# DataFrogs

Les structures de données

@conseilit

Microsoft® SQL Server®

~ since 1997 : SQL 6.5 / WinNT4

/conseilit

@conseilit

/in/christophelaporte

- Audit / Conseil
  - Infrastructure / Architecture
  - Virtualisation / Cloud
  - Haute disponibilité
  - Performance / Optimisation
  - Dépannage / Migrations
- Formations
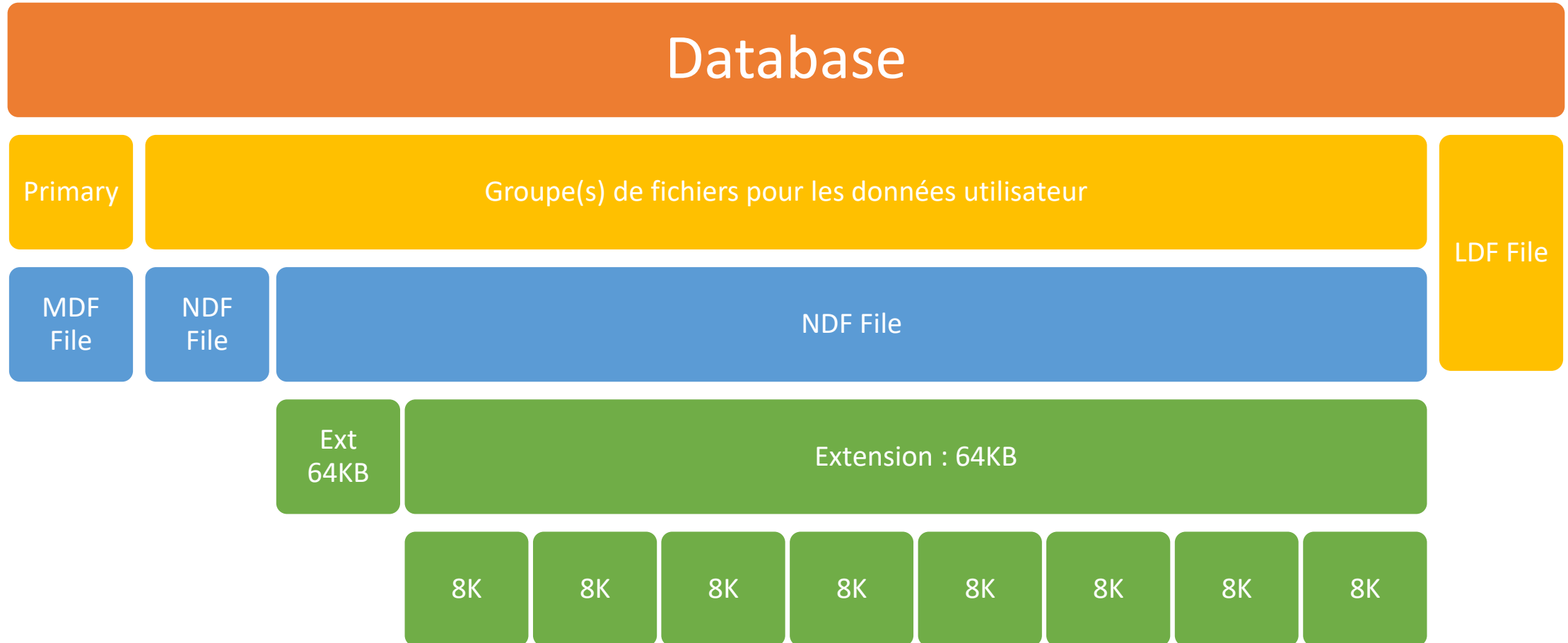- DBA-as-a-Service

conseilit@outlook.com

# Les structures de données

Row Store

# Anatomie d'une base de données

# Deux types d'extensions



MIXED_PAGE_ALLOCATION

TF 1118

**Mixed extent**

| Table2 | Index1 | Index2 | Table2 | Table3 | Index3 | Table2 | Table3 |

**Uniform extent**

| Table1 | Table1 | Table1 | Table1 | Table1 | Table1 | Table1 | Table1 |

# 1ère Extension

| Page 0 | Page 1 | Page 2 | Page 3 | Page 4 | Page 5 | Page 6 | Page 7 |
|--------|--------|--------|--------|--------|--------|--------|--------|
| File Header | PFS | GAM | SGAM | Unused | unused | DCM | BCM |

Track free space
1 / 8088 pages

Manage Extent Allocation
1 / 64 000 extents  (~4GB)

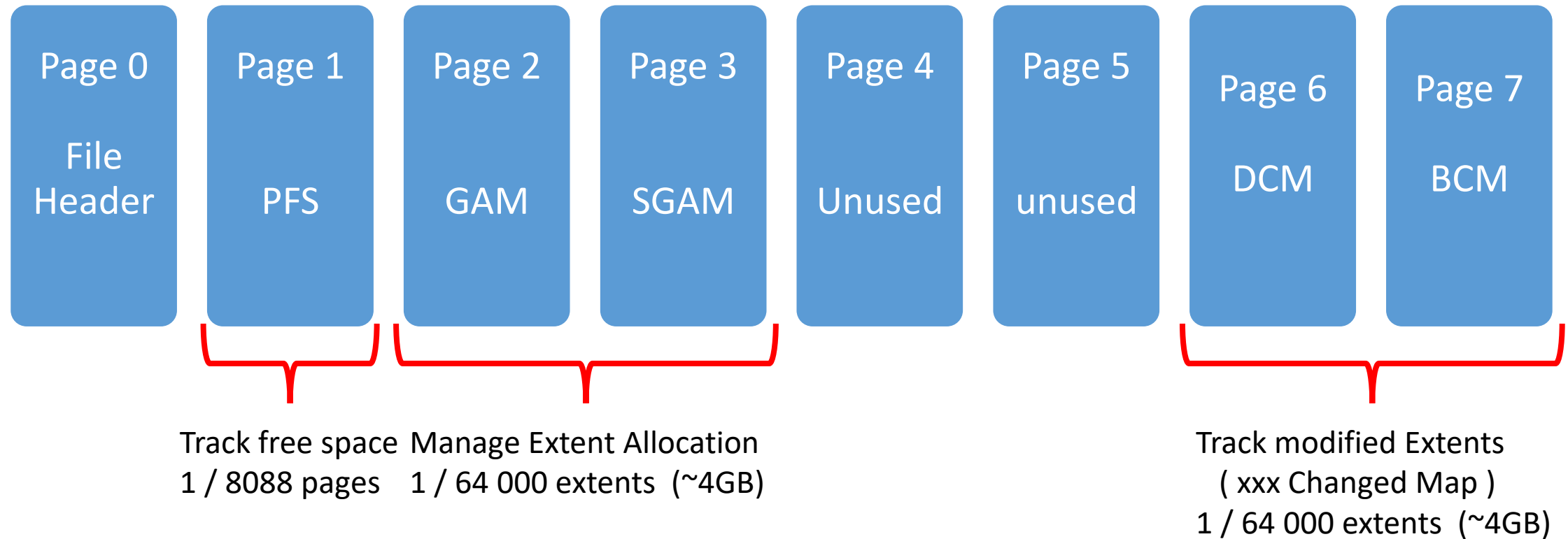Track modified Extents
( xxx Changed Map )
1 / 64 000 extents  (~4GB)

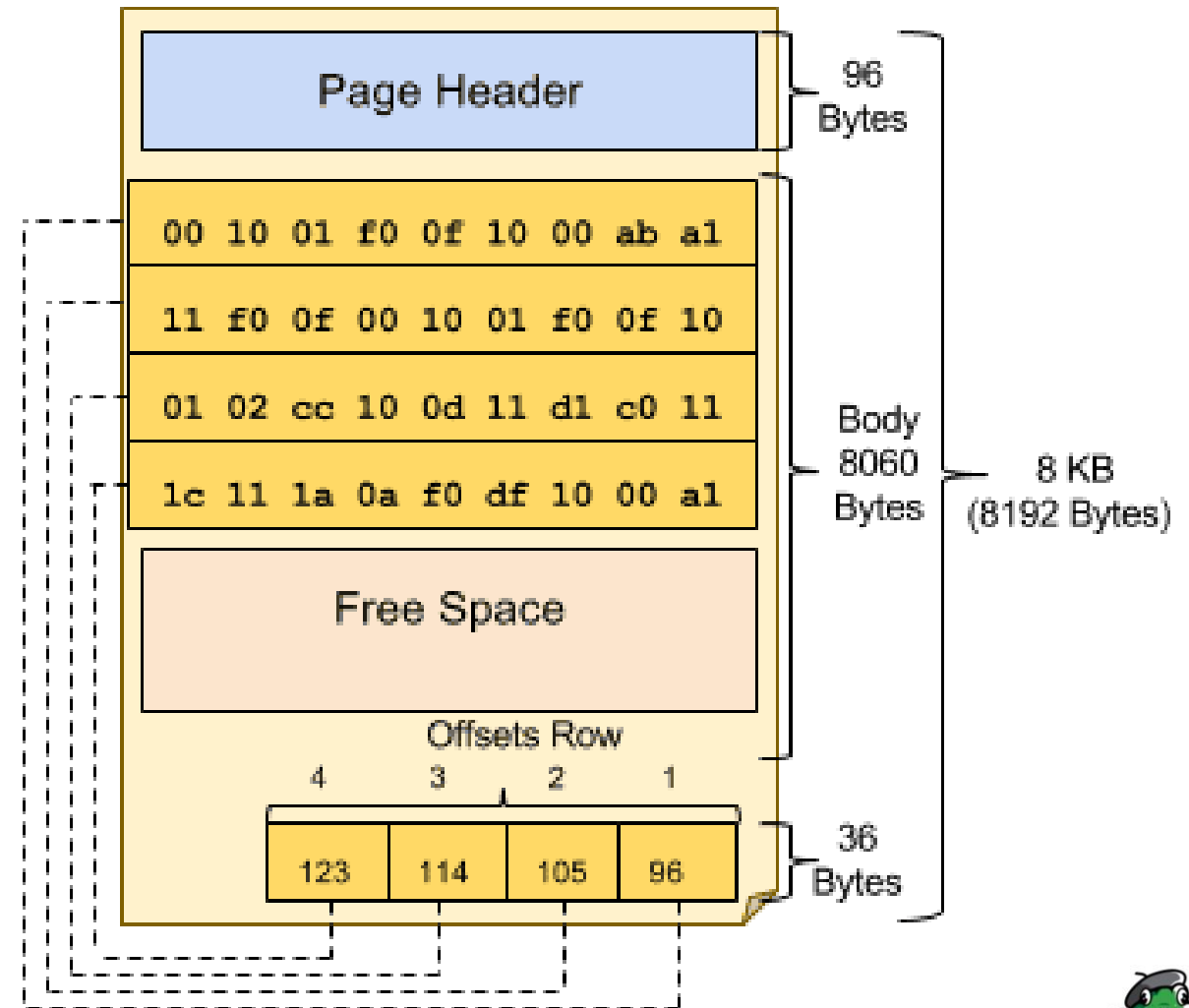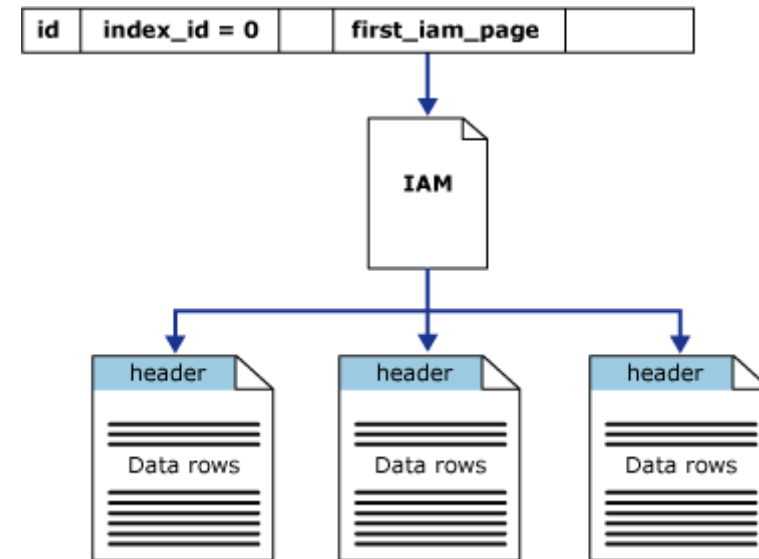DATAFROGS

# Pages de données

- Chaque page
  - Relative à un objet

- Un enregistrement
  - 8060 o maximum
  - Typages de données
  - LOB InRow / OffRow

- Une table
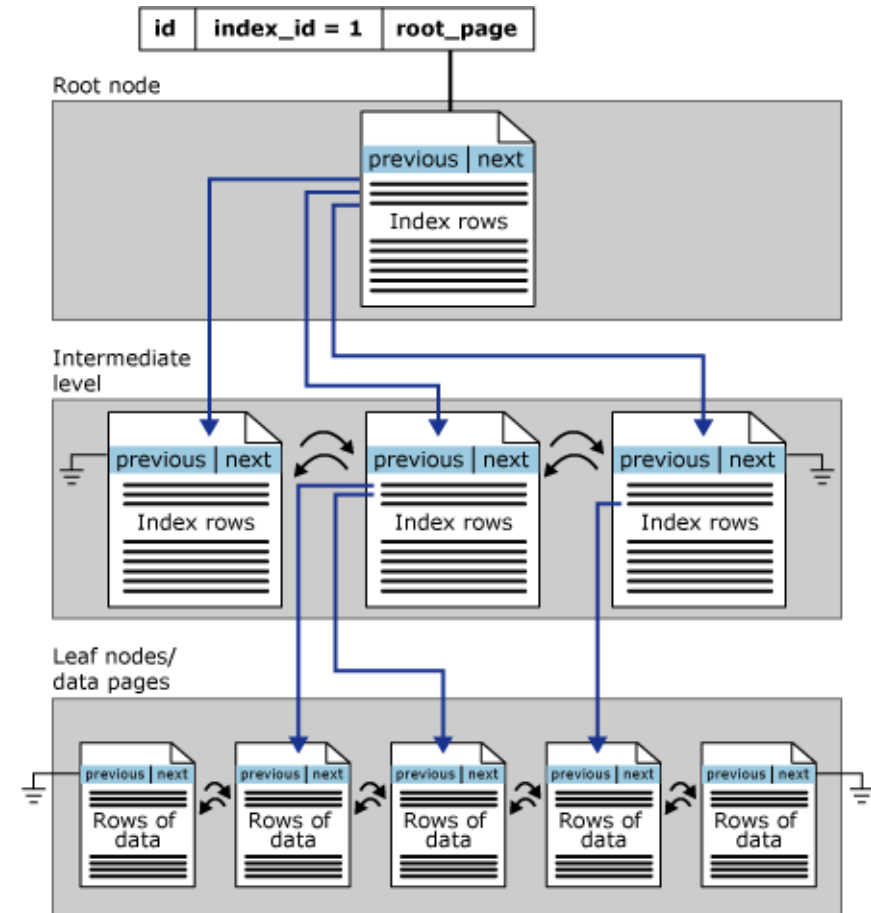  - Liste chaînée de Data Pages

# Heap (Row Store)

- Enregistrements
  - Non ordonnés
- Avantages
  - Insert rapide
- Inconvénients
  - Forwarded records
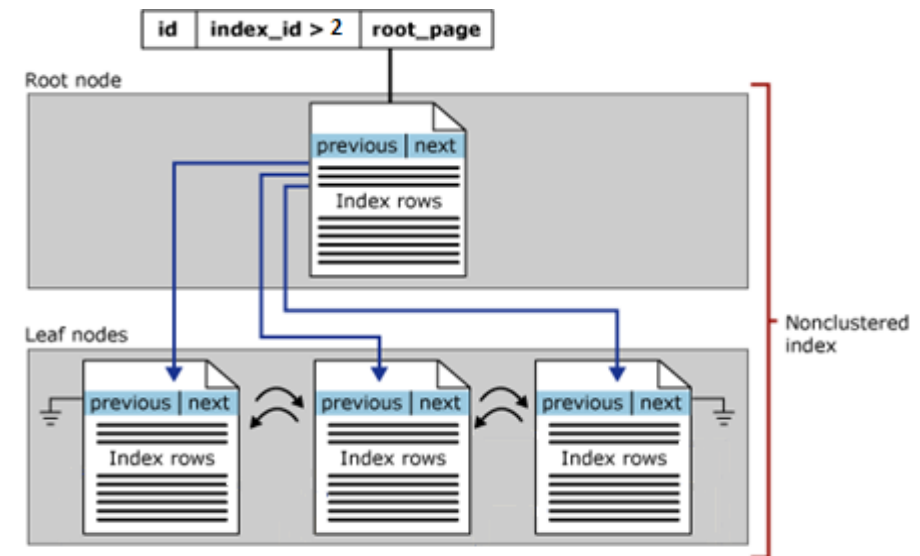  - Espace perdu (Del / Upd)

# Index Clustered (Row Store)

- Enregistrements
  - Ordonnés (inter pages)
  - Record size : 8060o
  - Key Size : 900o

- Avantages
  - Pages feuille : les données
  - Range Scan

- Inconvénients
  - Page split
  - Hot Page
  - Un seul  (trouver le bon !)

# Index NonClustered (Row Store)

- Enregistrements
  - Ordonnés (inter pages)
  - Key Size : 1700o
- Options
  - Index avec colonne(s) incluse(s)
  - Index filtré
  - Index unique
- Avantages
  - Accélère la recherche de données
- Inconvénients
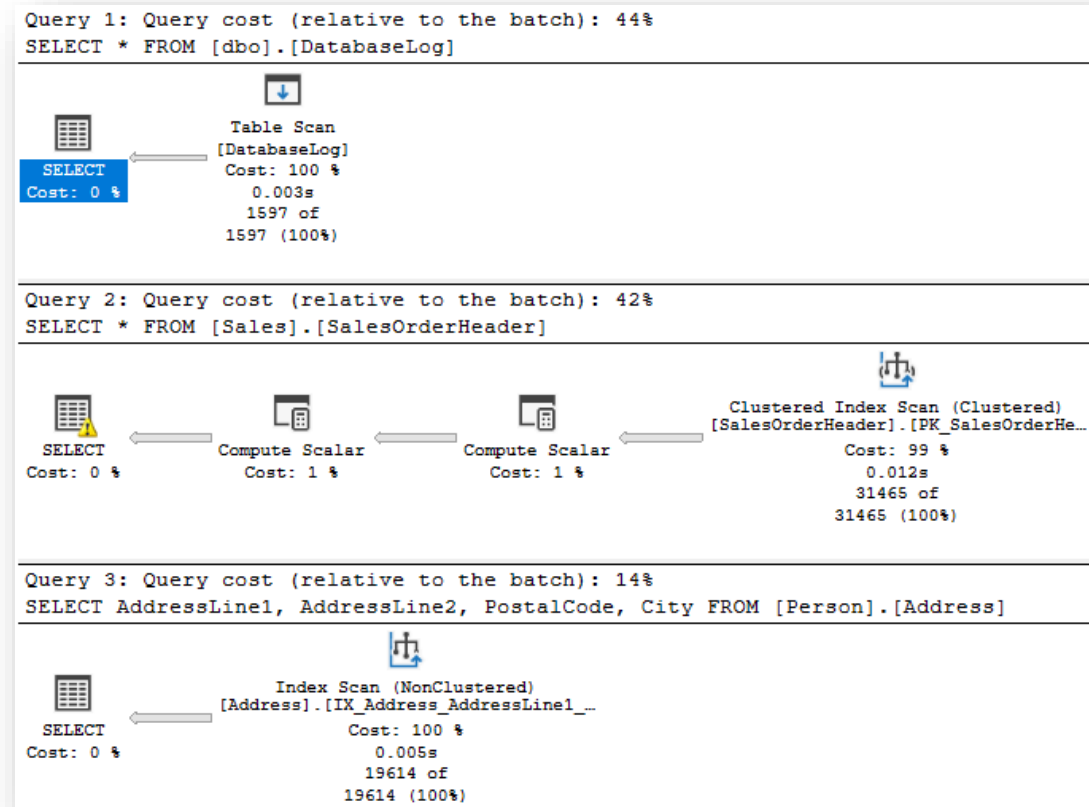  - Page split
  - Hot Page (IU/IX Locks)
  - 999 au maximum (!!!)

# Les opération

Scan, Seek & Lookup
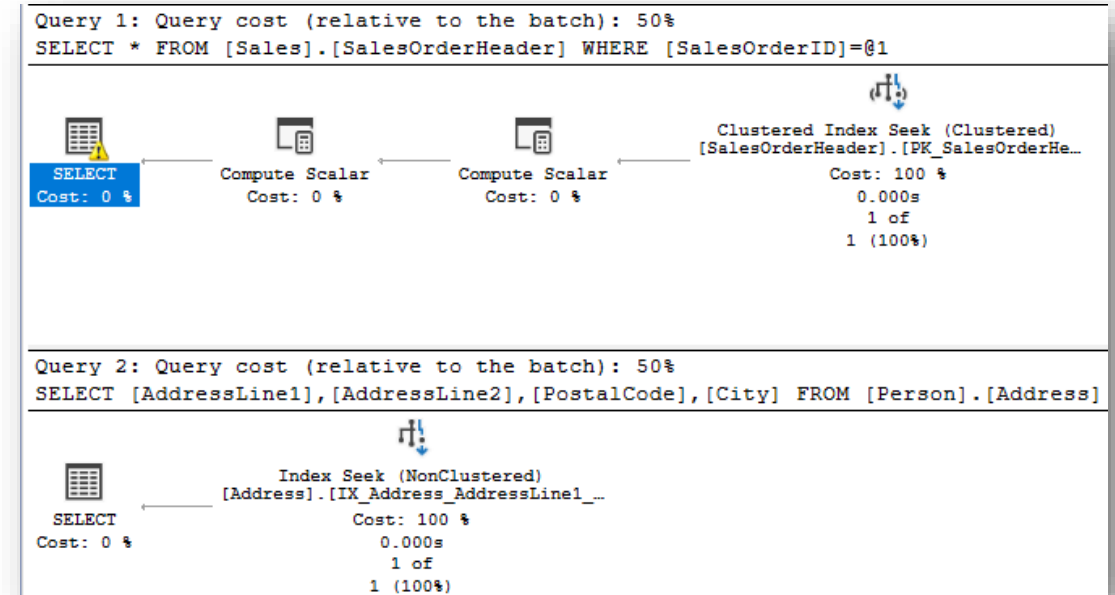
# Scan

- Table Scan

- Cluster Index Scan

- NonClustered Index Scan



Query 1: Query cost (relative to the batch): 44%
SELECT * FROM [dbo].[DatabaseLog]

```
                    Table Scan
                   [DatabaseLog]
SELECT              Cost: 100 %
Cost: 0 %             0.003s
                     1597 of
                    1597 (100%)
```

Query 2: Query cost (relative to the batch): 42%
SELECT * FROM [Sales].[SalesOrderHeader]

```
                                                      Clustered Index Scan (Clustered)
                                                      [SalesOrderHeader].[PK_SalesOrderHe...
SELECT      Compute Scalar    Compute Scalar                Cost: 99 %
Cost: 0 %    Cost: 1 %          Cost: 1 %                     0.012s
                                                             31465 of
                                                           31465 (100%)
```

Query 3: Query cost (relative to the batch): 14%
SELECT AddressLine1, AddressLine2, PostalCode, City FROM [Person].[Address]

```
                    Index Scan (NonClustered)
                   [Address].[IX_Address_AddressLine1_...
SELECT              Cost: 100 %
Cost: 0 %             0.005s
                     19614 of
                    19614 (100%)
```

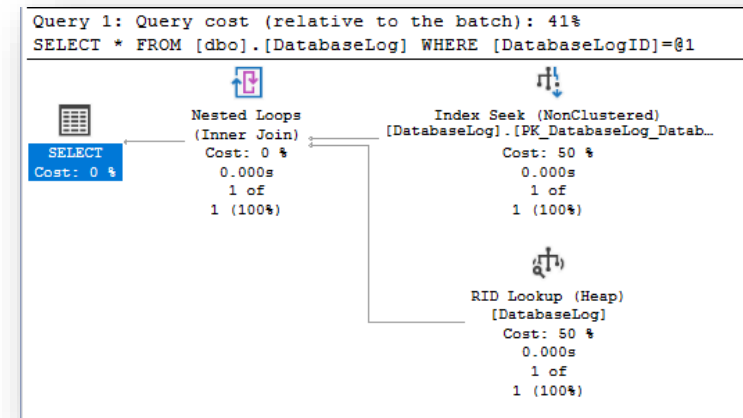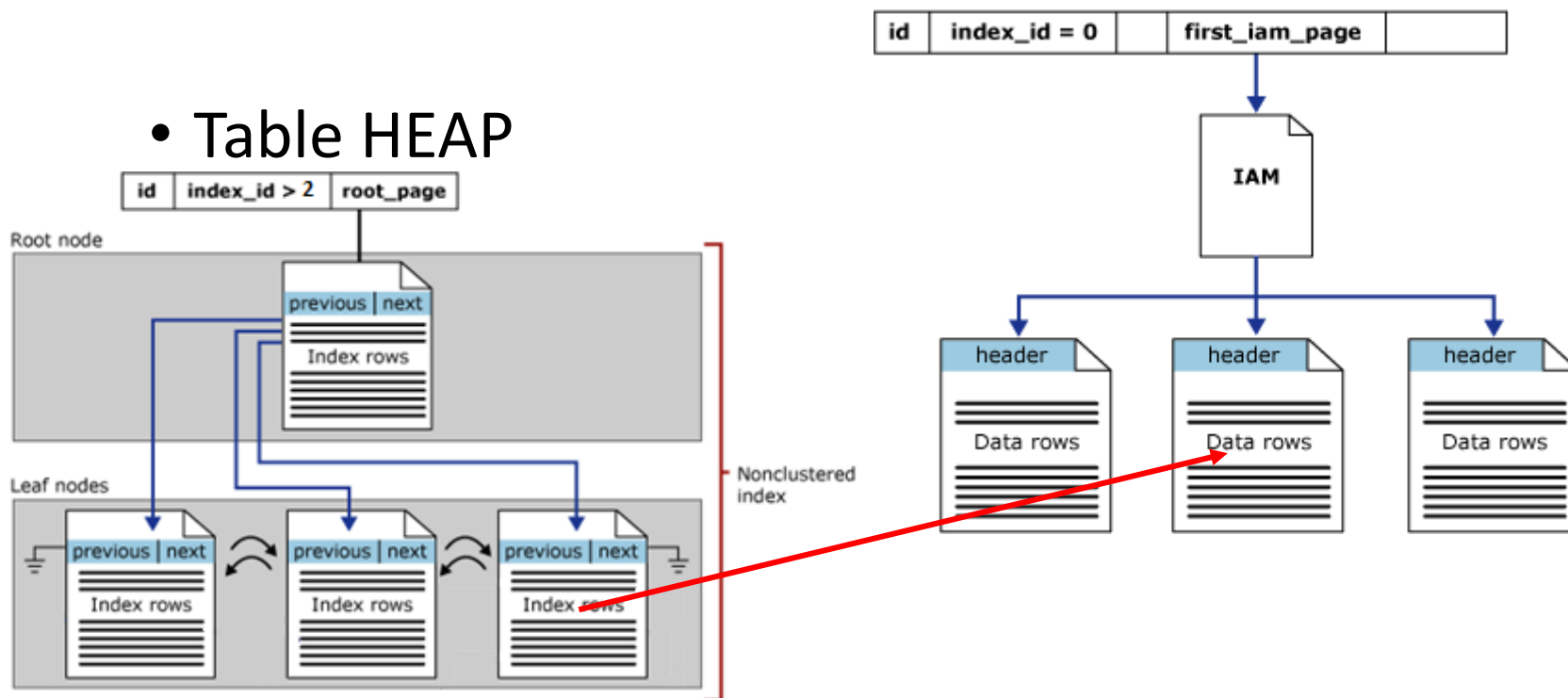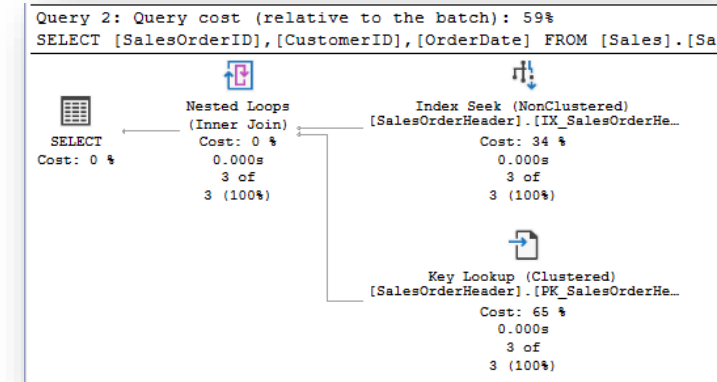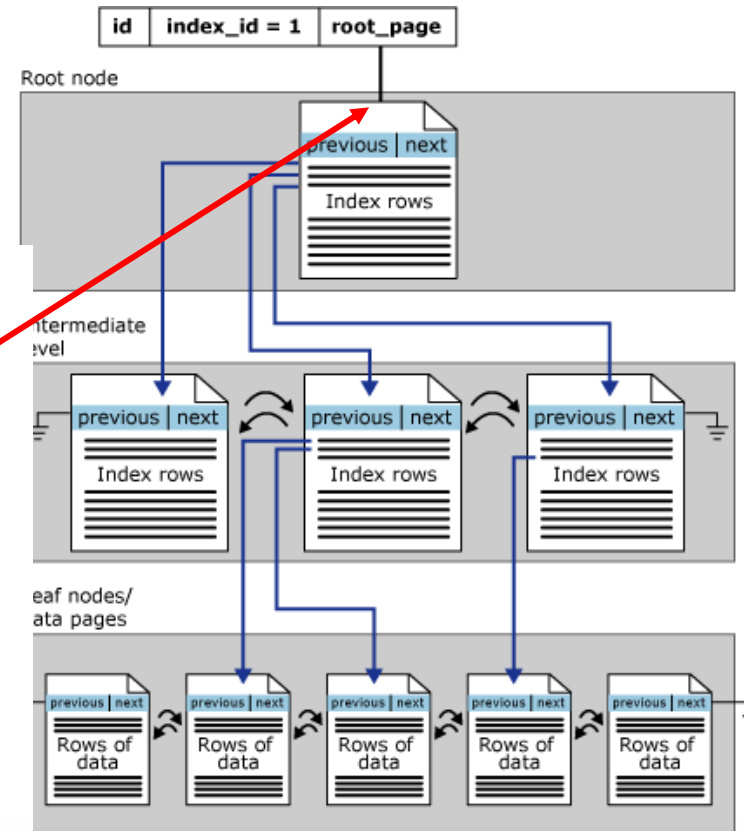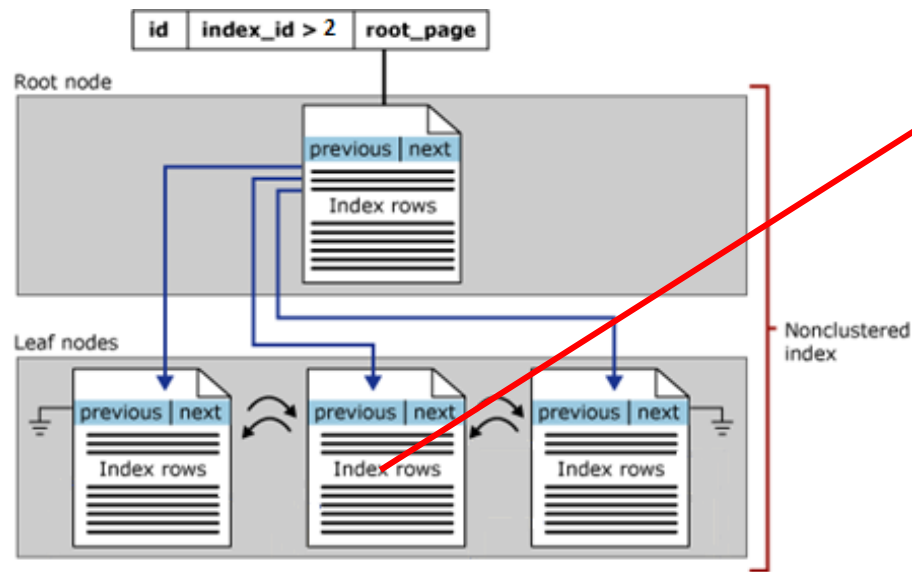| DatabaseName | SchemaName | TableName | index_name | object_id | index_id | user_scans | user_seeks | user_lookups | user_updates | last_user_scan | last_user_seek | last_user_lookup |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| AdventureWorks2017 | dbo | DatabaseLog | NULL | 901578250 | 0 | 1 | 0 | 0 | 0 | 2023-03-16 11:59:49.130 | NULL | NULL |
| AdventureWorks2017 | Sales | SalesOrderHeader | PK_SalesOrderHeader_SalesOrderID | 1922105888 | 1 | 1 | 0 | 0 | 0 | 2023-03-16 11:59:49.290 | NULL | NULL |
| AdventureWorks2017 | Person | Address | IX_Address_AddressLine1_AddressLine2_City_StateP... | 1029578706 | 3 | 1 | 0 | 0 | 0 | 2023-03-16 11:59:49.597 | NULL | NULL |

DATAFROGS

# Seek

- Clustered Index Seek
- NonClustered Index Seek



Query 1: Query cost (relative to the batch): 50%
SELECT * FROM [Sales].[SalesOrderHeader] WHERE [SalesOrderID]=@1

```
SELECT          Compute Scalar       Compute Scalar        Clustered Index Seek (Clustered)
Cost: 0 %       Cost: 0 %            Cost: 0 %             [SalesOrderHeader].[PK_SalesOrderHe...
                                                           Cost: 100 %
                                                           0.000s
                                                           1 of
                                                           1 (100%)
```

Query 2: Query cost (relative to the batch): 50%
SELECT [AddressLine1],[AddressLine2],[PostalCode],[City] FROM [Person].[Address]

```
SELECT          Index Seek (NonClustered)
Cost: 0 %       [Address].[IX_Address_AddressLine1_...
                Cost: 100 %
                0.000s
                1 of
                1 (100%)
```

| DatabaseName | SchemaName | TableName | index_name | object_id | index_id | user_scans | user_seeks | user_lookups | user_updates | last_user_scan | last_user_seek | last_user_lookup |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| AdventureWorks2017 | Sales | SalesOrderHeader | PK_SalesOrderHeader_SalesOrderID | 1922105888 | 1 | 1 | 1 | 0 | 0 | 2023-03-16 11:59:49.290 | 2023-03-16 12:04:00.593 | NULL |
| AdventureWorks2017 | Person | Address | IX_Address_AddressLine1_AddressLine2_City_StateP... | 1029578706 | 3 | 1 | 1 | 0 | 0 | 2023-03-16 11:59:49.597 | 2023-03-16 12:04:00.640 | NULL |

DATAFROGS

# Bookmark lookup – RID Lookup

- Table HEAP



| id | index_id = 0 | | first_iam_page | |

**IAM**

| id | index_id > 2 | root_page |

Root node

previous | next
Index rows

Nonclustered index

Leaf nodes

previous | next
Index rows

previous | next
Index rows

previous | next
Index rows

header
Data rows

header
Data rows

header
Data rows

```
Query 1: Query cost (relative to the batch): 41%
SELECT * FROM [dbo].[DatabaseLog] WHERE [DatabaseLogID]=@1
```

SELECT
Cost: 0 %

Nested Loops
(Inner Join)
Cost: 0 %
0.000s
1 of
1 (100%)

Index Seek (NonClustered)
[DatabaseLog].[PK_DatabaseLog_Datab...
Cost: 50 %
0.000s
1 of
1 (100%)

RID Lookup (Heap)
[DatabaseLog]
Cost: 50 %
0.000s
1 of
1 (100%)

| DatabaseName | SchemaName | TableName | index_name | object_id | index_id | user_scans | user_seeks | user_lookups | user_updates | last_user_scan | last_user_seek | last_user_lookup |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| AdventureWorks2017 | dbo | DatabaseLog | PK_DatabaseLog_DatabaseLogID | 901578250 | 2 | 0 | 1 | 0 | 0 | NULL | 2023-03-16 12:06:34.140 | NULL |
| AdventureWorks2017 | dbo | DatabaseLog | NULL | 901578250 | 0 | 1 | 0 | 1 | 0 | 2023-03-16 11:59:49.130 | NULL | 2023-03-16 12:06:34.140 |

DATAFROGS

# Bookmark lookup – Key Lookup

- Index Cluster



| DatabaseName | SchemaName | TableName | index_name | object_id | index_id | user_scans | user_seeks | user_lookups | user_updates | last_user_scan | last_user_seek | last_user_lookup |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| AdventureWorks2017 | Sales | SalesOrderHeader | IX_SalesOrderHeader_CustomerID | 1922105888 | 4 | 0 | 1 | 0 | 0 | NULL | 2023-03-16 12:10:53.713 | NULL |
| AdventureWorks2017 | Sales | SalesOrderHeader | PK_SalesOrderHeader_SalesOrderID | 1922105888 | 1 | 1 | 1 | 1 | 0 | 2023-03-16 11:59:49.290 | 2023-03-16 12:04:00.593 | 2023-03-16 12:10:53.713 |

# Bonus – Key Lookup Deep Dive

# Bonus – Key Lookup Deep Dive

| FileId | PageId | Row | Level | ChildFileId | ChildPageId | CustomerID (key) |
|--------|--------|-----|-------|-------------|-------------|------------------|
| 1 | 7496 | 0 | 1 | 1 | 7256 | NULL |
| 1 | 7496 | 1 | 1 | 1 | 7257 | 11191 |
| 1 | 7496 | 2 | 1 | 1 | 7258 | 11330 |
| 1 | 7496 | 3 | 1 | 1 | 7259 | 11519 |
| 1 | 7496 | 4 | 1 | 1 | 7260 | 11736 |
| 1 | 7496 | 5 | 1 | 1 | 7261 | 11967 |
| 1 | 7496 | 6 | 1 | 1 | 7262 | 12211 |
| 1 | 7496 | 7 | 1 | 1 | 7263 | 12475 |
| 1 | 7496 | 8 | 1 | 1 | 7264 | 12763 |
| 1 | 7496 | 9 | 1 | 1 | 7265 | 13068 |

| FileId | PageId | Row | Level | CustomerID (key) | SalesOrderID (key) | KeyHashValue | Row Size |
|--------|--------|-----|-------|------------------|--------------------|--------------|----------|
| 1 | 7256 | 0 | 0 | 11000 | 43793 | (dc9ad24a8317) | 12 |
| 1 | 7256 | 1 | 0 | 11000 | 51522 | (4ec3187d6991) | 12 |
| 1 | 7256 | 2 | 0 | 11000 | 57418 | (d0df796d1d13) | 12 |
| 1 | 7256 | 3 | 0 | 11001 | 43767 | (acd94b335138) | 12 |
| 1 | 7256 | 4 | 0 | 11001 | 51493 | (1c0f95fcd8f2) | 12 |
| 1 | 7256 | 5 | 0 | 11001 | 72773 | (4e1ca1670056) | 12 |
| 1 | 7256 | 6 | 0 | 11002 | 43736 | (dfd038d85b9f) | 12 |
| 1 | 7256 | 7 | 0 | 11002 | 51238 | (888ee458692b) | 12 |
| 1 | 7256 | 8 | 0 | 11002 | 53237 | (5b740c7fbbfb) | 12 |
| 1 | 7256 | 9 | 0 | 11003 | 43701 | (2547c24d9bdf) | 12 |

| FileId | PageId | Row | Level | CustomerID (key) | SalesOrderID (key) | KeyHashValue | Row Size |
|--------|--------|-----|-------|------------------|--------------------|--------------|----------|
| 1 | 7257 | 0 | 0 | 11191 | 70229 | (c7f2d17302a4) | 12 |
| 1 | 7257 | 1 | 0 | 11192 | 60704 | (23b7a4f27e66) | 12 |
| 1 | 7257 | 2 | 0 | 11192 | 67056 | (db2028f3f42d) | 12 |
| 1 | 7257 | 3 | 0 | 11193 | 53325 | (ae7b47f96322) | 12 |
| 1 | 7257 | 4 | 0 | 11193 | 64476 | (8a75be1a17bf) | 12 |
| 1 | 7257 | 5 | 0 | 11194 | 54355 | (c10b5a950ef3) | 12 |
| 1 | 7257 | 6 | 0 | 11194 | 72599 | (16785202e20e) | 12 |
| 1 | 7257 | 7 | 0 | 11195 | 56195 | (b9a6bf232bd5) | 12 |
| 1 | 7257 | 8 | 0 | 11195 | 68921 | (0cfe2667c8d5) | 12 |
| 1 | 7257 | 9 | 0 | 11196 | 53786 | (d4abeeb6cfd9) | 12 |

DATAFROGS

# Bonus – Key Lookup Deep Dive

# Bonus – Key Lookup Deep Dive

# Conclusion

- La modélisation physique a son importance
- /!\ Ne pas confondre
  - Clé primaire et Index Clustered
  - SQL Server / SSMS
    - Contrainte PK implémentée sous forme d'index cluster par défaut
  - MAIS
    - Gains de performance possibles à ne pas suivre cette règle …
- La fragmentation
  - N'est pas forcément votre ennemie !
- L'optimiseur est basé sur le cout …

DATAFROGS