a temporary distinct hash table was created in order to perform hash processing. A distinct hash table is a data structure that is identical to the temporary hash table, except all duplicate data is compressed out of the temporary being created. The resulting hash table can then be used to perform distinct or aggregate operations for the query



Visual Explain – les Icones



- The *Distinct Hash Table Scan* icon indicates that the entire temporary distinct hash table will be scanned and all of the entries contained with the hash table will be processed. A distinct hash table scan is generally considered when optimizer is considering a plan that requires the data values to be collated together and all duplicate removed but the sequence of the data is not required. The use of a distinct hash table scan will allow the optimizer to generate a plan that can take advantage of any non-join selection while creating the temporary distinct hash table. An additional benefit of using a distinct hash table scan is that the data structure of the temporary distinct hash table usually causes the table data within the distinct hash table to remain resident within main memory after creation, reducing paging on the subsequent scan operations.



- The *Distinct Hash Table Probe* icon indicates that the selection criteria that match the key columns used to create the temporary distinct hash table will be probed to find all of the matching values stored within the hash table. The use of a distinct hash table probe allows the optimizer to generate a plan that can take advantage of any non-join selection while creating the temporary distinct hash table. An additional benefit of using a distinct hash table probe is that the data structure of the temporary distinct hash table usually causes the table data to remain resident within main memory after creation, reducing paging on the subsequent probe operation.



Visual Explain – les Icones

- The *Temporary Sorted List* icon indicates that a temporary sorted list was created in order to perform a sequencing operation. A sorted list is a data structure where the table data is collated and sorted based upon the value of a column or columns referred to as the sort key. The sorted list can then be used to return the data in a specified sequence or to perform probe operations using the sort key to quickly retrieve all of the table data that matches a particular sort key.



- The *Sorted List Scan* icon indicates that the entire temporary sorted list will be scanned and all of the entries contained with the sorted list will be processed. A sorted list scan is generally considered when optimizer is considering a plan that requires the data values to be sequenced based upon the sort key of the sorted list. The use of a sorted list scan will allow the optimizer to generate a plan that can take advantage of any non-join selection while creating the temporary sorted list. An additional benefit of using a sorted list scan is that the data structure of the temporary sorted list usually causes the table data within the sorted list to remain resident within main memory after creation, reducing paging on the subsequent scan operations.



- The *Sorted List Probe* icon indicates that the selection criteria that match the key columns used to create the temporary sorted list will be probed to find all of the matching values stored within the sorted list. A sorted list probe is generally considered when determining the implementation for a secondary table of a join when either the join condition uses an operator other than equal or a temporary hash table is not allowed in this query environment. The use of a sorted list probe allows the optimizer to generate a plan that can take advantage of any non-join selection while creating the temporary sorted list. An additional benefit of using a sorted list probe is that the data structure of the temporary sorted list usually causes the table data to remain resident within main memory after creation, reducing paging on the subsequent probe operation.

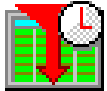




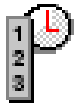
Visual Explain – les Icones



- The *Temporary List* icon indicates that a temporary list was created. The temporary list was required to either contain the intermediate results of the query, or the queried table could not be queried as it currently exists and a temporary list was created to replace it. The list is an unsorted data structure with no key. The data contained within the list can only be retrieved by a scan operation.



- The *List Scan* icon indicates that the entire temporary list will be scanned and all of the entries will be processed.



- The *Temporary Row Number List* icon indicates that a temporary row number list was created in order to help process any selection criteria. A row number list is a data structure used to represent the selected rows from a table that matches any specified selection criteria. Since the selected rows are represented by a sorted list of row numbers, multiple lists can be merged and combined to allow for complex selection and processing to be performed without having any paging occur against the table itself.



- The *Row Number List Scan* icon indicates that the entire row number list will be scanned and all of the entries will be processed. Scanning a row number list can provide large amounts of savings for the table data associated with the temporary row number list. Since the data structure of the temporary row number list guarantees that the row numbers are sorted, it closely mirrors the row number layout of the table data, ensuring that the paging on the table will never revisit the same page of data twice.



Visual Explain – les Icones



- The *Row Number List Probe* icon indicates that a row number list was used to verify that a row from a previous operation in the query matches the selection criteria used to create the temporary row number list. The use of a row number list probe allows the optimizer to generate a plan that can process the rows in the table in any manner regardless of any specified selection criteria. As the rows are processed, the ordinal number from the row is used to probe into the row number list to determine if that row matches the selection criteria. This is generally found when an index is used to satisfy the ORDER BY from a query and a separate viable index exists to process the selection criteria.



- The *Bitmap Scan* icon indicates that the entire bitmap will be scanned and all of the entries that represent selected rows will be processed. Scanning a bitmap can provide large amounts of savings for the table data associated with the temporary bitmap. Since the data structure of the temporary bitmap mirrors the row number layout of the table data, the bitmap can be used to efficiently schedule paging of the table for all selected rows.



- The *Bitmap Probe* icon indicates that a bitmap was used to verify that a row from a previous operation in the query matches the selection criteria used to create the temporary bitmap. The use of a bitmap probe allows the optimizer to generate a plan that can process the rows in the table in any manner regardless of any specified selection criteria. As the rows are processed, the ordinal number from the row is used to probe into the bitmap to determine if that row matches the selection criteria. This is generally found when an index is used to satisfy the ORDER BY from a query and a separate viable index exists to process the selection criteria.



- The *Index Scan* icon indicates that the entire temporary index will be scanned causing all of the entries in the index to be paged into main memory to be processed. Any selection criteria, whose predicates match the key columns of the index, can then be applied against the index entries. Only those key entries that match the specified key selection are used to select rows from the corresponding table data.



- The *Index Probe* icon indicates that the selection criteria whose predicates matched the leading key columns of the index were used to probe directly into the temporary index. The probe selection minimizes the number of key entries that must be processed and paged into main memory. Additional key selection can be applied against the non-leading key columns of the temporary index to further reduce the number of selected key entries. Only those key entries that matched the specified probe and key selection are used to select rows from the corresponding table data.



Visual Explain – les Icones



- The *Temporary Correlated Hash Table* icon indicates that a temporary correlated hash table was created in order to perform hash processing. A hash table is a data structure where the table data is collated based upon the value of a column or columns referred to as the *hash key*. The hash table can then be used to perform probe operation using the hash key to quickly retrieve all of the table data that matches a particular hash value. Because this is a correlated hash table, the hash table needs to be rebuilt prior to any scan or probe operations being performed.



- The *Correlated Hash Table Scan* icon indicates that the entire temporary hash table will be scanned and all of the entries contained with the hash table will be processed. A correlated hash table scan is generally considered when optimizer is considering a plan that requires the data values to be collated together but the sequence of the data is not required. In addition, the some of the values used to create the correlated hash table can change from one scan to another. The use of a correlated hash table scan will allow the optimizer to generate a plan that can take advantage of any non-join selection while creating the temporary correlated hash table. An additional benefit of using a correlated hash table scan is that the data structure of the temporary correlated hash table usually causes the table data within the hash table to remain resident within main memory after creation thus reducing paging on the subsequent scan operations.



- The *Correlated Hash Table Probe* icon indicates that the selection criteria that match the key columns used to create the temporary correlated hash table will be probed to find all of the matching values stored within the hash table. A correlated hash table probe is generally considered when determining the implementation for a secondary table of a join. The use of a hash table probe allows the optimizer to generate a plan that can take advantage of any non-join selection while creating the temporary correlated hash table. An additional benefit of using a correlated hash table probe is that the data structure of the temporary correlated hash table usually causes the table data to remain resident within main memory after creation, reducing paging on the subsequent probe operation.



- The *Temporary Correlated List* icon indicates that a temporary correlated list was created. The temporary correlated list was required to either contain the intermediate results of the query, or the queried table could not be queried as it currently exists and a temporary correlated list was created to replace it. The list is an unsorted data structure with no key that must be rebuilt prior to any scan operation being performed.



Visual Explain – les Icones



- The *Correlated List Scan* icon indicates that the entire temporary list will be scanned and all of the entries will be processed.



- The *Temporary Buffer* icon indicates that a temporary buffer was created to store the intermediate rows of an operation. The temporary buffer is generally considered at a serialization point within a query to help facilitate operations such as parallelism. The buffer is an unsorted data structure but it differs from other temporary data structures in that the buffer does not have to be fully populated in order allow its results to be processed.



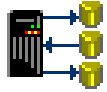
- The *Buffer Scan* icon indicates that the entire temporary buffer will be scanned and all of the entries will be processed.



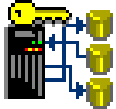
- The *Table Random Pre-Fetch* icon indicates that the pages required for the table probe operation will be requested synchronously in the background prior to the actual table probe operation being performed. The system attempts to manage the paging for the table probe to maintain that all of the pages of data necessary to perform the table probe operation stay resident within main memory until they are processed. The amount of pre-fetch paging that is performed by this data access method is dynamically controlled by the system based upon memory consumption and the rate at which rows continue to be processed.



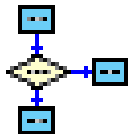
Visual Explain – les Icones



- The *Table Clustered Pre-Fetch* icon indicates that the pages required for the table probe operation will be requested synchronously in the background prior to the actual table probe operation being performed. The system attempts to manage the paging for the table probe to maintain that all of the pages of data necessary to perform the table probe operation stay resident within main memory until they are processed. The amount of pre-fetch paging that is performed by this data access method is dynamically controlled by the system based upon memory consumption and the rate at which rows continue to be processed.



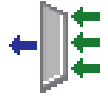
- The *Index Random Pre-Fetch* icon indicates that the pages required for the index probe operation will be requested synchronously in the background prior to the actual index probe operation being performed. The system attempts to manage the paging for the index probe to maintain that all of the pages of data necessary to perform the index probe operation stay resident within main memory until they are processed. The amount of pre-fetch paging that is performed by this data access method is dynamically controlled by the system based upon memory consumption and the rate at which rows continue to be processed.



- The *Logic* icon indicates that the query needed to perform an operation or test against the data in order to generate the selected rows.



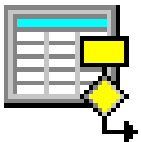
Visual Explain – les Icones



- The *Fetch N Rows* icon indicates that a limit was placed upon the number of selected rows. The fetch n rows access method can either be used to implement a user specified limit on the selected rows or it can be combined with other access methods by the optimizer to satisfy complex implementation plans.



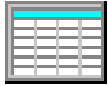
- The *Lock Row for Update* icon indicates that an update lock was acquired for the associated table data in order to perform an update or delete operation. To minimize contention between queries, the optimizer attempts to place the lock row for update operation such that the lock is not acquired and held for a long duration.



The *User-defined table* function icon indicates that a user-defined function that returns a table was used. A table function can be referenced in an SQL FROM clause in the same way that a table or view can be referenced.



Visual Explain – les Icones



- The *Select* icon indicates a point in the query where multiple results are brought together into a single result set. For example, if a query is the union of two different select statements, at the point before the union occurs, the Select icon indicates the points where the select statements finished and the union is about to occur. This icon also represents the default information for an operation that is unknown or not defined elsewhere with Visual Explain. It can help to represent tables or insert, update and delete operations for a query. The summary information for this icon contains any available information to help describe the operation being performed and what the icon represents.



- The *Incomplete Information* icon indicates that a query could not be displayed due to incomplete information.

Visual Explain - Conclusion

- **Usez et abusez** de Visual Explain afin de comprendre vos requêtes.
- Analysez et suivez les recommandations d'Index
- Réfléchissez à la mise en œuvre d'index avec des clés dérivées
- Essayez de mettre en œuvre des index parfaits et mieux encore « Only Access »
- Sachez bien que les index d'aujourd'hui, ne seront pas forcément les index de demain....

Enjoy Visuel Explain

ithankyou

