# Informix Warehouse Accelerator Demo using Informix 12.10

Mike Walker

# Mike Walker

Mike Walker has been using Informix databases for 18 years, as a developer and as a database administrator.

Mike has been developed and supported large data warehouses for the Department of Agriculture and provides database support for other Informix installations

Contact Info:
**mike@advancedatatools.com**
**www.advancedatatools.com**
**703-256-0267 x 101**

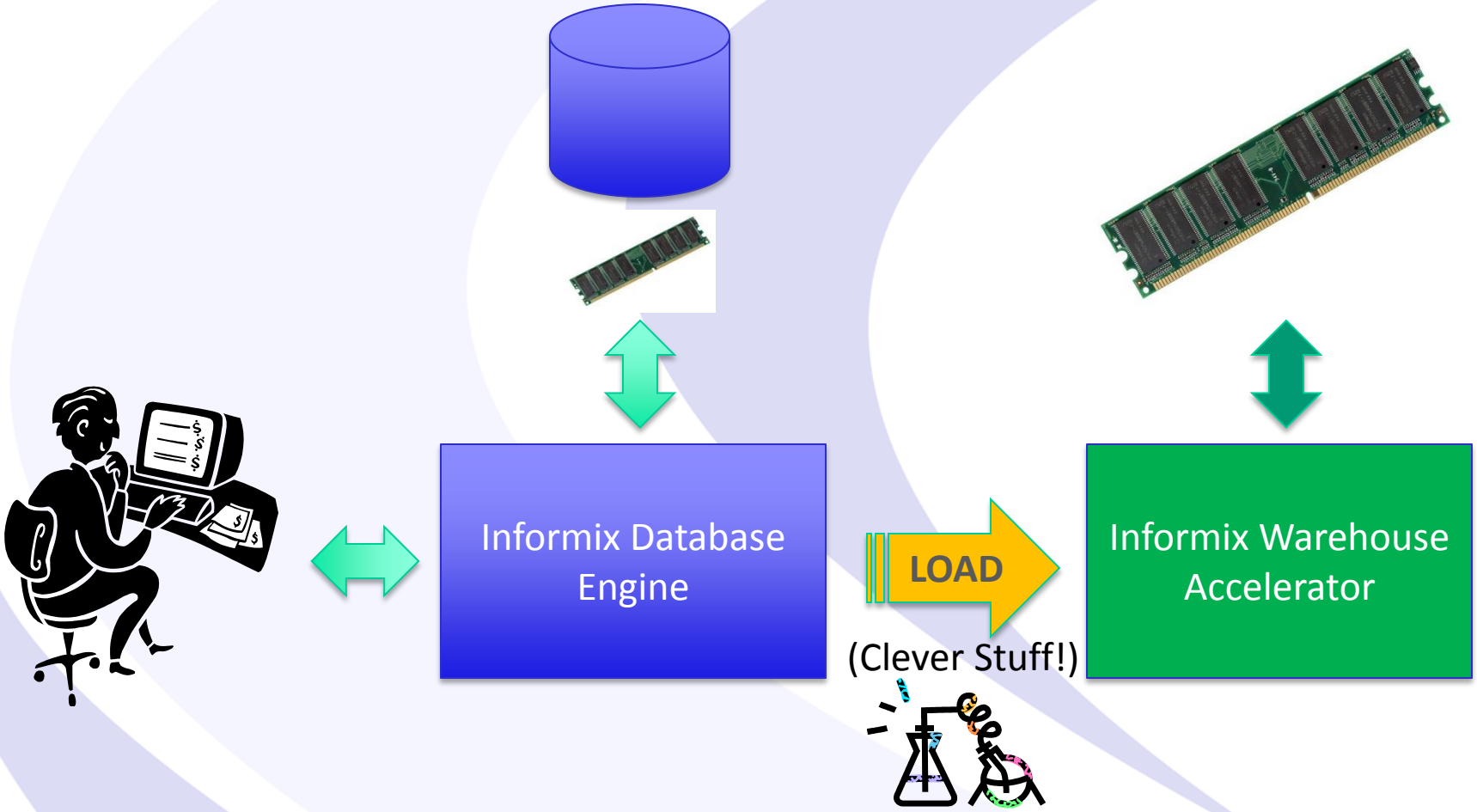
# Agenda

- What is the Informix Warehouse Accelerator?
- How do I use it?
- Benchmarks - Fast Performance Demo
- How to Setup the Accelerator
- How to Automatically Build a Data Mart using Workload Analysis

# Informix Warehouse Accelerator – what is it?

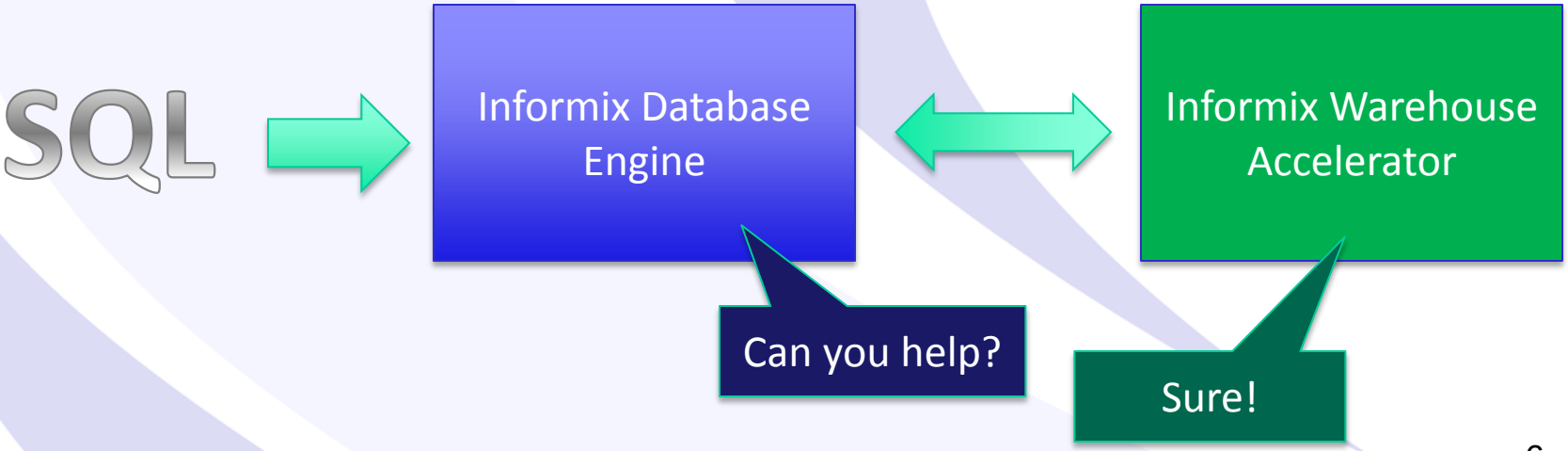
- Hardware or Software?
  - IWA is **software** that works with the Informix database to speed up queries …*a LOT*
  - Can run on its own **hardware**, or the same hardware as your database server
    - **Platform: Linux (Intel)**
      **Hardware: Intel® x86_64(EM64T/ AMD64)**

# IWA sits alongside the DB



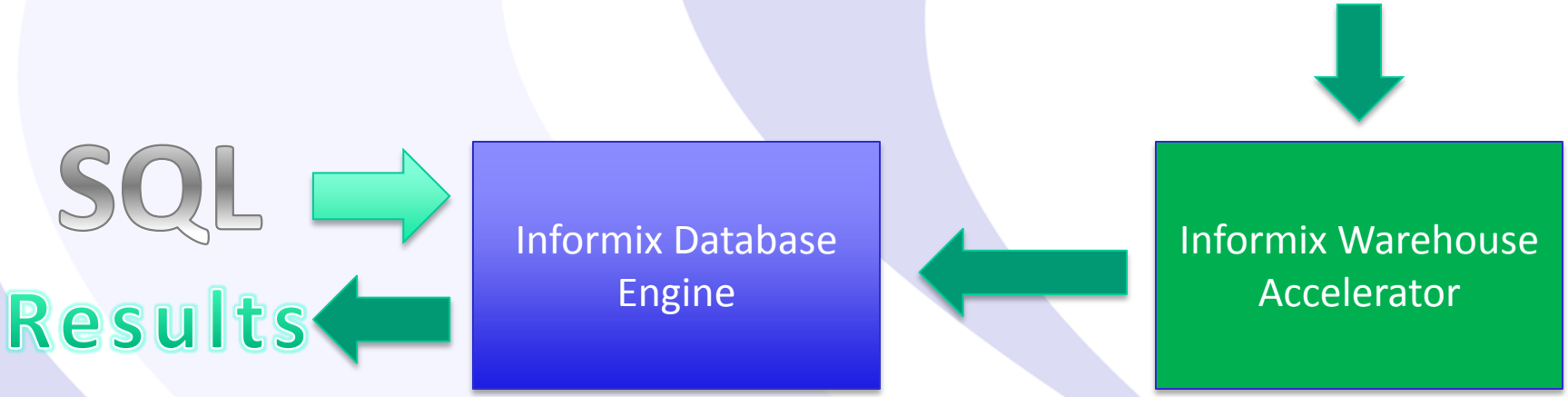
# IWA Integration

The database engine decides whether a query can be accelerated or not



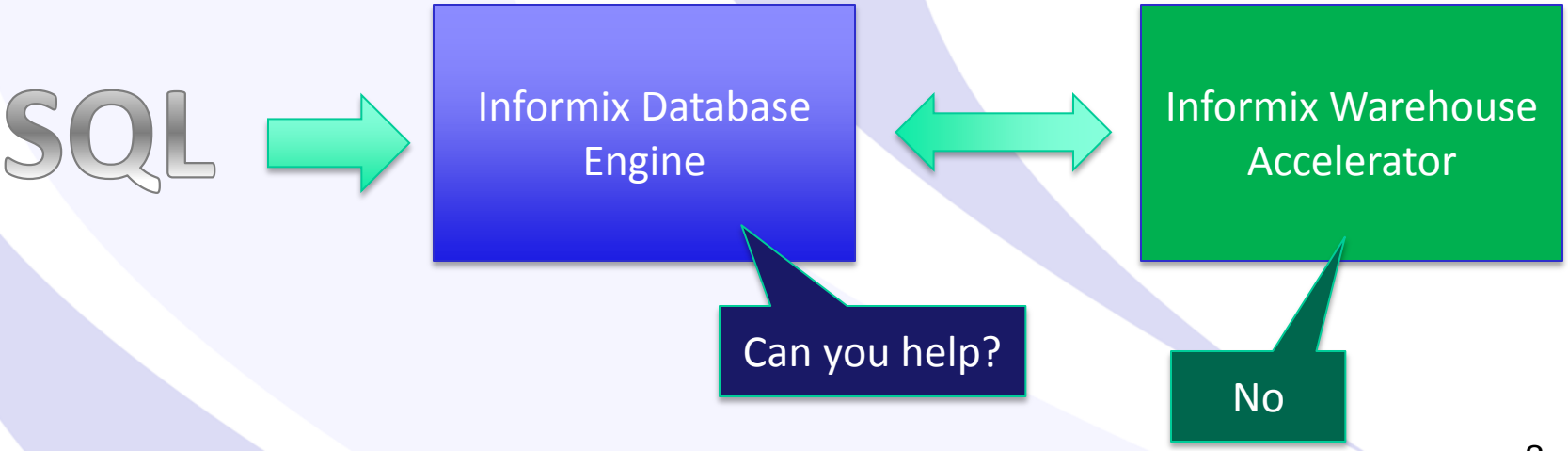
# IWA Integration

Query can be accelerated



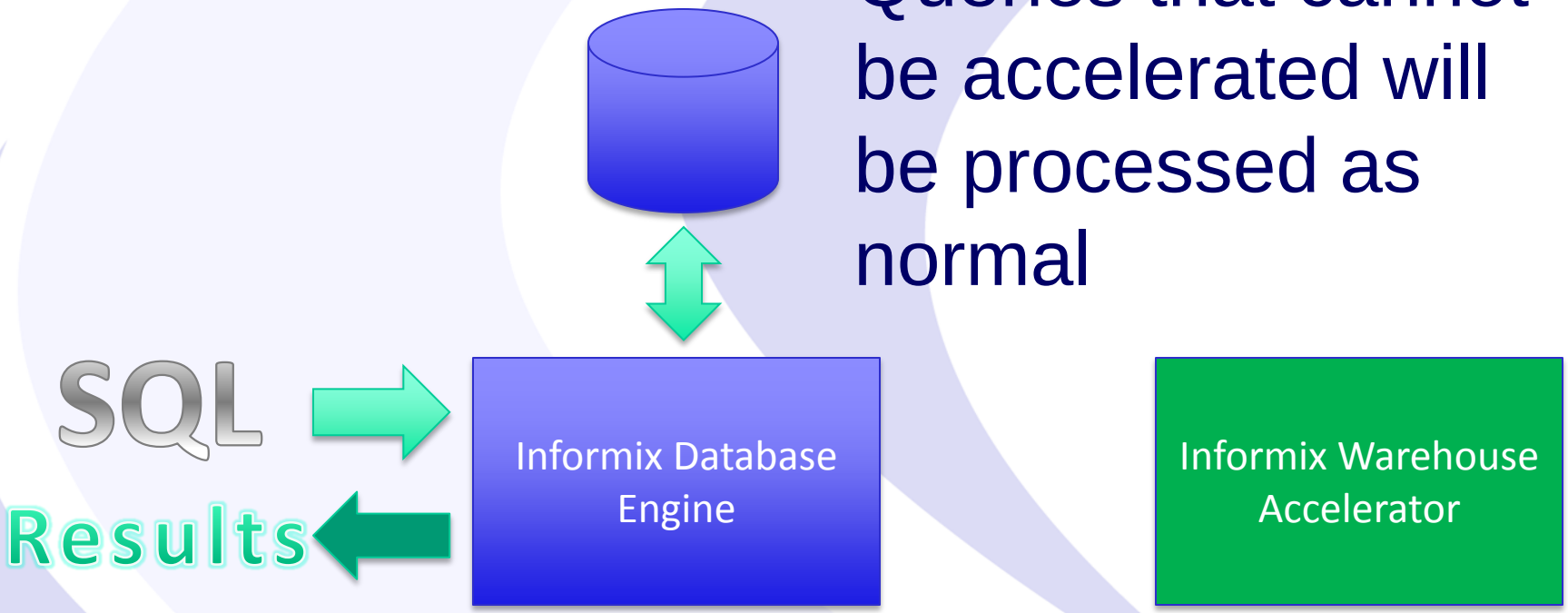
# IWA Integration

Unlikely that IWA can accelerate all queries



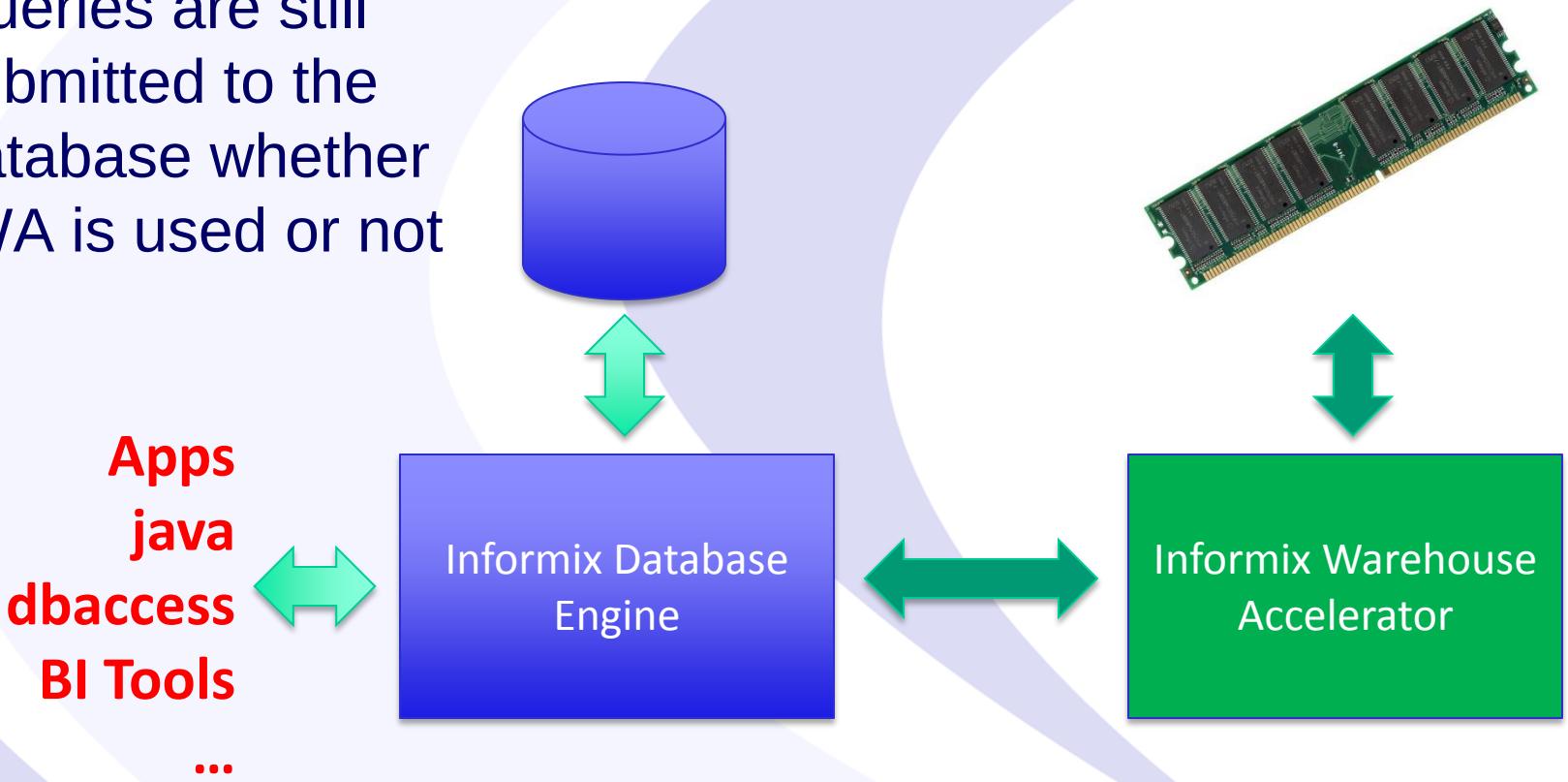
# IWA Integration

Queries that cannot be accelerated will be processed as normal

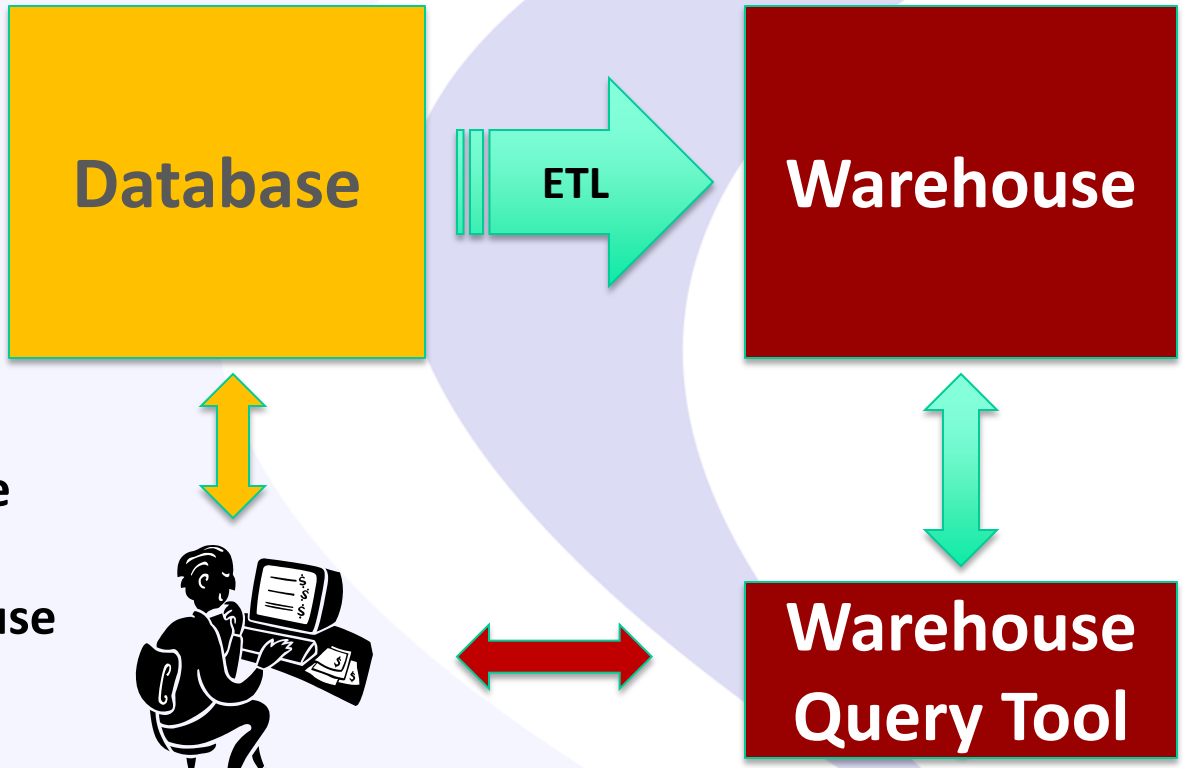


# Entry Point is the Same

Queries are still submitted to the database whether IWA is used or not



# What IWA is NOT




Database and Warehouse accessed through separate interfaces

# Test Environment

- IBM System x3500 M3
- 2 Intel Xeon X5675 CPUs (24 threads)
- 128 GB RAM
- SUSE Enterprise 11
- IDS 12.10.FC1
- IWA installed on same box

# Demo

- How do I use IWA for my queries?

# How do I use it?

Summarize the total sales ($) by year:

```
SET ENVIRONMENT use_dwa 'accelerate on';

select transaction_year, sum(total_price_paid)
from sales_fact
group by 1
order by 1;
```

Can include
set environment use_dwa 'accelerate on'
in the sysdbopen() stored procedure so that it applies to all statements

IWA takes this query from over 14 minutes to 7 seconds

# Explain Plan of Accelerated Query

Explain Plan shows the REMOTE PATH and the Accelerated Query Table (AQT):

```
select transaction_year, sum(total_price_paid)
from sales_fact
group by 1
order by 1

Estimated Cost: 3074594816
Estimated # of Rows Returned: 12

 1) sales@SALESACL1210:informix.aqt44de2a69-50b5-42e0-82b8-
10d04a07ec4f: DWA REMOTE PATH

    Remote SQL Request:
    {QUERY {FROM  informix.aqt44de2a69-50b5-42e0-82b8-
10d04a07ec4f} {SELECT {SYSCAST COL02 AS SMALLINT NULLABLE} {SUM
COL12 } } {GROUP COL02 } {ORDER COL02  ASC  NULLSFIRST } }
```

# Explain Plan of Accelerated Query

```
select transaction_year, sum(total_price_paid)
from sales_fact
group by 1
order by 1

Estimated Cost: 3074594816
Estimated # of Rows Returned: 12
Temporary Files Required For: Group By

  1) informix.sales_fact: SEQUENTIAL SCAN  (Serial, fragments: ALL)


Query statistics:
-----------------

  Table map :
  ----------------------------
  Internal name     Table name
  ----------------------------

  type     rows_prod  est_rows  time       est_cost
  ----------------------------------------------
  dwa      12         0         00:06.66   0
```

# How do I use it?

- If a query cannot be accelerated, it will be processed as a regular query

- Disable this behavior with:

```
SET ENVIRONMENT use_dwa 'fallback off'
```

# How do I use it?

```
SET ENVIRONMENT use_dwa 'accelerate on';
SET ENVIRONMENT use_dwa 'fallback off';

select transaction_year, max(load_id)
from sales_fact
group by 1;
#        ^
#26404: query cannot be accelerated, fallback to
local execution is not allowed
#
```

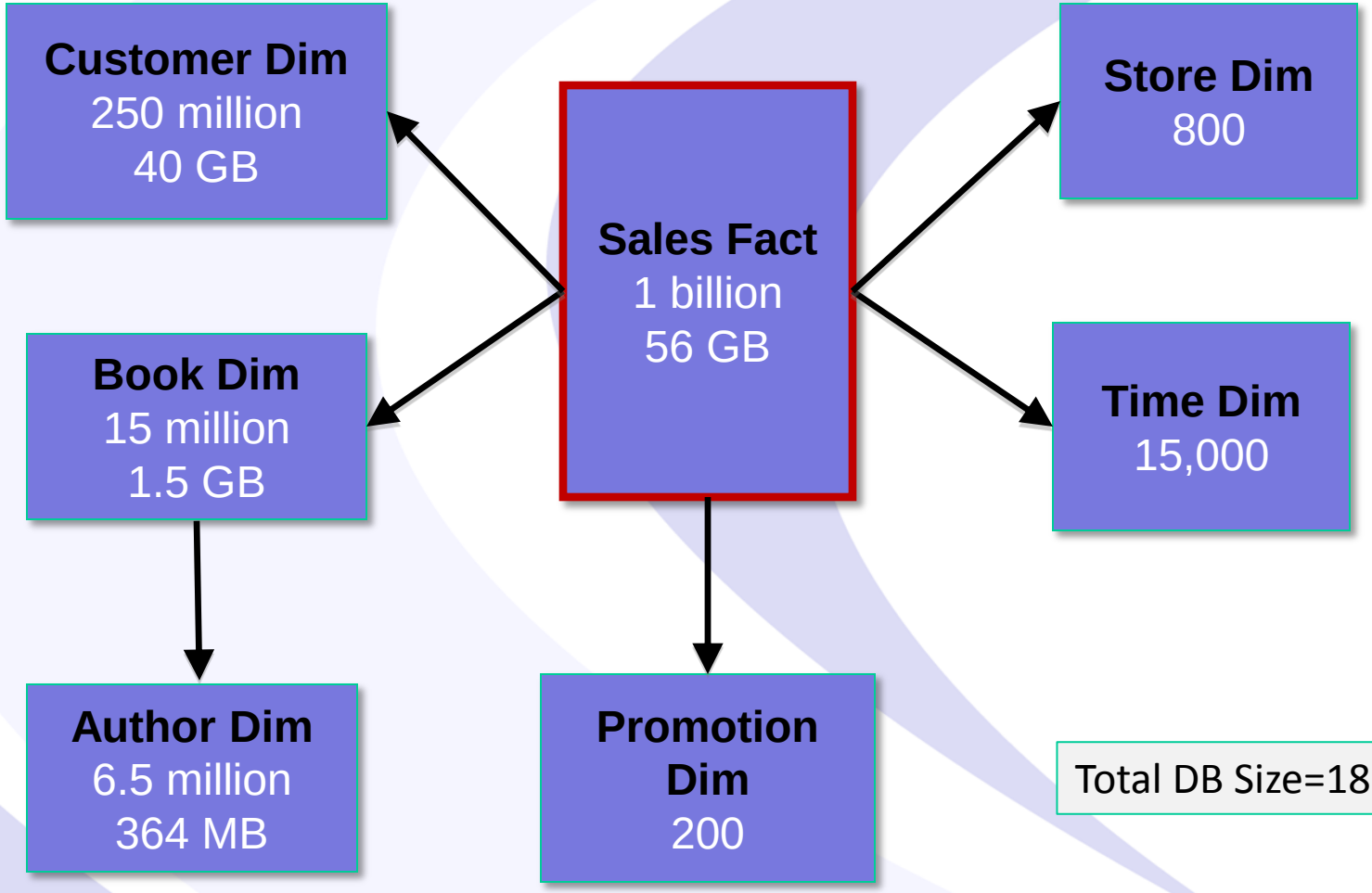
Column ‘load_id’ is not included in the datamart – queries that use it cannot be accelerated

# Benchmarks

# How these benchmarks differ from earlier ones

- Queries expanded to return larger data sets
- Additional query tuning
- Additional engine tuning
- Non Accelerated timings performed with much more memory given to the engine and many more CPU VPs
- IWA configuration modified (fewer nodes, more CPU per node)

# Test Schema




Total DB Size=181 GB

# Test Queries

| | | sales_fact | time_dim | stores_dim | books | customers | authors | promotion_dim |
|---|---|---|---|---|---|---|---|---|
| Test 1 | List the total sales (by dollar amount) by year. | X | | | | | | |
| Test 2 | Show the days which had the highest revenue over the previous 12 years. | X | X | | | | | |
| Test 3 | Show the 10 bestselling books in a specified 6 months | X | X | | X | | | |
| Test 4 | Show the top 10 customers (by amount spent) over 6 states. | X | | | | X | | |
| Test 5 | Show the number of sales of books by a specified author over 12 years, by period and geographical area. | X | X | X | X | | X | |
| Test 6 | Show the non-web sale promotions over a 12 year period, with the cost of each. | X | | | | | | X |
| Test 7 | Show the number of books sold by the day of the week over an 11 year period. | X | X | | | | | |
| Test 8 | Show the number of books sold to customers resident in the same state as the book store vs non-resident. | X | | X | | X | | |
| Test 9 | Show the income per square foot for each store over a 12 year period. | X | X | X | | | | |
| Test 10 | Show a selected customer's purchase history (look up by customer_id [indexed]) | X | X | X | X | X | X | X |
| Test 11 | Show a selected customer's purchase history (look up by customer name [not indexed]) | X | X | X | X | X | X | X |
| Test 12 | Show the non-web sales by period and store. | X | X | X | | | | |
| Test 13 | Show a count of customer sales by state | X | | | | X | | |

# Test Timings

| Test | Informix (h:mm:ss) | With IWA (h:mm:ss) | % Decrease with IWA |
|---|---|---|---|
| 1 | 0:14:22 | 0:00:07 | 99% |
| 2 | 0:02:43 | 0:00:20 | 88% |
| 3 | 0:04:11 | 0:00:35 | 86% |
| 4 | 0:13:49 | 0:02:06 | 85% |
| 5 | 0:00:11 | 0:00:05 | 55% |
| 6 | 0:03:43 | 0:00:28 | 87% |
| 7 | 0:02:21 | 0:00:13 | 91% |
| 8 | 0:14:53 | 0:05:48 | 61% |
| 9 | 0:06:46 | 0:01:24 | 79% |
| 10 | 0:00:01 | 0:00:27 | -2600% |
| 11 | 0:08:06 | 0:00:29 | 94% |
| 12 | 0:10:05 | 0:00:37 | 94% |
| 13 | 0:15:35 | 0:04:52 | 70% |
| TOTAL | 1:36:45 | 0:17:22 | 82% |

# Test Query 1

## 14m 22s → 7s *(99% decrease)*

Summarize the total sales ($) by year:

```
select transaction_year, sum(total_price_paid)
from sales_fact
group by 1
order by 1;
```

# Test Query 2

## 2m 43s → 20s *(88% decrease)*

Show the days which had the highest revenue over the previous 12 years:

```
select /*+USE_HASH(t)*/ f.transaction_year,
       t.date,
       sum(f.product_cost * f.product_qty) revenue
from sales_fact f,
     time_dim t
where f.date_id = t.date_id
  and t.year between 2000 and 2011
  and f.transaction_year between 2000 and 2011
group by 1,2
order by 3 desc;
```

# Test Query 3

## 4m 11s → 35s *(86% decrease)*

Show the 10 bestselling books in a specified 6 months:

```
select first 10 b.product_id, b.title, b.price, sum(f.product_qty) num_sold
from time_dim t,
    sales_fact f,
    books b
where t.quarter in (1,2)
  and t.year = 2010
  and t.date_id = f.date_id
  and f.product_id = b.product_id
group by 1,2,3
order by 4 desc;
```

# Test Query 4

## 13m 49s → 2m 6s *(85% decrease)*

Show the top 10 customers (by amount spent) over 6 states:

```
select /*+ORDERED FULL(f) FULL(c)*/ first 10
      c.customer_id,
      c.state,
      sum(total_price_paid) total_paid,
      sum(product_qty) num_books
from
    sales_fact f,
    customers c
where f.transaction_year between 2000 and 2011
  and f.customer_id = c.customer_id
  and c.state in ("CA", "TX", "PA", "FL", "NY", "GA")
group by 1,2
order by 3 desc;
```

# Test Query 5

## 11s → 5s *(55% decrease)*

Show the number of sales of books by a specified author over 12 years, broken down by period and geographical area:

```
select  /*+ORDERED AVOID_HASH(f)*/
  f.transaction_year, t.quarter, s.regionname, s.divisionname, sum(product_qty)
from authors a,
    books b,
    sales_fact f,
    stores_dim s,
    time_dim t
where a.name = "Tom Clancy"
 and a.author_id = b.author_id
 and b.product_id = f.product_id
 and f.store_id = s.store_id
 and f.date_id = t.date_id
 and f.transaction_year >= 2000
 and t.year >= 2000
group by 1,2,3,4
order by 1,2,3,4;
```

# Test Query 6

## 3m 43s → 28s *(87% decrease)*

Show the non-web sale promotions over a 12 year period, with the cost of each:

```
select f.promotion_id,
       p.promotion_name,
       sum((f.product_cost * f.product_qty)) full_price,
       sum(f.total_price_paid) discount_price,
       sum((f.product_cost * f.product_qty) - f.total_price_paid) discount_amount
from sales_fact f,
     promotion_dim p
where f.transaction_year between 2000 and 2011
  and f.promotion_id = p.promotion_id
  and p.website = "N"
group by 1,2
order by 1,2;
```

# Test Query 7

## 2m 11s → 13s *(91% decrease)*

Show the number of books sold by the day of the week over an 11 year period:

```
select /*+FULL(f) FULL(t)*/
  t.dayofweek, t.dayofweekname, sum(f.product_qty)
from sales_fact f,
    time_dim t
where f.date_id = t.date_id
 and f.transaction_year > 2000
group by 1,2
order by 1;
```

# Test Query 8

## 14m 53s → 5m 48s *(61% decrease)*

Show the number of books sold to customers resident in the same state as the book store, and the number sold to customers not-resident in the same state:

```
select /*+FULL(s) FULL(c) FULL(f)*/
  case
    when s.state = c.state then "IN_STATE"
    else "OUT_STATE"
  end,
  sum(f.product_qty)
from sales_fact f,
   stores_dim s,
   customers c
where f.store_id = s.store_id
 and f.customer_id = c.customer_id
group by 1
order by 1;
```

# Test Query 9

## 6m 46s → 1m 24s *(79% decrease)*

Show the income per square foot for each store over a 12 year period:

```
select /*+FULL(f) USE_HASH(t)*/ s.store_id,
       t.year,
       t.quarter,
       s.state,
       s.postalcode zip,
       s.coffee_shop,
       (today - date_opened) days_open,
       sum(round(f.product_cost * f.product_qty)/s.size_sq_ft) rev_sq_ft
from sales_fact f,
     stores_dim s,
     time_dim t
where f.store_id = s.store_id
  and f.date_id = t.date_id
  and t.year between 2000 and 2011
group by 1,2,3,4,5,6,7
order by 1,2,3,8 desc;
```

# Test Query 10

## 1s → 27s (*2600% increase*)

Show the purchase history of a specific customer, using the unique customer identifier (customer_id [INDEXED]):

```
select /*+AVOID_HASH(s) AVOID_HASH(p)*/ c.last_name,  c.first_name,
      t.date,
      s.storename,
      s.state,
      b.title,
      a.name,
      f.product_qty,
      p.promotion_name
from customers c,
    sales_fact f,
    stores_dim s,
    time_dim t,
    books b,
    authors a,
    promotion_dim p
where c.customer_id = 250183
  and c.customer_id = f.customer_id
  and f.store_id = s.store_id
  and f.date_id = t.date_id
  and f.product_id = b.product_id
  and b.author_id = a.author_id
  and f.promotion_id = p.promotion_id
order by 3;
```

The customer_id field is indexed, as are all joins

IDS performance is better than the IWA performance

# Test Query 11

## 8m 6s → 29s *(94% decrease)*

Show the purchase history of a specific customer, using the customer name (last & first [NOT INDEXED]):

```
select /*+AVOID_HASH(s) AVOID_HASH(t) AVOID_HASH(a) AVOID_HASH(p)*/ c.last_name, c.first_name,
      t.date,
      s.storename,
      s.state,
      b.title,
      a.name,
      f.product_qty,
      p.promotion_name
from customers c,
    sales_fact f,
    stores_dim s,
    time_dim t,
    books b,
    authors a,
    promotion_dim p
where c.last_name="CHRISTENSEN"
  and c.first_name="RUSSELL"
  and c.customer_id = f.customer_id
  and f.store_id = s.store_id
  and f.date_id = t.date_id
  and f.product_id = b.product_id
  and b.author_id = a.author_id
  and f.promotion_id = p.promotion_id
order by 3;
```

The customer name fields are NOT indexed. All joins are indexed

IWA performance is better than the IDS performance, and similar to the previous query example

# Test Query 12

## 10m 5s → 37s *(94% decrease)*

Show the non-web sales by period and store:

```
select
     f.transaction_year,
     t.quarter,
     t.month,
     f.store_id,
     s.state,
     sum(f.product_cost * f.product_qty) tot_cost_before_promo,
     sum(f.total_price_paid) tot_paid,
     sum(f.product_qty) tot_books
from
    time_dim t,
    stores_dim s,
    sales_fact f
where f.date_id = t.date_id
  and f.store_id = s.store_id
  and s.website = "N"
group by 1,2,3,4,5;
```

# Test Query 13

## 15m 35s → 4m 42s *(70% decrease)*

Count the number of sales by customer state:

```
select  /*+ORDERED FULL(f) FULL(c)*/
  c.state, count(*)
from sales_fact f, customers c
where c.customer_id = f.customer_id
group by 1
order by 2 desc;
```

# Timings without IWA

```
Running ALL Tests.  Use Accelerator: N.  PDQPRIORITY: 100

Test 1  OK        12 row(s) retrieved.        Time: 14:25.82
Test 2  OK        4359 row(s) retrieved.      Time: 2:41.78
Test 3  OK        10 row(s) retrieved.        Time: 4:07.25
Test 4  OK        10 row(s) retrieved.        Time: 11:17.74
Test 5  OK        432 row(s) retrieved.       Time: 0:09.96
Test 6  OK        214 row(s) retrieved.       Time: 3:47.63
Test 7  OK        7 row(s) retrieved.         Time: 2:21.75
Test 8  OK        2 row(s) retrieved.         Time: 12:20.63
Test 9  OK        33766 row(s) retrieved.     Time: 6:50.03
Test 10 OK        61 row(s) retrieved.        Time: 0:00.75
Test 11 OK        1345 row(s) retrieved.      Time: 6:47.21
Test 12 OK        100989 row(s) retrieved.    Time: 10:12.22
Test 13 OK        48 row(s) retrieved.        Time: 15:21.28


real 5425.20
user 1.06
sys 0.10
```

1 hour 30 minutes

# Timings with IWA

```
Running ALL Tests.  Use Accelerator: Y.  PDQPRIORITY: 0

Test 1  OK        12 row(s) retrieved.        Time: 0:07.20
Test 2  OK        4359 row(s) retrieved.      Time: 0:20.34
Test 3  OK        10 row(s) retrieved.        Time: 0:34.61
Test 4  OK        10 row(s) retrieved.        Time: 2:06.26
Test 5  OK        432 row(s) retrieved.       Time: 0:05.38
Test 6  OK        214 row(s) retrieved.       Time: 0:27.65
Test 7  OK        7 row(s) retrieved.         Time: 0:13.24
Test 8  OK        2 row(s) retrieved.         Time: 5:47.74
Test 9  OK        33766 row(s) retrieved.     Time: 1:24.21
Test 10 OK        61 row(s) retrieved.        Time: 0:26.73
Test 11 OK        1345 row(s) retrieved.      Time: 0:28.73
Test 12 OK        100989 row(s) retrieved.    Time: 0:37.39
Test 13 OK        48 row(s) retrieved.        Time: 4:42.34

real 1042.23
user 1.03
sys 0.09
```

**No IWA**

| |
|---|
| 14:25.82 |
| 2:41.78 |
| 4:07.25 |
| 11:17.74 |
| 0:09.96 |
| 3:47.63 |
| 2:21.75 |
| 12:20.63 |
| 6:50.03 |
| 0:00.75 |
| 6:47.21 |
| 10:12.22 |
| 15:21.28 |

17 minutes 22 seconds

# Test Notes

- Fact table compressed from 56 GB to 19 GB in IWA
- Best performance gains on table and partition scans
- Not much benefit for selects for few records that can be retrieved from an Informix Index

# Informix Warehouse Accelerator

- No change was made to queries for testing with and without the Informix Warehouse Accelerator
- Used Environment Setting to turn IWA on and off
- All queries ran in dbaccess

# Test Configuration

| | Without IWA | With IWA | |
|---|---|---|---|
| BUFFERS | 5 GB | 4 GB | onconfig |
| CPU VPs | 100 | 100 | |
| SHMVIRTSIZE | 100 GB | 23 GB | |
| DS_TOTAL_MEMORY | 95 GB | 22 GB | |
| PDQPRIORITY | 100 | 0 | environment |
| PSORT_NPROCS | 40 | 40 | |
| tmpfs | | 100 GB | |
| NUM_NODES | | 2 | dwainst.conf |
| WORKER_SHM | | 60 GB | |
| COORDINATOR_SHM | | 4 GB | |
| CORES_FOR_SCAN_THREADS_PERCENTAGE | | 100 | |
| CORES_FOR_LOAD_THREADS_PERCENTAGE | | 100 | |

# Non Accelerated Queries

- Timings of non-accelerated queries were obtained with IWA stopped and the 100 GB of memory assigned to the engine

- Large DS_TOTAL_MEMORY helps with many queries, esp. Test 8 and Test 13
  - Lowering DS_TOTAL_MEMORY from 95 GB to 22 GB changes total run time from 1.5 hours to 5 hours

# Non Accelerated Queries

- Caching of linux filesystems helps performance of queries run without IWA

  Test 1

  - Initial run – 14 min 23 sec
  - After engine bounce, next run 1 min 12 sec

  Test 13

  - In test series, query completes in about 15 mins
  - Run on its own after cache is cleared, query completed in over 23 mins
- Without the free memory on this box, this caching would be limited

# Informix Warehouse Accelerator - Setup

- Install IWA software
- Start IWA
- Create an Accelerator
- Create a Datamart definition
- Load the Datamart
- Configure applications to use Accelerator

# IWA - Installation

- Get the package
- iwa_install
- Prompts for:
  - Installation directory
  - Starting port
  - Number of nodes
  - Memory for worker nodes/coordinator node

# IWA - Installation

- Create a SBSPACE and add it as a default sbspace to the config:

  SBSPACENAME sbspace1

- Set:
  - PATH
  - CLASSPATH
  - IBM_JAVA_OPTIONS (optional)
- Create a connection file for java commands

# IWA - Starting

- Use the ondwa command to administer IWA
- To start IWA for the first time:

```
ondwa setup
ondwa start
```

# ondwa status

Shows 1 Coordinator Node and the 3 Worker Nodes that have been configured for an earlier test

```
> ondwa status

ID    | Role         | Cat-Status   | HB-Status     | Hostname          | System ID
-----+-------------+-------------+--------------+------------------+----------
0     | COORDINATOR | ACTIVE       |   Healthy     | Babe              | 1
1     | WORKER       | ACTIVE       |   Healthy     | Babe              | 2
2     | WORKER       | ACTIVE       |   Healthy     | Babe              | 4
3     | WORKER       | ACTIVE       |   Healthy     | Babe              | 3

Cluster is in state     : Fully Operational
Expected node count     : 1 coordinator and 3 worker nodes
```

# “ondwa” Command

Use ondwa to monitor, start, stop and rebuild the Accelerator

```
usage: ondwa setup
       start
       status
       getpin
       tasks
       stop [-f]
       reset
       clean
```

# IWA – Create an Accelerator

- IWA is now running, but need to make the engine aware of it – need to create an Accelerator

# IWA – Create an Accelerator

- Java

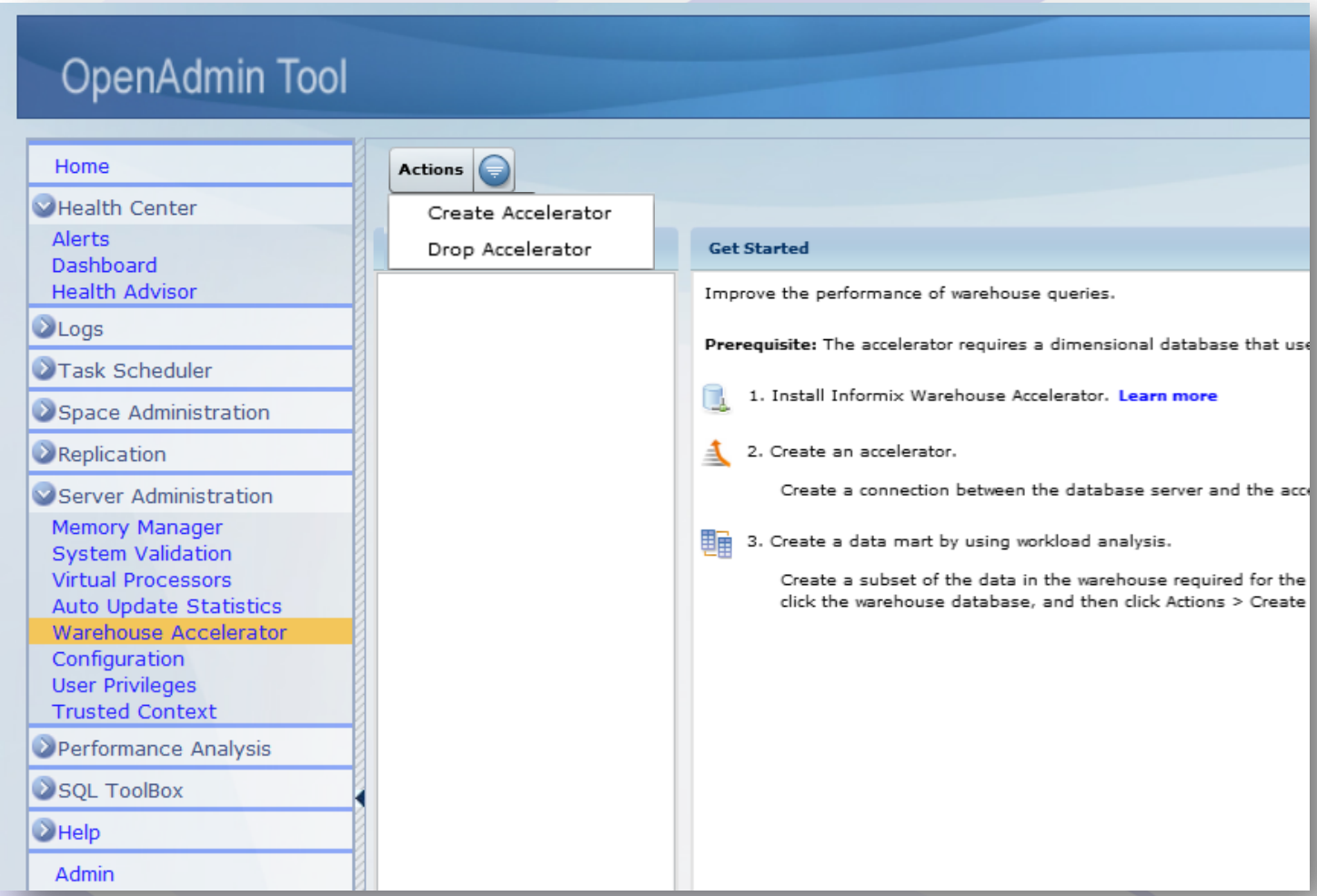
```
java setupDWA <accelerator name>
<IP address> <port> <password>
```

- SQL Administration Routine

```
EXECUTE FUNCTION
ifx_setupDWA('MyAccelerator',
'127.0.0.1', '21022', '1234');
```

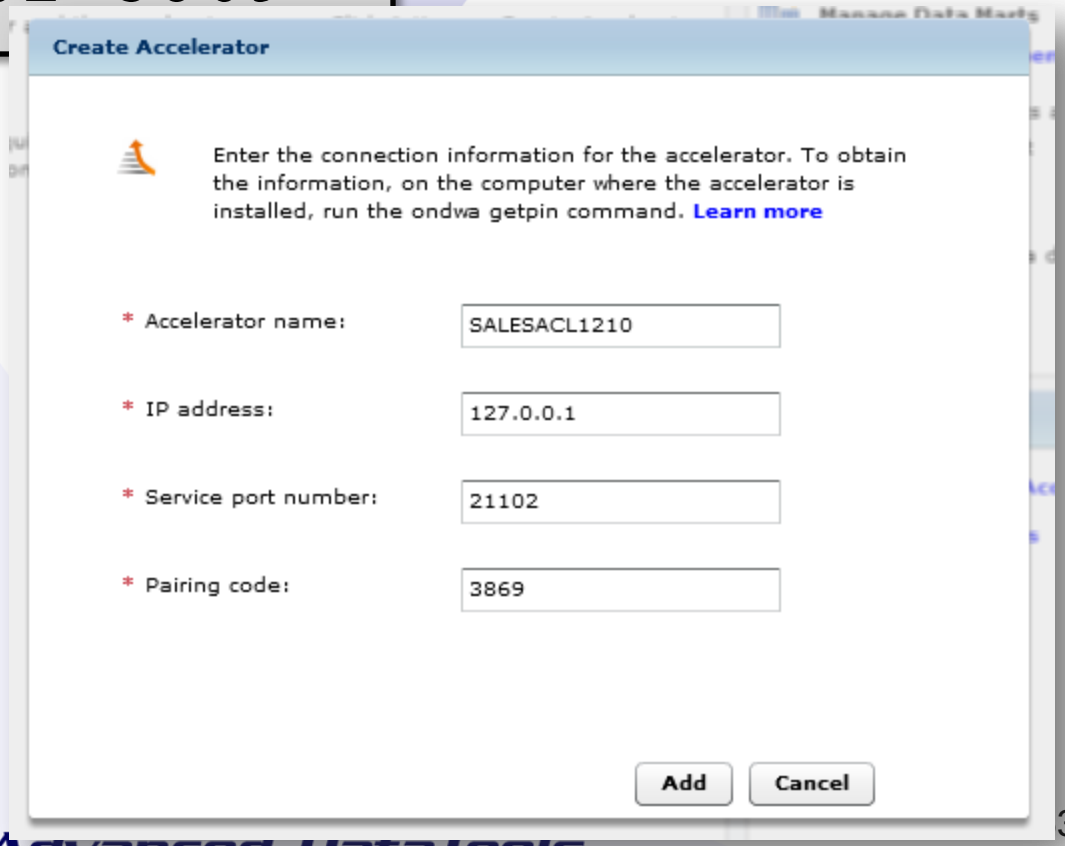
- Smart Analytics Optimizer Studio
- OAT

# Create an Accelerator with OAT

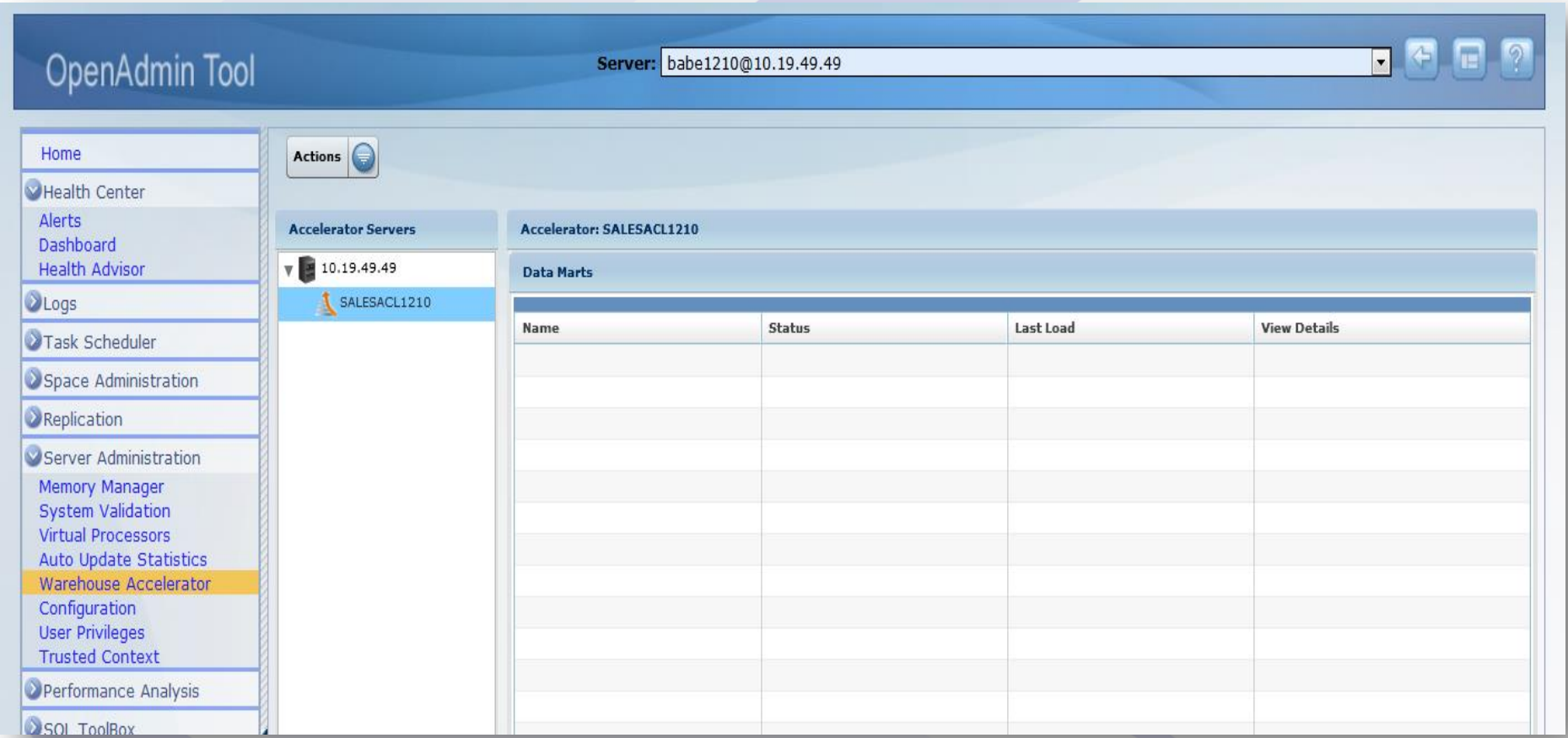


# Create an Accelerator with OAT

```
ondwa getpin
127.0.0.1 21102 3869
```



# Create an Accelerator with OAT



# IWA – Create an Accelerator

- Entry added to sqlhosts

```
SALESACL1210    group   -       -
c=1,a=3b56325f32543e60487e36242437564e49796
15e2053437775786a5f3e7b2a4a2e5f473f69304271
2c593e78752270646d587a22345339
SALESACL1210_1  dwsoctcp        127.0.0.1
21102   g=SALESACL1210
```

# IWA – Create a Data Mart

- The Data Mart is a representation of the database
- Contains the definition of what is to be accelerated:
  - Tables
  - Columns
  - Joins
- Objects not in the Data Mart definition will NOT be Accelerated

# IWA – Create a Data Mart

- Create initial Data Mart using
  - Smart Analytics Optimizer Studio
  - Workload Analysis
    - Trace SQL and process with administration functions
    - OAT

# Smart Analytics Optimizer Studio

- GUI
- Can run from local computer or server
- Simple to create the initial Data Mart

# Creating a Data Mart with Studio

- Use Studio to draw a picture of the items to be accelerated
- Choose the tables to include
- Joins will be added automatically if referential integrity is in place
- Choose the columns to be accelerated

# Studio – “sales” Data Mart

Data - Deployed data marts - SALESACL - sales/sales/default.mart_diagram - IBM Smart Analytics Optimizer Studio

File Edit Diagram Navigate Search Project Data Data Mart Window Help

Project Explor | Data Project E

- Deployed data marts - SALESACL - sales
  - Data Marts
    - sales (sales)

Data Source Explorer

- Database Connections
  - benchmark3
  - sales (Informix 11.1)
    - sales
      - Accelerators
        - SALESACL (Fully Operational)
          - sales [Enabled]
      - Roles
      - Schemas
      - Users

sales

Estimated size: 2.0 MB - Not all tables are included

customers N/A
customer_id
first_name
last_name
middle_name
address
city
state
zip

sales_fact N/A
transaction_year
transaction_id
store_id
customer_id
date_id
product_id
promotion_id
product_cost
product_qty
unit_price_paid
total_price_paid
transaction_dttm
load_id

time_dim 1.8 MB
date_id
date
day
month
year
datestring
dayofweekname
dayofweekname_abbr
monthname
dayofweek
dayofyear
week
quarter
weekend

stores_dim 0.2 MB
store_id
storename
street
city
state
postalcode
country
phone
statename
region
regionname
division
divisionname
countryname
size_sq_ft
coffee_shop
date_opened
website

books N/A
product_id
book_id
language
title
price
author_id
last_modified
revision
published

promotion_dim 27 KB
promotion_id
promotion_name
start_date
end_date
discount_percent
website

authors N/A
author_id
serial_id
name
last_modified
revision

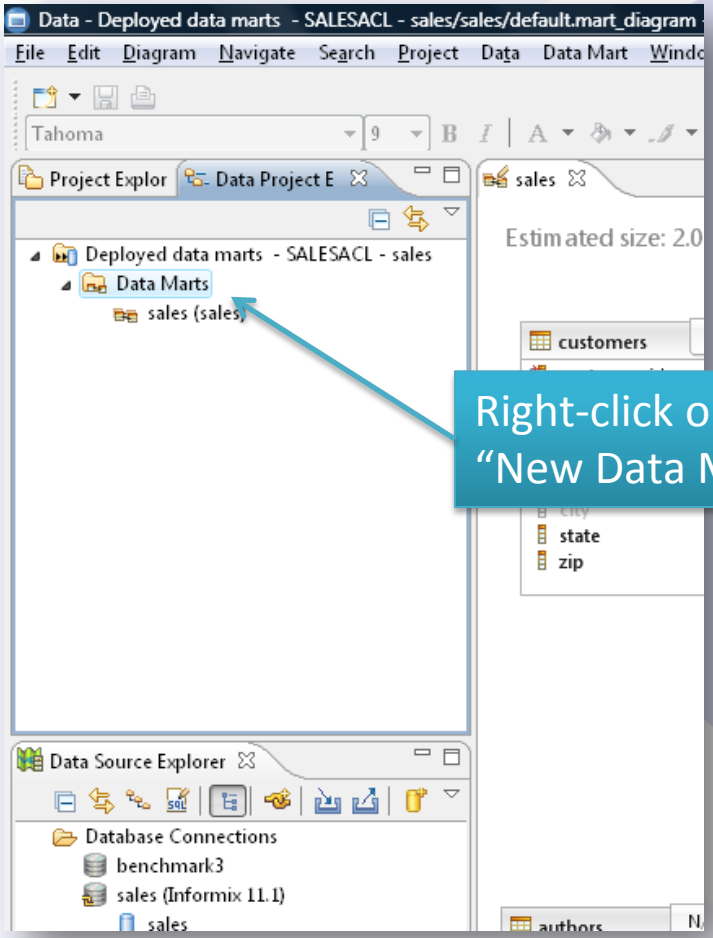
Properties | Capture SQL for Tuning | SQL Category | Monitor List | SQL Results | Workload List

1 items selected

# Demo

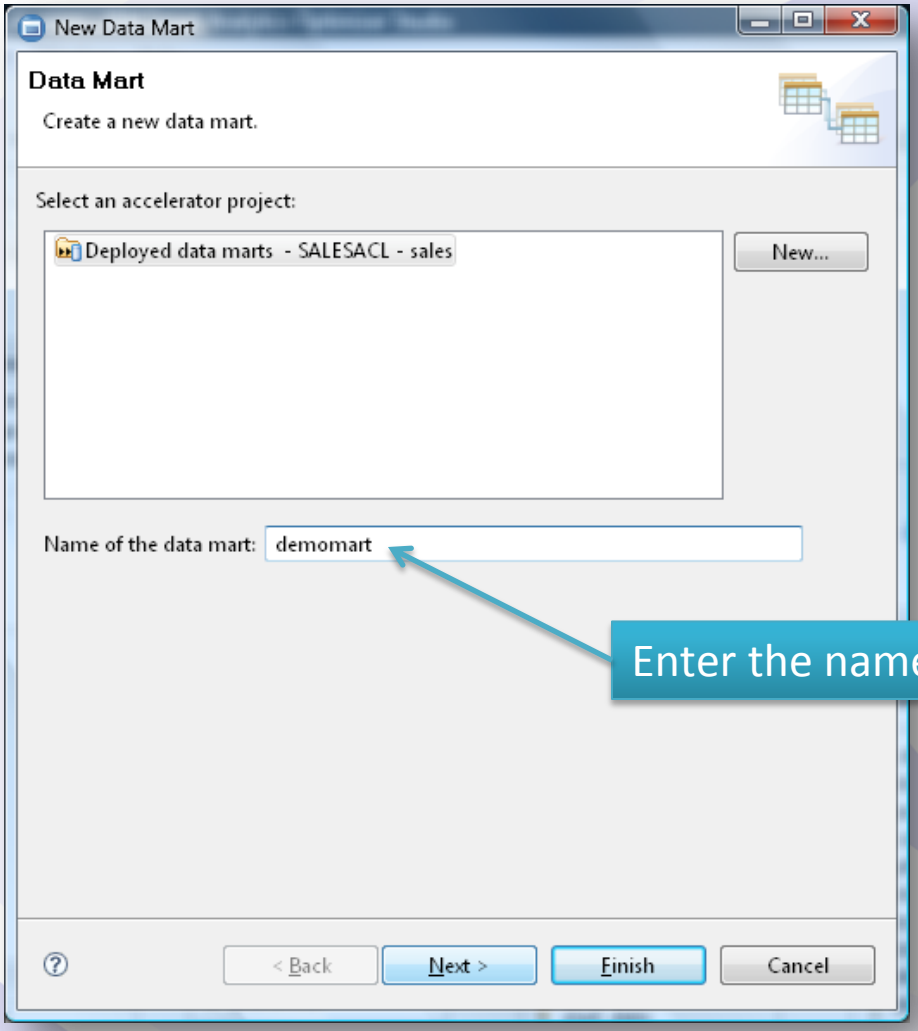
- Create a simple Data Mart using Studio

# Example - Creating a Data Mart



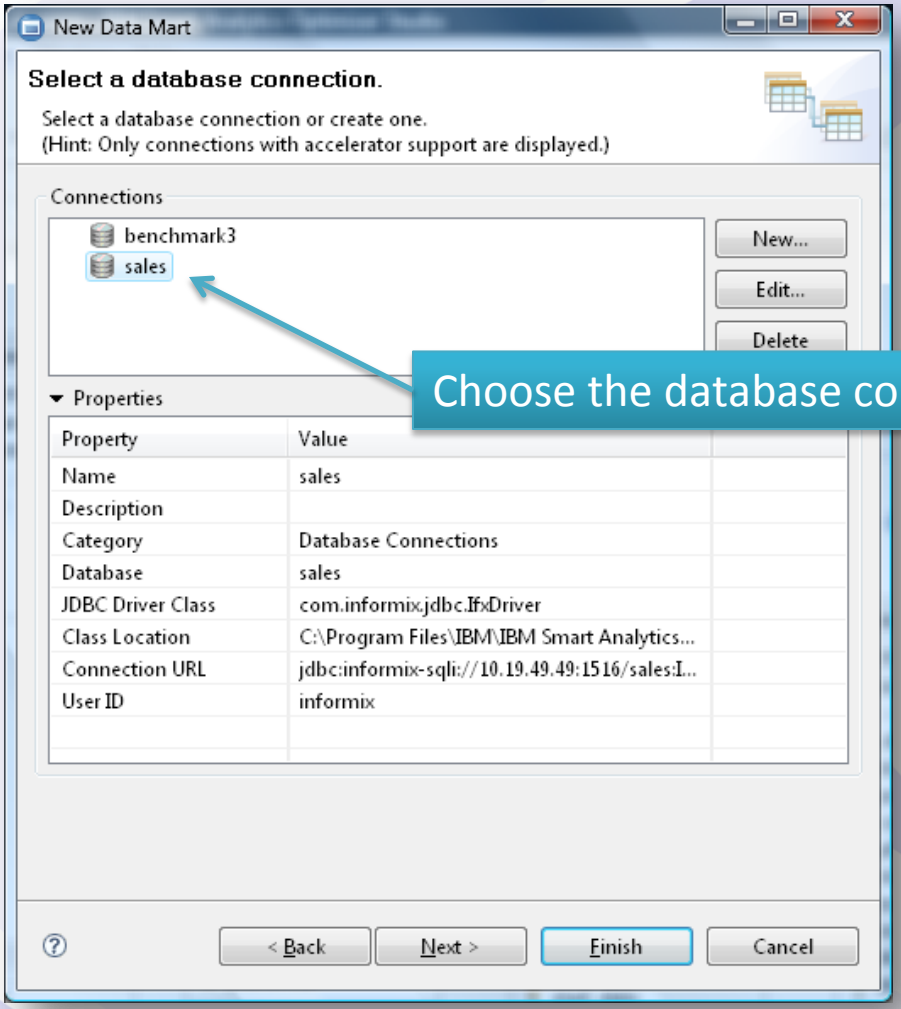

Right-click on Data Marts and select "New Data Mart"

# Example - Creating a Data Mart



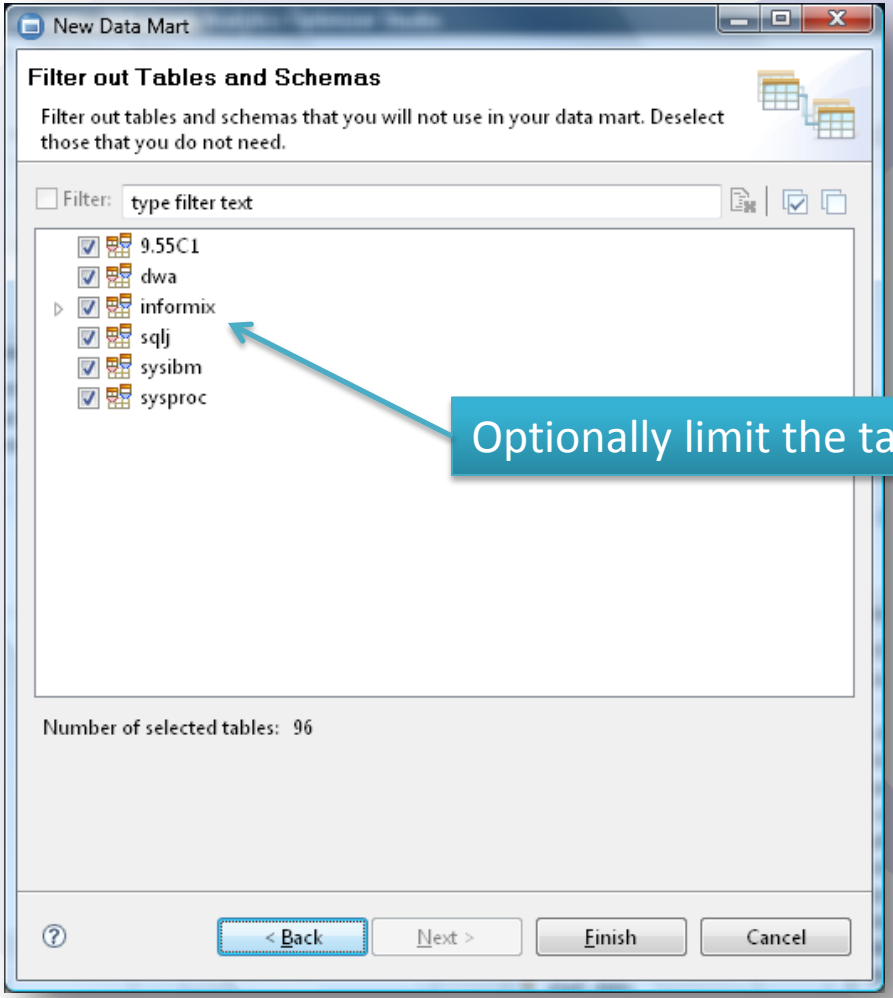

Enter the name of the new Data Mart

# Example - Creating a Data Mart



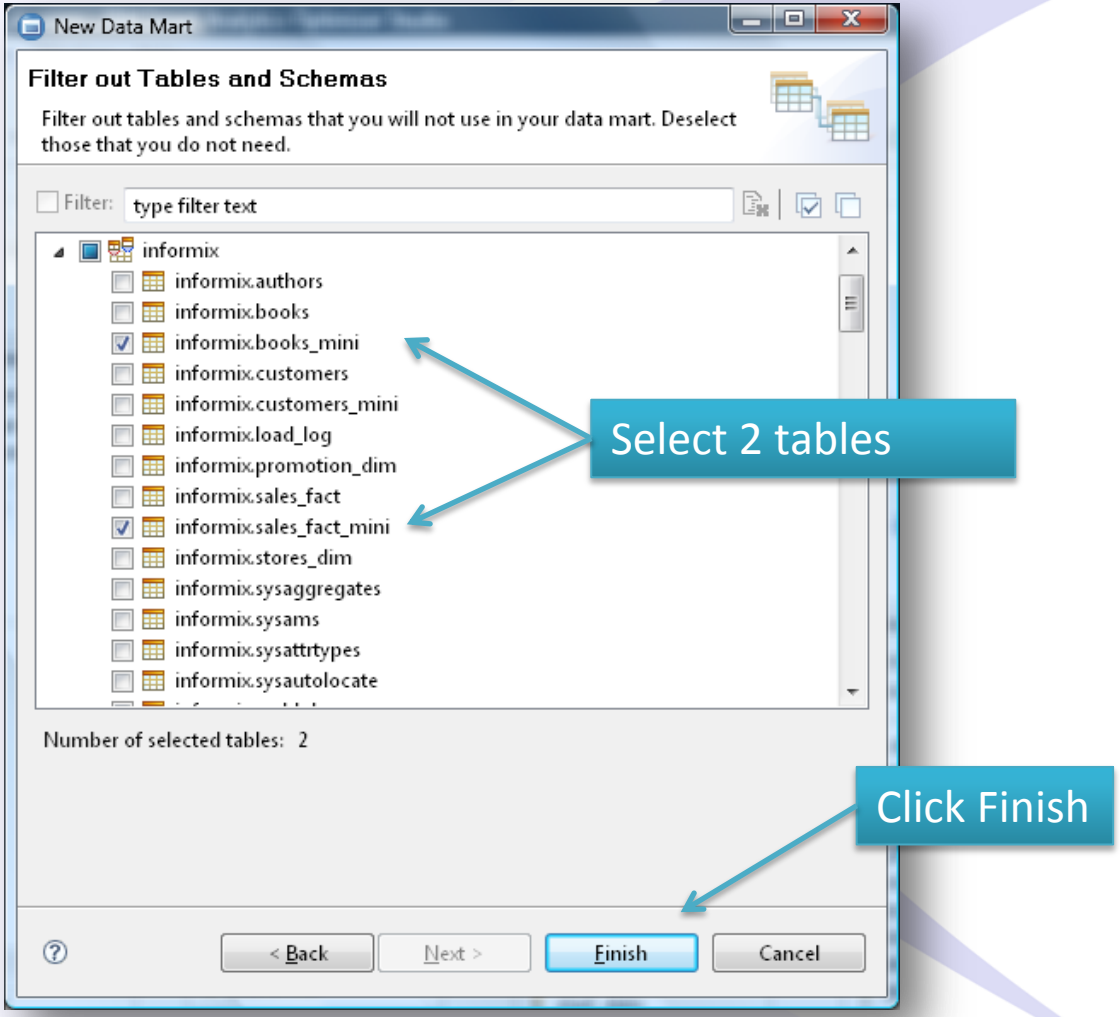

Choose the database connection to use

# Example - Creating a Data Mart




Optionally limit the tables to include

# Example - Creating a Data Mart



# Example - Creating a Data Mart

Data - Deployed data marts - SALESACL - sales/demomart/default.mart_diagram - IBM Smart Analytics Optimizer Studio

File Edit Diagram Navigate Search Project Data Data Mart Window Help

Project Explor | Data Project E

Deployed data marts - SALESACL - sales
- Data Marts
  - demomart (sales)
  - sales (sales)

Data Source Explorer

Database Connections
- benchmark3
- sales (Informix 11.1)
  - sales
    - Accelerators
      - SALESACL (Fully Operational)
        - sales [Enabled]
    - Roles
    - Schemas
    - Users

sales | *demomart

Estimated size: 0.0 B

Palette
- Data Mart
  - Table
  - 1:n
  - n:m

New Data Mart is now listed

Choose the tables. Double-click or Drag Table to workspace

Properties | Capture SQL for Tuning | SQL Category | Monitor List | SQL Results | Workload List

<Data Mart> demomart

General | Tables | Validation | Space Required

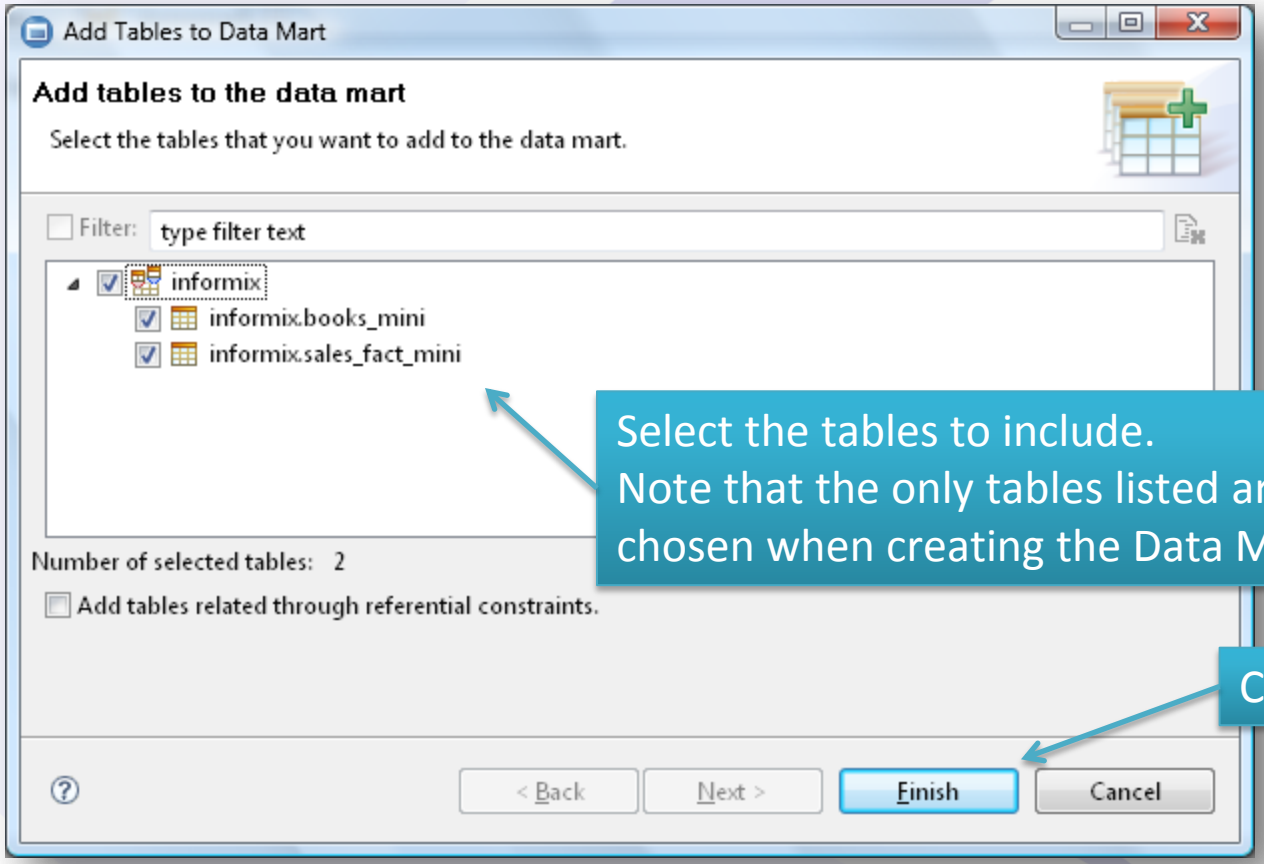
Set Database... | Refresh Database Cache | Validate | Deploy...

Name: demomart

Database: sales

Estimated size: 0.0 B

# Example - Creating a Data Mart




Select the tables to include.
Note that the only tables listed are those chosen when creating the Data Mart

Click Finish

# Example - Creating a Data Mart

Data - Deployed data marts - SALESACL - sales/demomart/default.mart_diagram - IBM Smart Analytics Optimizer Studio

File Edit Diagram Navigate Search Project Data Data Mart Window Help

Project Explor | Data Project E

- Deployed data marts - SALESACL - sales
  - Data Marts
    - demomart (sales)
    - sales (sales)

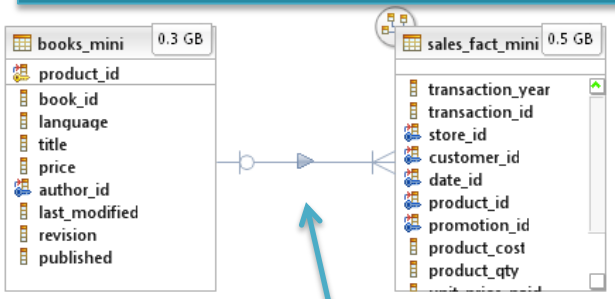
Data Source Explorer

- Database Connections
  - benchmark3
  - sales (Informix 11.1)
    - sales
      - Accelerators
        - SALESACL (Fully Operational)
          - sales [Enabled]
      - Roles
      - Schemas
      - Users

sales | *demomart

Estimated size: 0.8 GB

Selected tables are shown




Join is included automatically

Palette

Data Mart

- Table
- 1:n
- n:m

Properties | Workload List

<Mart> demomart (sales)

| General | Property | Value |
|---|---|---|
| | Info | |
| | derived | false |
| | editable | true |
| | last modified | February 26, 2012 1:11:39 PM |
| | linked | false |
| | location | C:\Users\Mike\IBM\Smart Analytics Optimizer Studio 1.1\workspace\Deployed data marts - SALESA. |

1 items selected

# Example - Creating a Data Mart

Data - Deployed data marts - SALESACL - sales/demomart/default.mart_diagram - IBM Smart Analytics Optimizer Studio

File Edit Diagram Navigate Search Project Data Data Mart Window Help

Project Explor | Data Project E

- Deployed data marts - SALESACL - sales
  - Data Marts
    - demomart (sales)
    - sales (sales)

sales | *demomart

books_mini 8.6 MB
- product_id
- book_id
- language
- title
- price
- author_id
- last_modified
- revision
- published

sales_fact_mini 9.7 MB
- transaction_year
- transaction_id
- store_id
- customer_id
- date_id
- product_id
- promotion_id
- product_cost
- product_qty
- unit_price_paid
- total_price_paid
- transaction_dttm

Palette

By default, ALL columns are selected for acceleration

Data Source Explorer

- Database Connections
  - benchmark3
  - sales (Informix 11.1)
    - sales

Properties | Capture SQL for Tuning | SQL Category | Monitor List | SQL Results | Workload List

<Table> informix.books_mini

General | Columns | References | Space Required

Select the columns to accelerate.

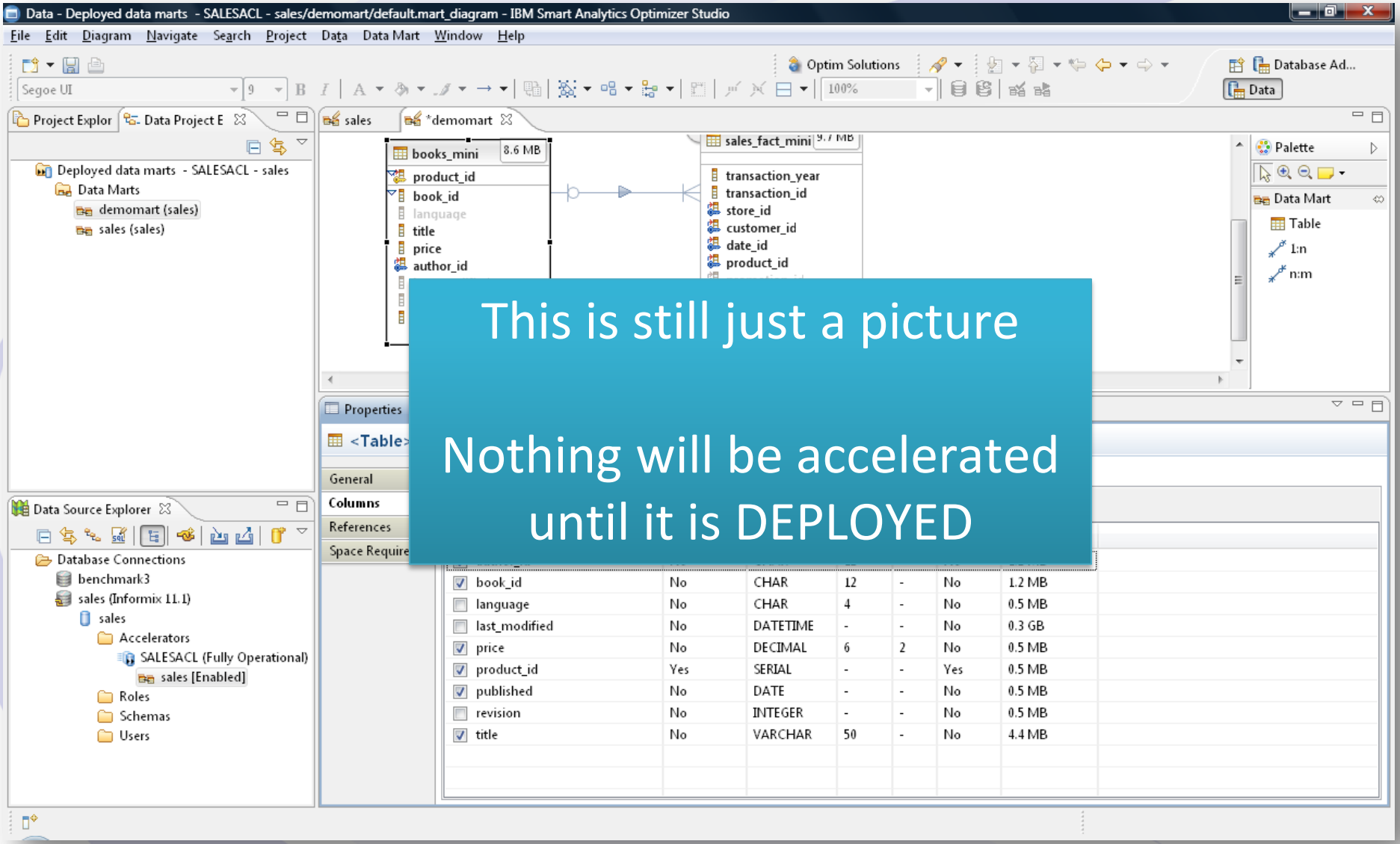
Filter: type filter text

| Name | Primary Key | Data Type | Length | Scale | Not Null | Estimated Size |
|---|---|---|---|---|---|---|
| author_id | No | CHAR | 12 | - | No | 1.2 MB |
| book_id | No | CHAR | 12 | - | No | 1.2 MB |
| language | No | CHAR | 4 | - | No | 0.5 MB |
| last_modified | No | DATETIME | - | - | No | 0.3 GB |
| price | No | DECIMAL | 6 | 2 | No | 0.5 MB |
| product_id | Yes | SERIAL | - | - | Yes | 0.5 MB |
| published | No | DATE | - | - | No | 0.5 MB |
| revision | No | INTEGER | - | - | No | 0.5 MB |
| title | No | VARCHAR | 50 | - | No | 4.4 MB |

Choose the columns to exclude

Size shown per column

# Example - Creating a Data Mart




This is still just a picture

Nothing will be accelerated until it is DEPLOYED

# Example - Creating a Data Mart

Data - Deployed data marts - SALESACL - sales/demomart/default.mart_diagram - IBM Smart Analytics Optimizer Studio

File Edit Diagram Navigate Search Project Data Data Mart Window Help

Project Explor | Data Project E

- Deployed data marts - SALESACL - sales
  - Data Marts
    - demomart (sales)
    - sales (sales)

sales | *demomart

books_mini 8.6 MB
- product_id
- book_id
- language
- title
- price
- author_id
- last_modified
- revision
- published

sales_fact_mini 9.7 MB
- transaction_year
- transaction_id
- store_id
- customer_id
- date_id
- product_id
- promotion_id
- product_cost
- product_qty
- unit_price_paid
- total_price_paid
- transaction_dttm

Palette
- Data Mart
- Table
- 1:n
- n:m

SQL Category | Monitor List | SQL Results | Workload List

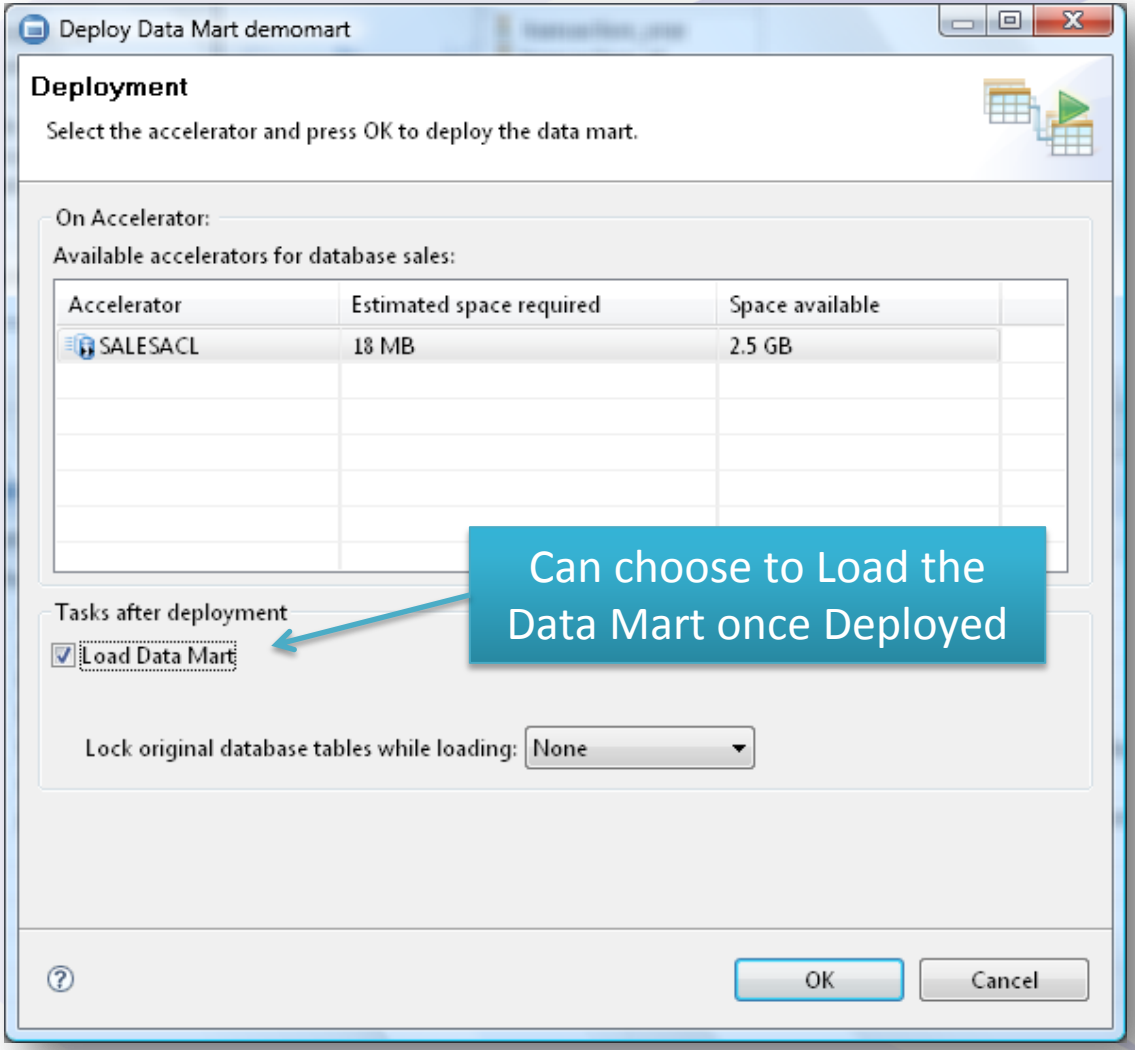
Right-click on the new Data Mart and choose: "Deploy Data Mart"

Data Source Explorer
- Database Connections
  - benchmark3
  - sales (Informix 11.1)
    - sales
      - Accelerators
        - SALESACL (Fully Operational)
          - sales [Enabled]
      - Roles
      - Schemas
      - Users

References | Space Required

| Name | Primary Key | Data Type | Length | Scale | Not Null | Estimated Size |
|---|---|---|---|---|---|---|
| author_id | No | CHAR | 12 | - | No | 1.2 MB |
| book_id | No | CHAR | 12 | - | No | 1.2 MB |
| language | No | CHAR | 4 | - | No | 0.5 MB |
| last_modified | No | DATETIME | - | - | No | 0.3 GB |
| price | No | DECIMAL | 6 | 2 | No | 0.5 MB |
| product_id | Yes | SERIAL | - | - | Yes | 0.5 MB |
| published | No | DATE | - | - | No | 0.5 MB |
| revision | No | INTEGER | - | - | No | 0.5 MB |
| title | No | VARCHAR | 50 | - | No | 4.4 MB |

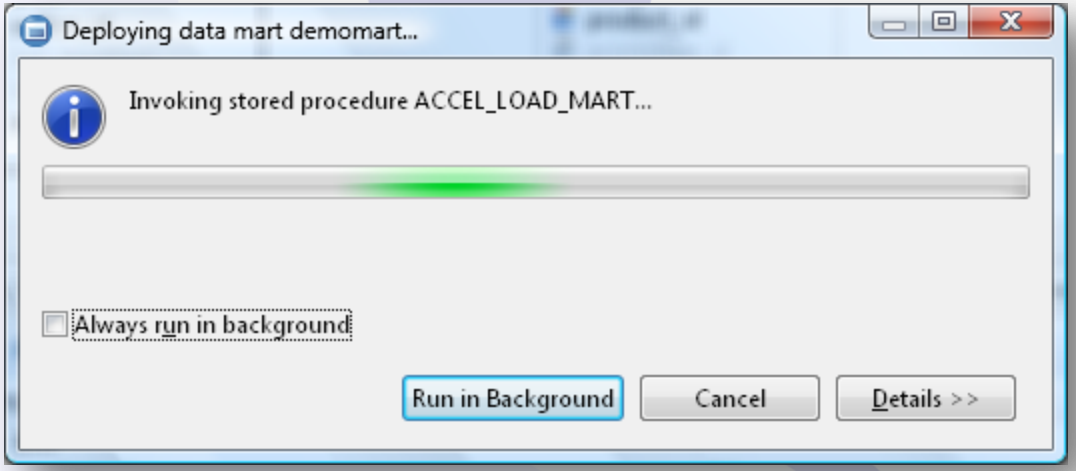
# Example - Creating a Data Mart




Can choose to Load the Data Mart once Deployed

# Example - Creating a Data Mart

Data Mart will be Deployed and Loaded (if chosen)



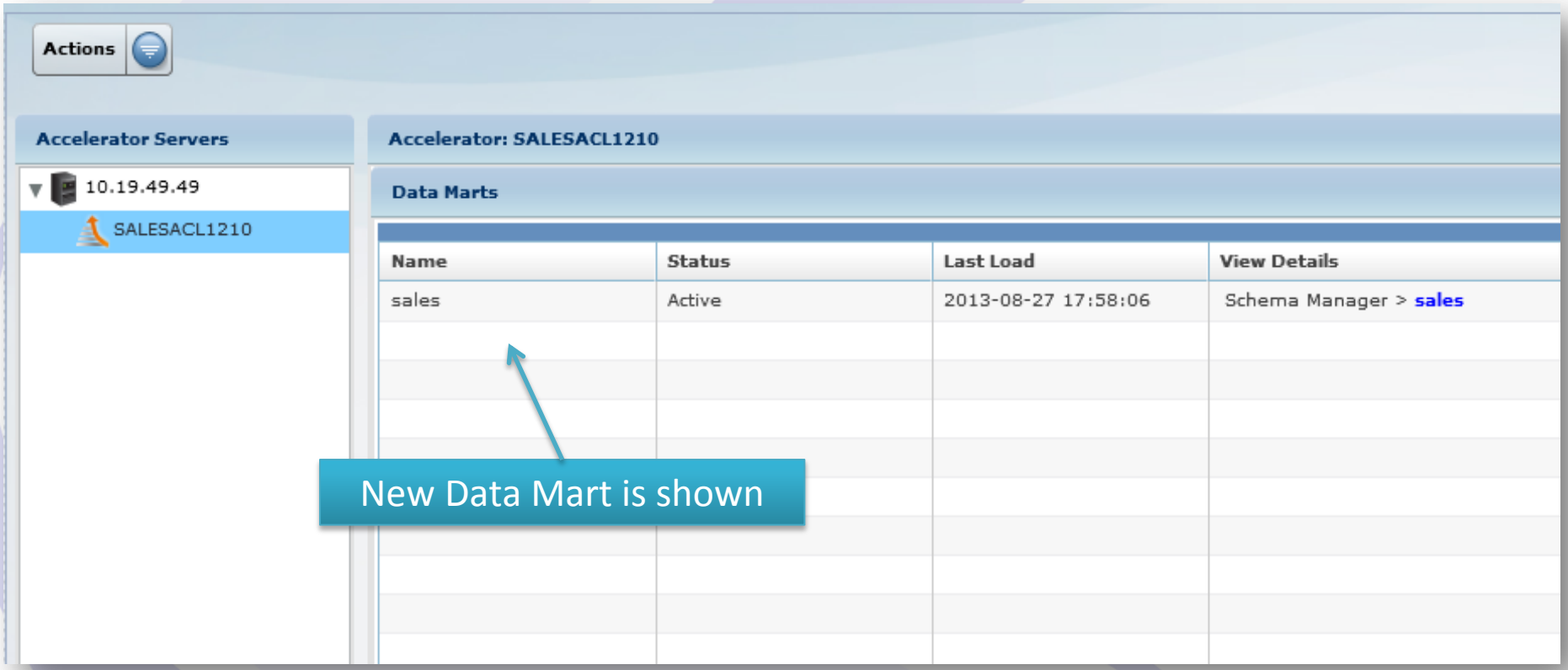
# Example - Creating a Data Mart

Deployed Data Mart is now listed beneath the Accelerator

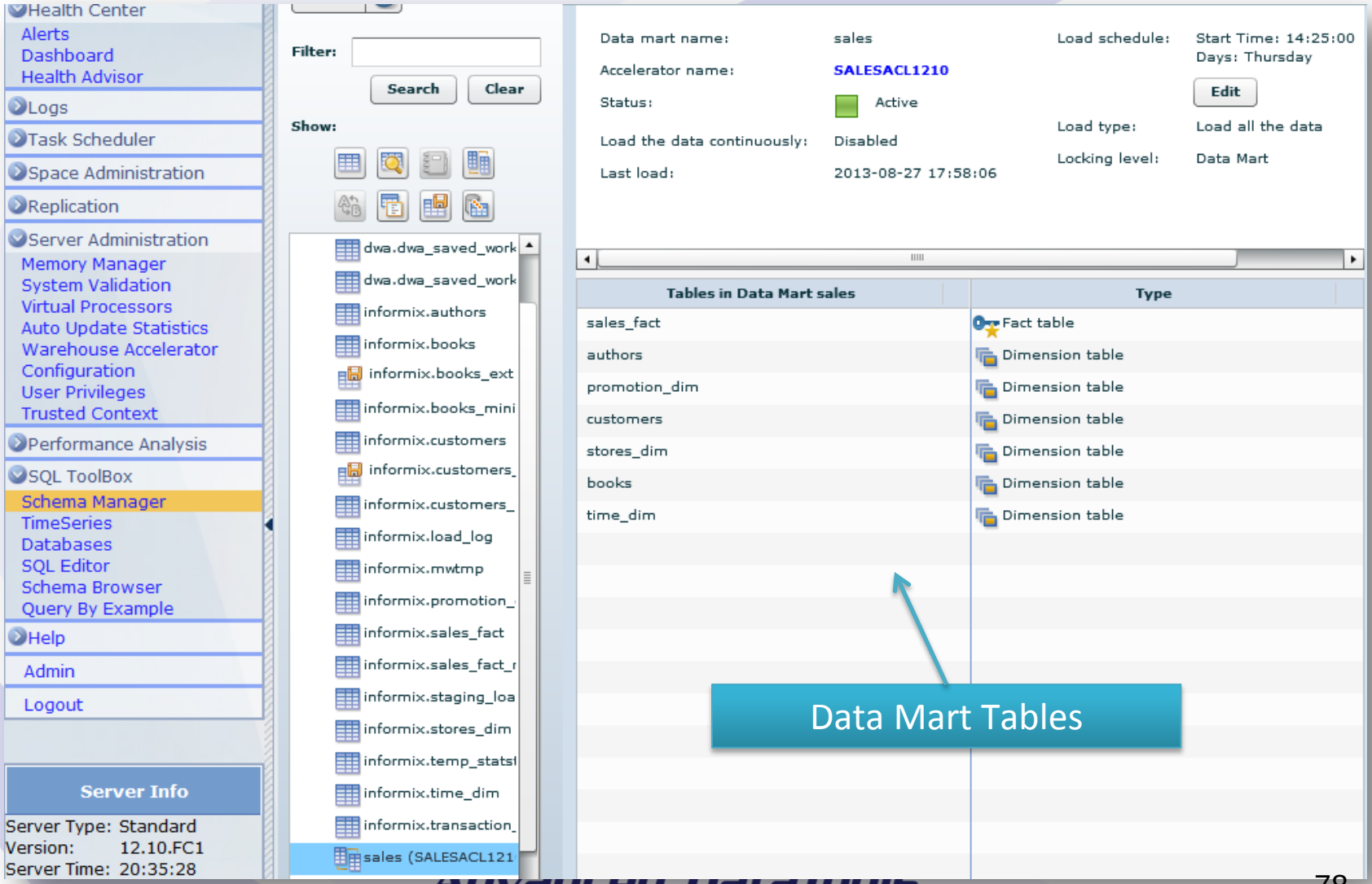
# Loading the Data Mart

- Load the Data Mart
  - Studio
  - OAT
  - java loadMart SALESACL1210 sales NONE
  - execute function ifx_loadMart ('SALESACL1210', 'sales', 'NONE');

# Loading the Data Mart - OAT

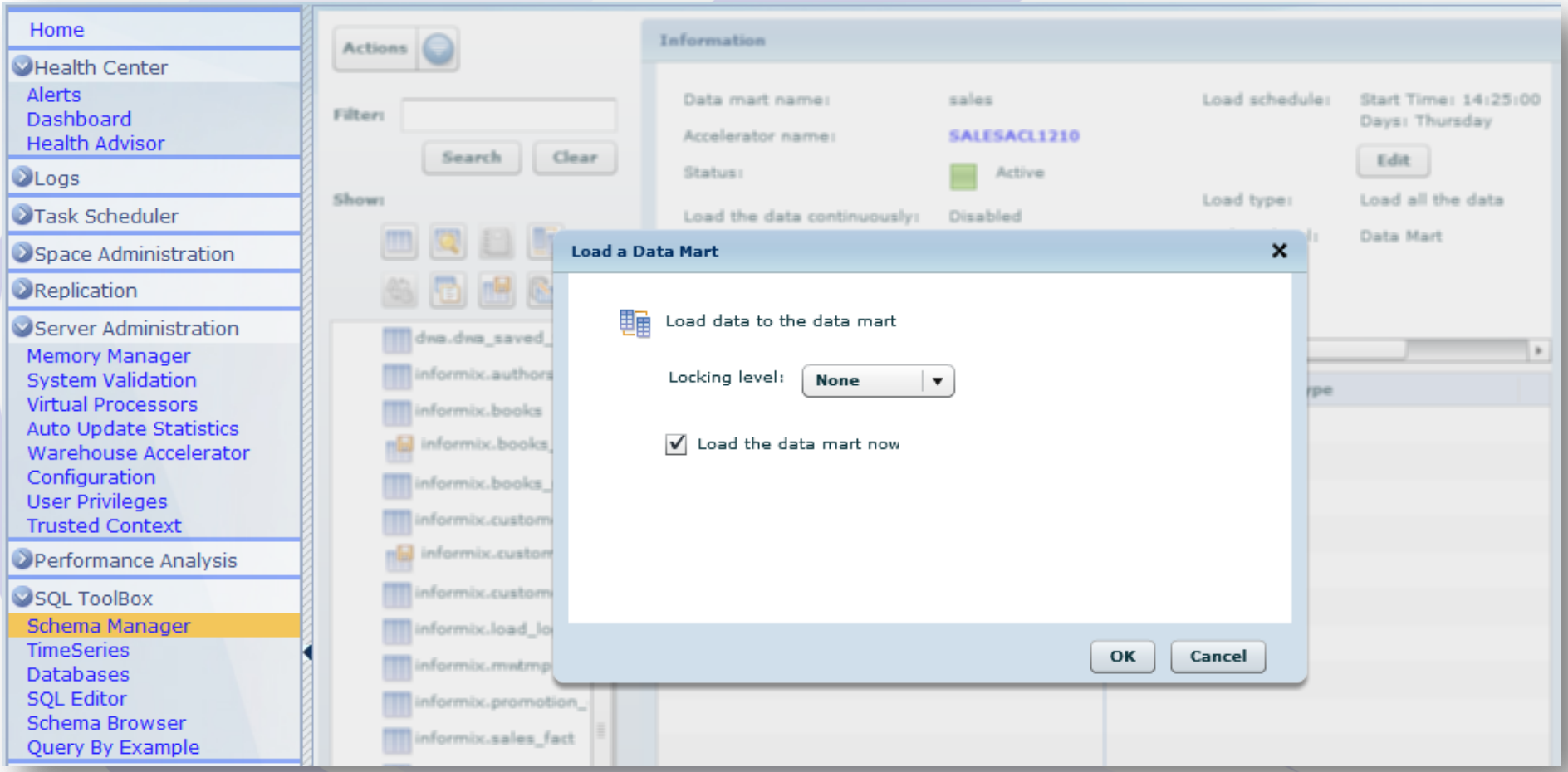


# Loading the Data Mart - OAT



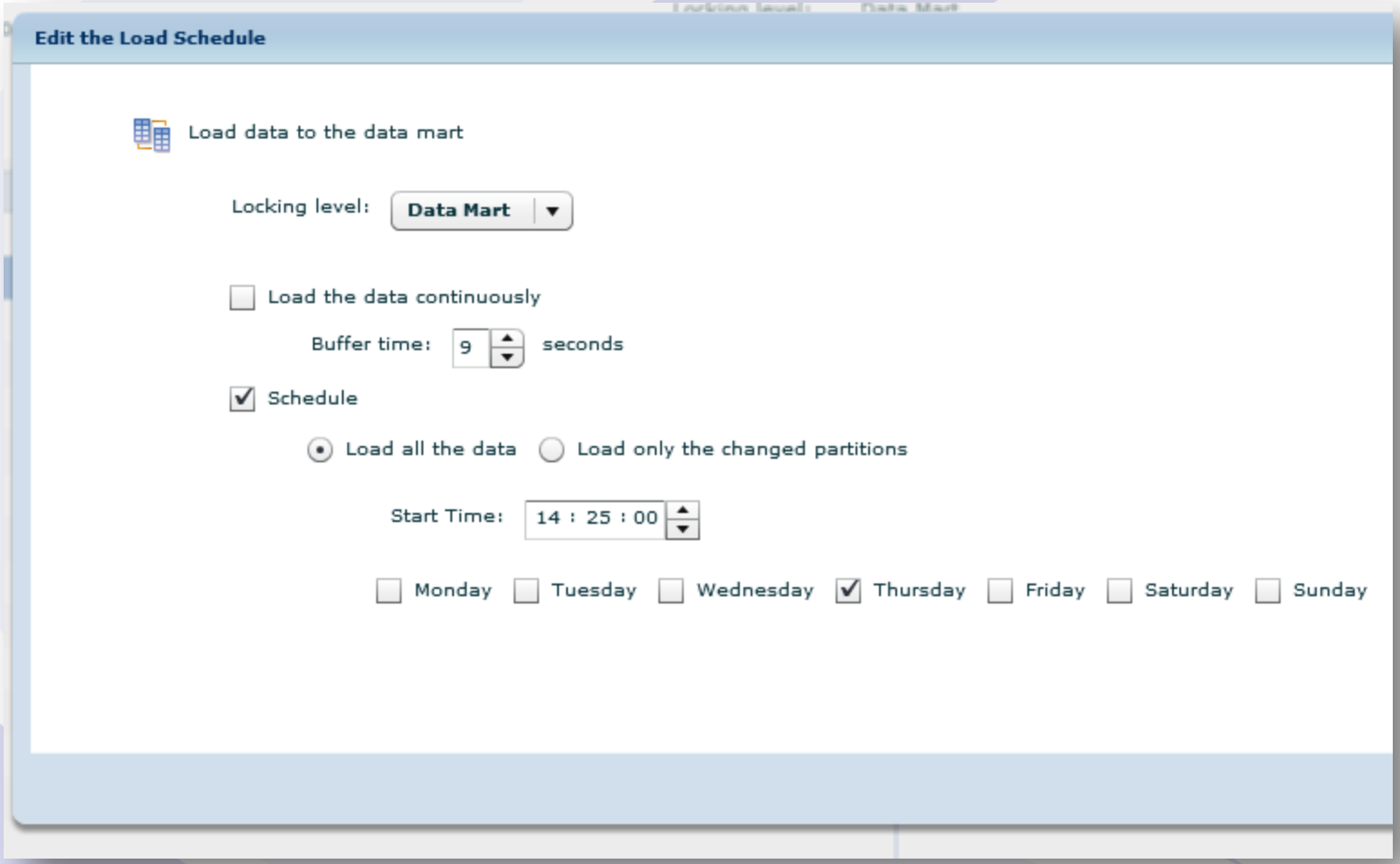
# Loading the Data Mart - OAT

## One-time Load of the Data Mart



# Loading the Data Mart - OAT

## Schedule the Data Mart Load



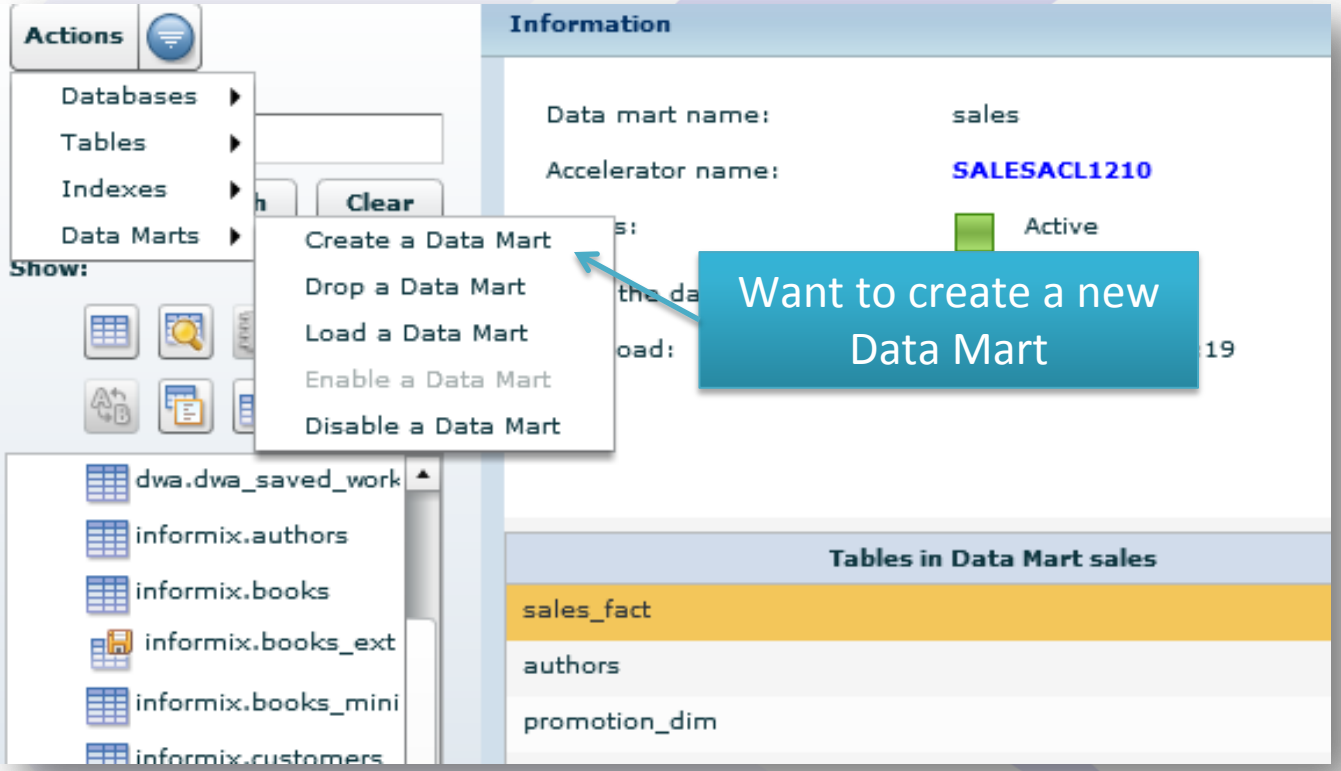
# Creating a Data Mart using Workload Analysis in OAT

- Examine sample SQL to generate a Data Mart
  - Capture some SQL
  - Start monitoring through OAT
  - Run the SQL (set explain on avoid_execute)
- Can also do this
- ANY SQL that uses the tables/columns in the generated Data Mart will be accelerated – not just those statements in the sample SQL

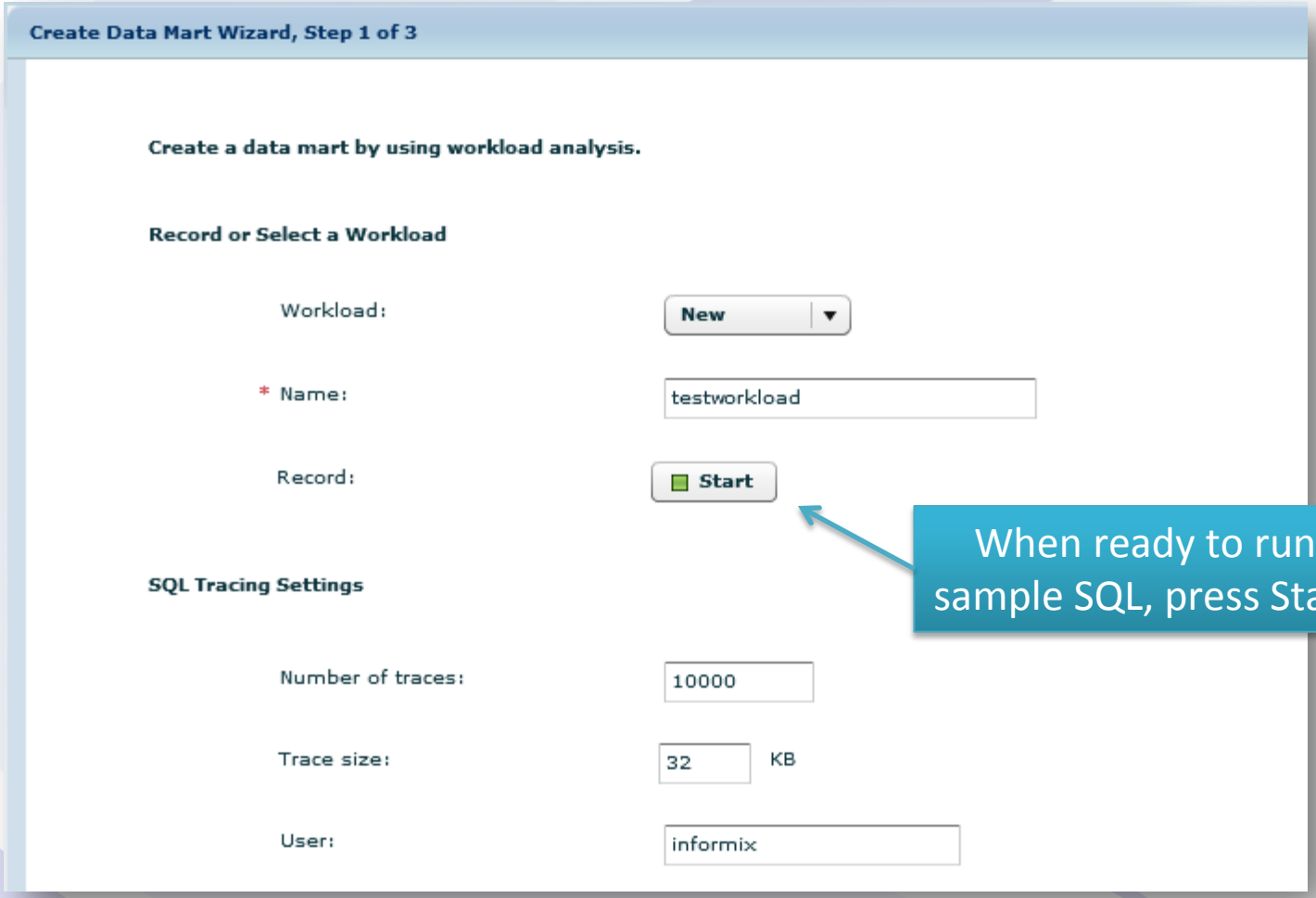
# Demo

- Demonstrate the creation of a Data Mart with Workload Analysis through OAT

# Workload Analysis in OAT

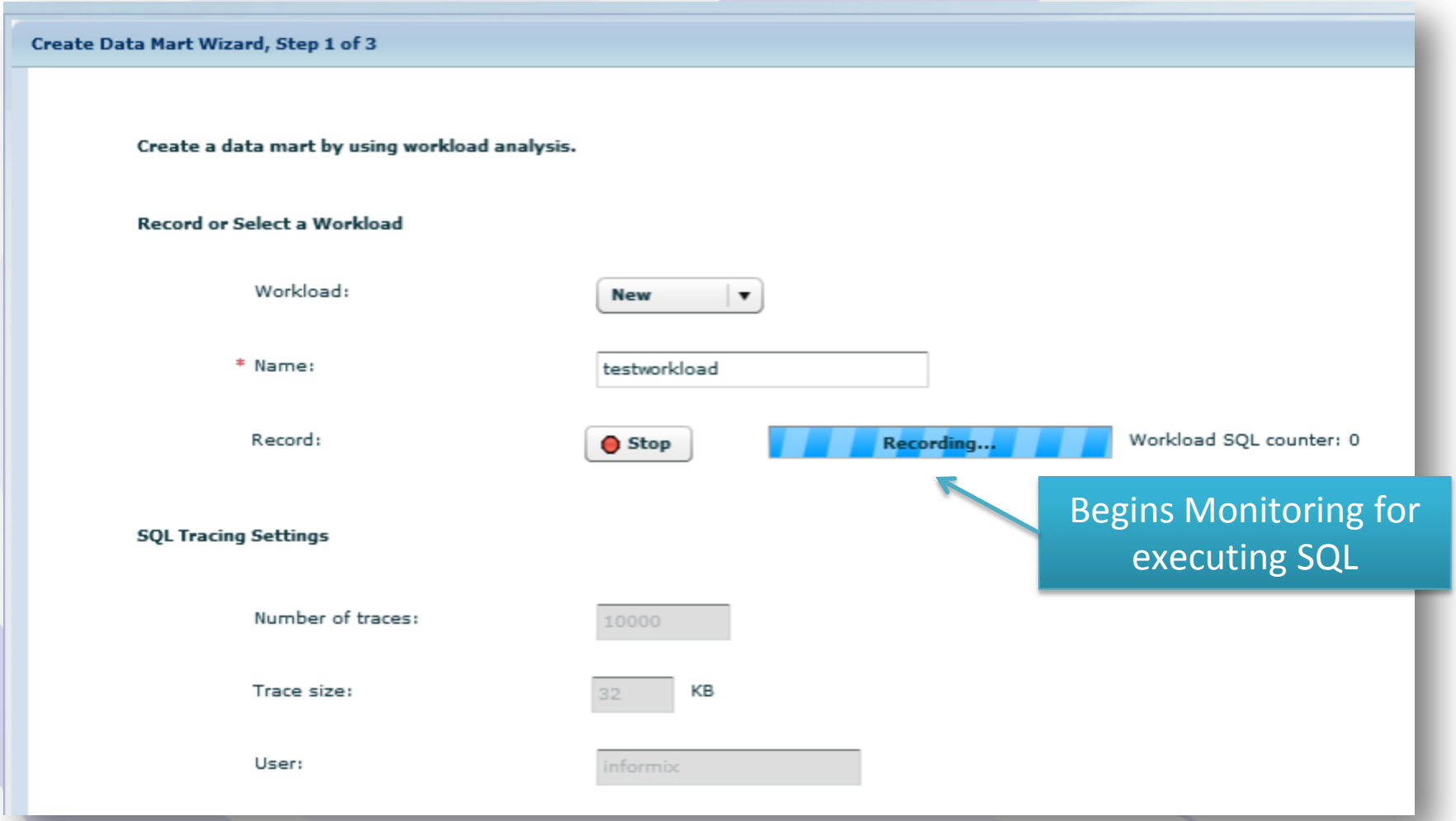


# Workload Analysis in OAT




When ready to run sample SQL, press Start

# Workload Anlaysis in OAT




Begins Monitoring for executing SQL

# Workload Analysis in OAT

babe - 10.19.49.49

```
babe1210:~/salesdb/accelerator>dbaccess sales sales_workload.sql

Database selected.


Explain set.


transaction_year            (sum)

No rows found.


transaction_year date                revenue

No rows found.
```

Run sample SQL

set explain on avoid_execute allows SQL to run quickly

# Workload Analysis in OAT

# Workload Analysis in OAT

Create Data Mart Wizard, Step 2 of 3

Captured statements are displayed

**Review the SQL Statements**

The number of statements that can be accelerated: 12

Search Clear

| SQL ID | Workload Statement | Can Accelerate |
|---|---|---|
| 13 | select<br>f.transaction_year,<br>t.quarter, | ✓ |
| 12 | select c.last_name,<br>c.first_name,<br>t.date, | ✓ |
| 11 | select c.last_name,<br>c.first_name,<br>t.date, | ✓ |
| 10 | select s.store_id,<br>t.year,<br>t.quarter, | ✓ |
| 9 | select<br>case<br>when s.state = c.state then "IN_STATE" | ✓ |
| 8 | select<br>t.dayofweek, t.dayofweekname, sum(f.product_qty)<br>from sales_fact f, | ✓ |
| 7 | select f.promotion_id,<br>p.promotion_name,<br>sum((f.product_cost * f.product_qty)) full_price, | ✓ |
| 6 | select f.transaction_year, t.quarter, s.regionname, s.divisionname, sum(product_qty)<br>from authors a,<br>books b, | ✓ |

12 total items 25 Per Page 1 of 1

# Workload Analysis in OAT

**Create Data Mart Wizard, Step 3 of 3**

**Create the Data Mart**

Ready to create the Data Mart

Database name: sales

* Data mart name: workloadmart

* Accelerator: **SALESACL1210 (10.19.49.49)**

Select an accelerator

SALESACL1210 (10.19.49.49)

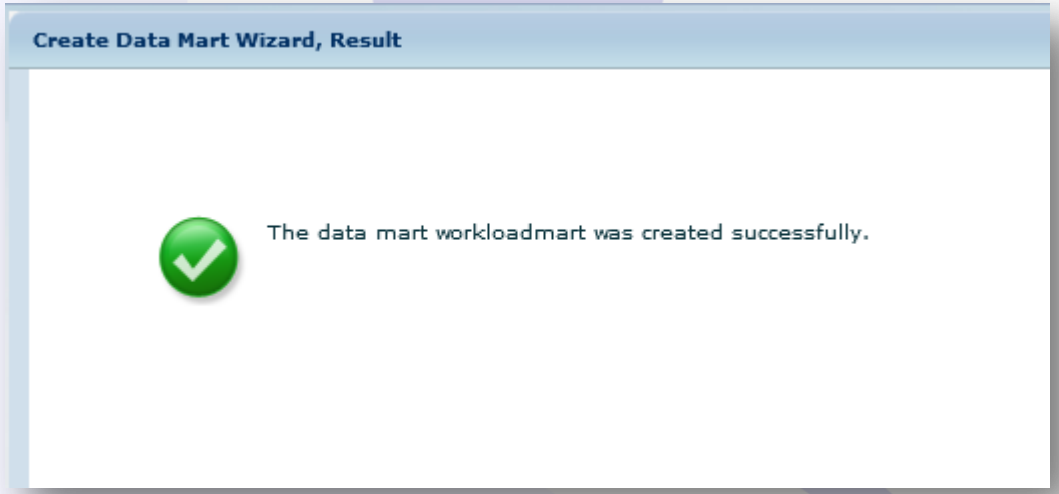
☐ Load data after the data mart is created

Locking level: None

☐ Keep the recording of the workload trace after the data mart is created

# Workload Analysis in OAT

The new Data Mart is created from the sample SQL



# Workload Analysis in OAT

Actions

Accelerator Servers

10.19.49.49

SALESACL1210

Accelerator: SALESACL1210

Data Marts

| Name | Status | Last Load | View Details |
|---|---|---|---|
| demomart | Active | 2013-08-28 14:58:07 | Schema Manager > **demomart** |
| sales | Active | 2013-08-27 23:42:19 | Schema Manager > **sales** |
| workloadmart | Load pending | | Schema Manager > **workloadmart** |

New Data Mart is listed in OAT

Note that IWA will try and accelerate queries that match the tables/columns in the Data Mart generated from the workload SQL – not just statements matching the workload SQL

# Show Data Marts

```
>java listMarts SALESACL1210
listMarts SALESACL1210
-----------------------------------------------------------------------
-------
result message:
<?xml version="1.0" encoding="UTF-8" ?> <dwa:messageOutput
xmlns:dwa="http://www.ibm.com/xmlns/prod/dwa" version="1.0"> <message
severity="informational" reason-code="AQT10000I"><text>The operation was
completed successfully.</text></message></dwa:messageOutput>
-----------------------------------------------------------------------
-------
mart list:
<?xml version="1.0" encoding="UTF-8" ?>
<dwa:martList xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:dwa="http://www.ibm.com/xmlns/prod/dwa" version="1.0">
<mart name="demomart" status="Active" memoryConsumptionInMB="12"
lastLoadTimestamp="2013-08-28T14:58:07.115094Z" />
<mart name="sales" status="Active" memoryConsumptionInMB="25178"
lastLoadTimestamp="2013-08-27T23:42:19.023442Z" />
<mart name="workloadmart" status="LoadPending" />
</dwa:martList>
```

# Data Marts in Studio

Data - SALESACL1210 - sales/demomart/default.mart_diagram - IBM Smart Analytics Optimizer Studio

File Edit Diagram Navigate Search Project Data Mart Window Help

Project Explorer | Data Project Explorer

- SALESACL1210 - sales
  - Data Marts
    - demomart (sales)
    - sales (sales)

Data Source Explorer

- Database Connections
  - sales (Informix 11.1)
    - sales
      - Accelerators
        - SALESACL1210 (Fully Operational)
          - demomart [Enabled]
          - sales [Enabled]
          - workloadmart [Disabled]
      - Roles
      - Schemas
      - Users

1 items selected

sales | *demomart

Estimated size: 18 MB

books_mini 8.6 MB
- product_id
- book_id
- language
- title
- price
- author_id
- last_modified
- revision
- published

sales_fact_mini 9.7 MB
- transaction_year
- transaction_id
- store_id
- customer_id
- date_id
- product_id
- promotion_id
- product_cost
- product_qty
- unit_price_paid
- total_price_paid
- transaction_dttm

Properties | Capture SQL for Tuning | SQL Category | Workload List | Monitor List | SQL Results

<Accelerator> SALESACL1210

General | Data Marts | Software Version

Enable | Disable | Deploy... | Open... | Drop | Load...

| Name | Status | Enabled | Last Load | Last Update |
|---|---|---|---|---|
| demomart | Active | Yes | N/A | None |
| sales | Active | Yes | N/A | None |
| workloadmart | *Load pending* | No | N/A | None |

New Data Mart is now listed for the Accelerator

# Data Marts in Studio

- Unload the XML definition of the new Data Mart

```
EXECUTE FUNCTION
lotofile(ifx_getMartdef('SALESACL1210',
'workloadmart'),'workloadmart.xml!','client');


(expression)  workloadmart.xml


1 row(s) retrieved.
```

- Save XML to local drive

# Data Marts in Studio


```
<?xml version="1.0" encoding="utf8" standalone="no" ?><dwa:martModel xmlns:dwa="http://www.ibm.com/xmlns/prod/dwa"
version="1.0" xmlns:xs="http://www.w3.org/2001/XMLSchema">
<mart name="workloadmart">
<table consistencyToken="0x00760000" isFactTable="false" name="authors" schema="informix">

<column ccsid="819" charEncoding="SBCS" dataType="CHAR" name="author_id" nullable="false" precision="12"/>
<column ccsid="819" charEncoding="SBCS" dataType="VARCHAR" name="name" nullable="true" precision="80"/>
</table>
<table consistencyToken="0x007A0000" isFactTable="false" name="books" schema="informix">

<column ccsid="819" charEncoding="SBCS" dataType="CHAR" name="author_id" nullable="true" precision="12"/>
<column dataType="DECIMAL" name="price" nullable="true" precision="6" scale="2"/>
<column dataType="INTEGER" name="product_id" nullable="false"/>
<column ccsid="819" charEncoding="SBCS" dataType="VARCHAR" name="title" nullable="true" precision="50"/>
</table>
<table consistencyToken="0x00780000" isFactTable="false" name="customers" schema="informix">

<column dataType="INTEGER" name="customer_id" nullable="false"/>
<column ccsid="819" charEncoding="SBCS" dataType="CHAR" name="first_name" nullable="true" precision="20"/>
<column ccsid="819" charEncoding="SBCS" dataType="CHAR" name="last_name" nullable="true" precision="20"/>
<column ccsid="819" charEncoding="SBCS" dataType="CHAR" name="state" nullable="true" precision="2"/>
</table>
<table consistencyToken="0x00720000" isFactTable="false" name="promotion_dim" schema="informix">
…
```

# Data Marts in Studio

Data - SALESACL1210 - sales/demomart/default.mart_diagram - IBM Smart Analytics Optimizer Studio

File Edit Diagram Navigate Search Project Data Mart Window Help

Project Explorer | Data Project Explorer

- SALESACL1210 - sales
  - Data Marts
    - demomart (sales)
    - sales (sales)

Right-click and choose "Import"

Data Source Explorer

- Database Connections
  - sales (Informix 11.1)
    - sales
      - Accelerators
        - SALESACL1210 (Fully Operational)
          - demomart [Enabled]
          - sales [Enabled]
          - workloadmart [Disabled]
      - Roles
      - Schemas
      - Users

sales | *demomart

Estimated size: 18 MB

books_mini 8.6 MB
- product_id
- book_id
- language
- ce
- thor_id
- t_modified
- ision
- published

sales_fact_mini 9.7 MB
- transaction_year
- transaction_id
- store_id
- customer_id
- date_id
- product_id
- promotion_id
- product_cost
- product_qty
- unit_price_paid
- total_price_paid
- transaction_dttm

Properties | Capture SQL for Tuning | SQL Category | Workload List | Monitor List | SQL Results

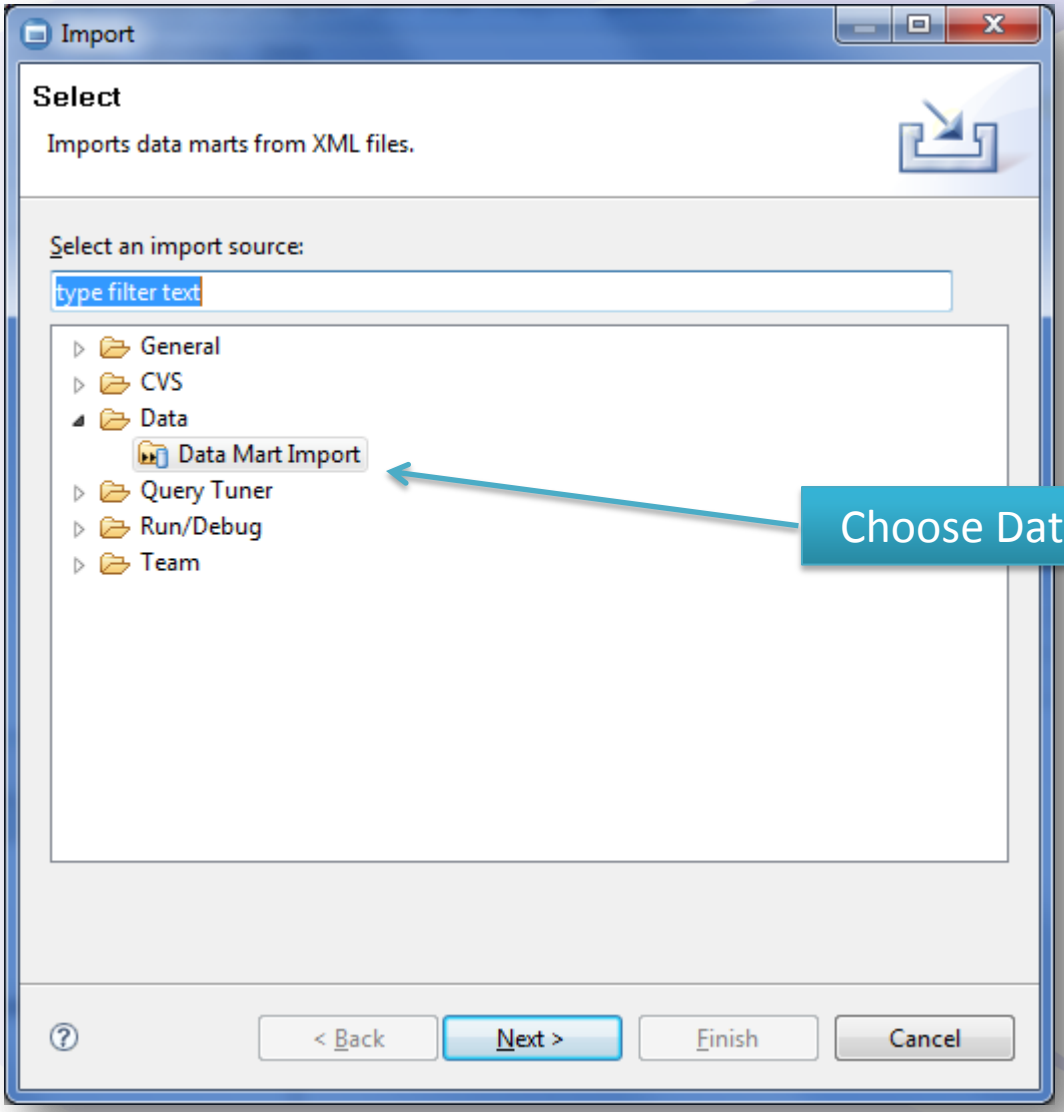
<Accelerator> SALESACL1210

General | Data Marts | Software Version

Enable | Disable | Deploy... | Open... | Drop | Load...

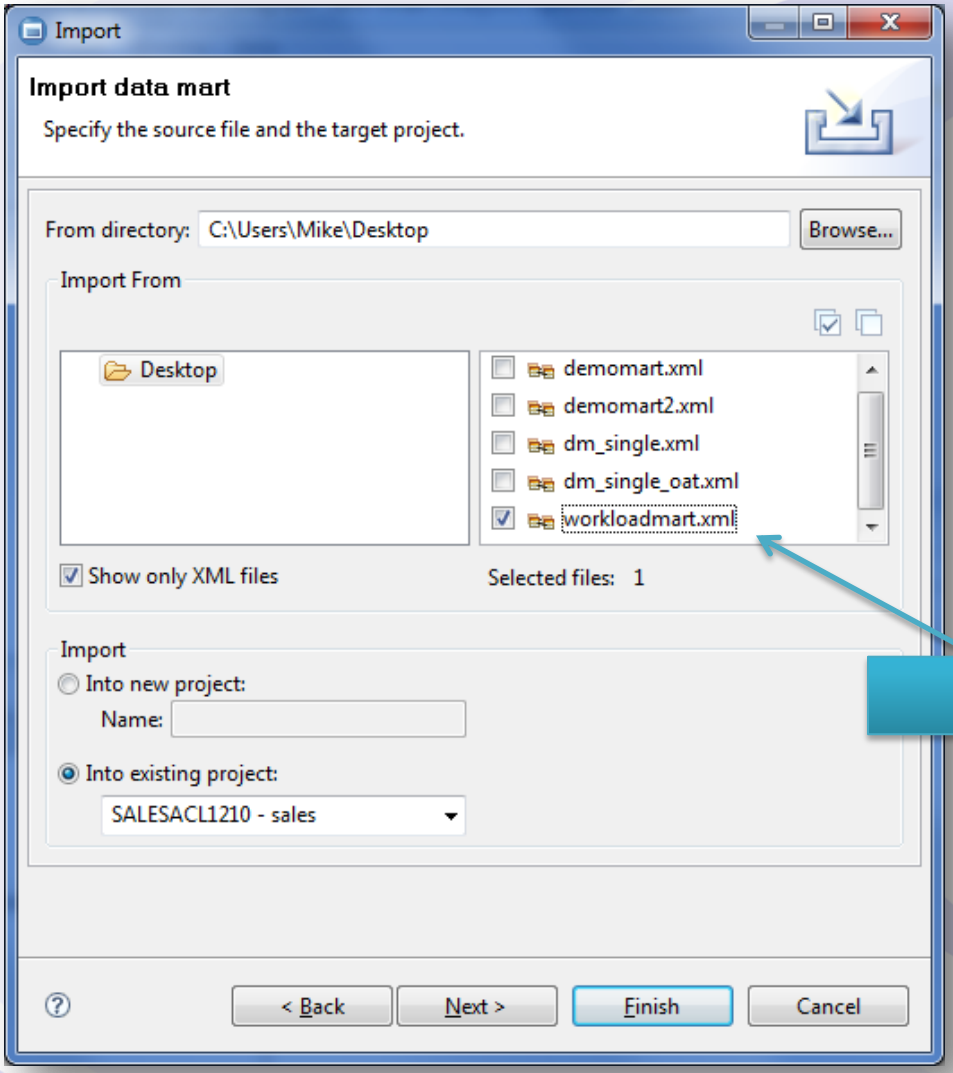
| Name | Status | Enabled | Last Load | Last Update |
|---|---|---|---|---|
| demomart | Active | Yes | N/A | None |
| sales | Active | Yes | N/A | None |
| workloadmart | *Load pending* | No | N/A | None |

# Data Marts in Studio




Choose Data Mart Import

# Data Marts in Studio




Choose the XML file

# Data Marts in Studio

Database Administration - SALESACL1210 - sales/workloadmart/default.mart_diagram - IBM Smart Analytics Optimizer Studio

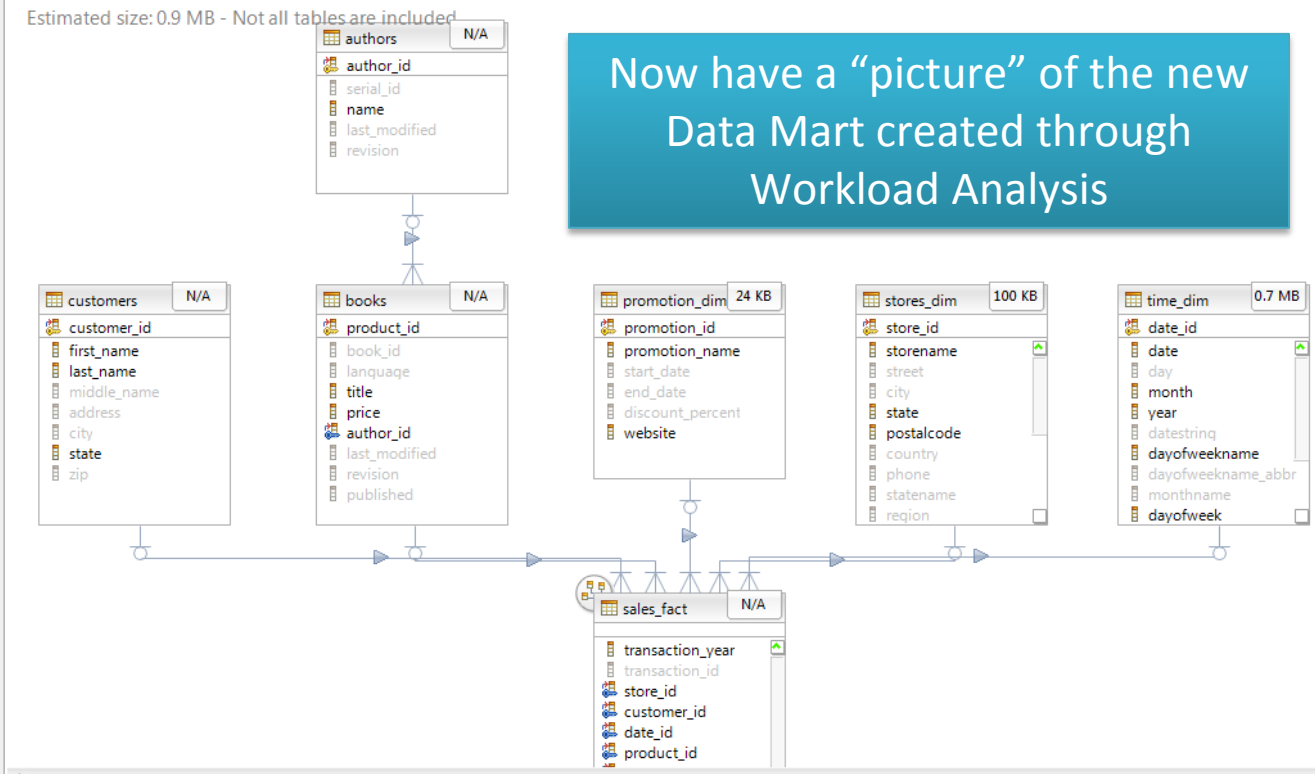
File Edit Diagram Navigate Search Project Data Mart Window Help

Data Project Explorer
- SALESACL1210 - sales
  - Data Marts
    - demomart (sales)
    - sales (sales)
    - workloadmart (sales)

Data Source Explorer
- Database Connections
  - sales (Informix 11.1)
    - sales
      - Accelerators
        - SALESACL1210 (Fully Operational)
          - demomart [Enabled]
          - sales [Enabled]
          - workloadmart [Disabled]
      - Roles
      - Schemas
      - Users

sales | demomart | *workloadmart

Estimated size: 0.9 MB - Not all tables are included




Now have a "picture" of the new Data Mart created through Workload Analysis

Properties | SQL Results | Workload List

<Data Mart> workloadmart

# Data Mart Refresh

- When a Data Mart is loaded, the data is compressed in memory
- Changes to the data in the database tables are not reflected in queries satisfied by IWA
- Used to require a full reload of the Data Mart, but now can refresh just part of the Data Mart

# Refresh a Partition (or Table)

```
execute function ifx_dropPartMart(
'SALESACL1210',
'sales‘,
'informix','sales_fact','salesfact_2011');
```

Drop the specified partition from the Data Mart

```
execute function ifx_loadPartMart(
'SALESACL1210',
'sales‘,
'informix','sales_fact','salesfact_2011');
```

Load the specified partition into the Data Mart

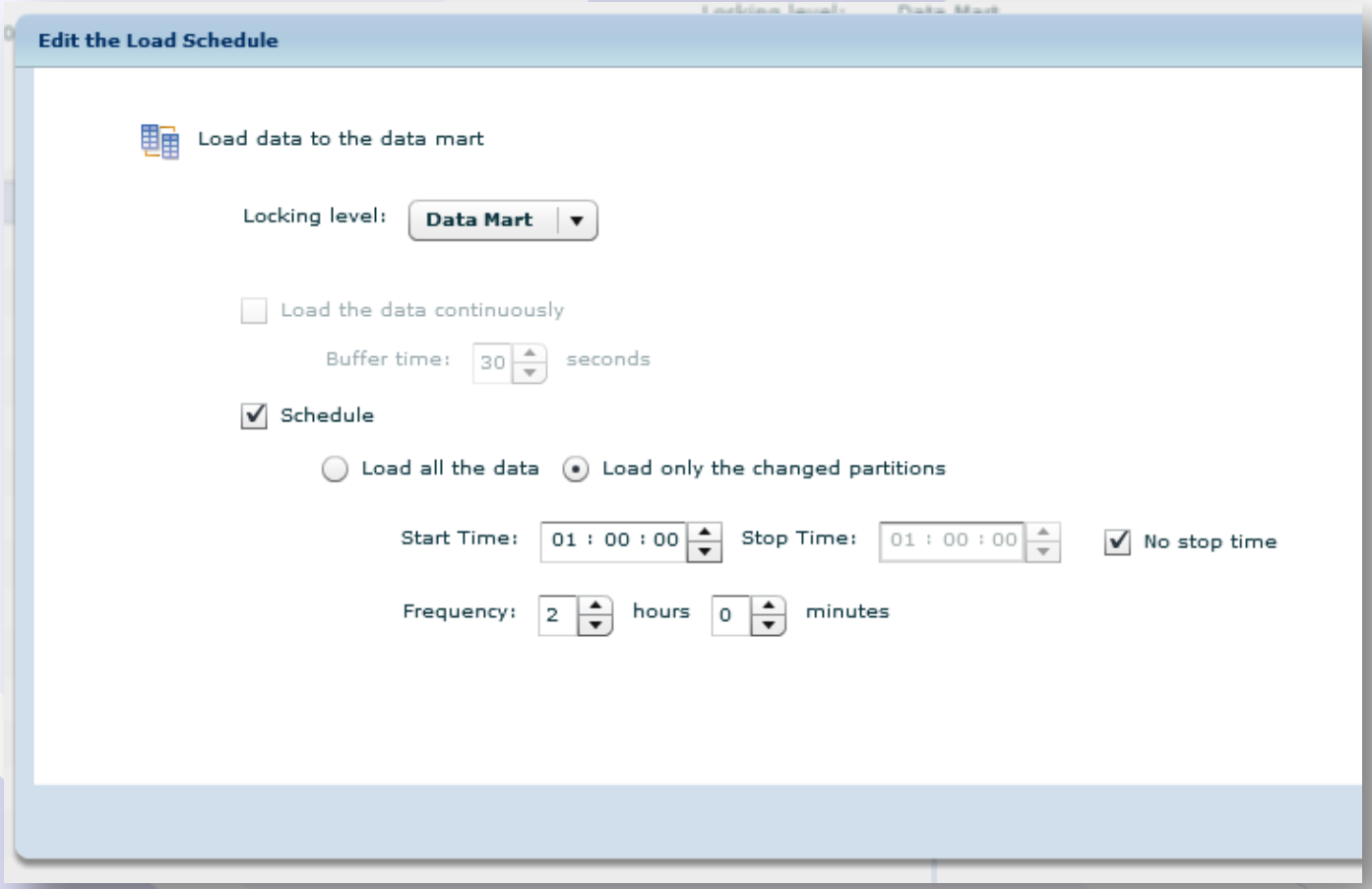
# Automatically Refresh changed Partitions (or Tables)

- Look for changes in the data and refresh the partitions or tables that have changed:

```
execute function ifx_refreshMart(
'AcceleratorName',
'DataMartName',
'LockMode');
```

# Automatically Refresh changed Partitions (or Tables)

## Set up in OAT



# Trickle Feed

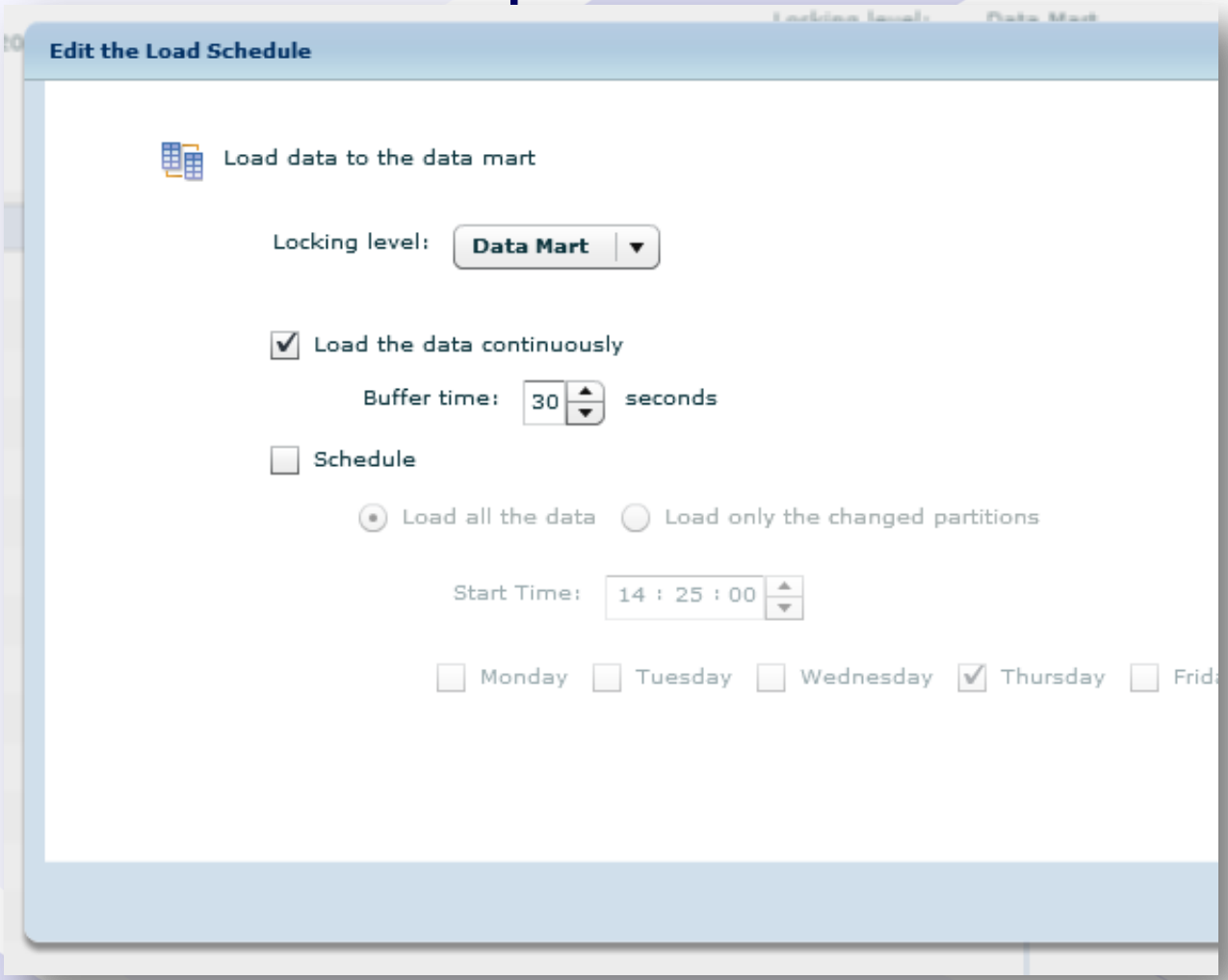
- The following will refresh the sales Data Mart every 60 seconds:

```
EXECUTE FUNCTION
ifx_setupTrickleFeed('SALESACL1210', 'sales', 60);
```

- Only NEW records inserted into the fact table will be included – not updates or deletes. All data changes made to dimension tables will be applied to the Data Mart
- Specify a negative interval to refresh the fact table only

# Trickle Feed

## Set up in OAT



# IWA requires you to do…

No Query Tuning and Optimizer Hints

No Database tuning

No Index creation, reorganization

No Update Statistics

No Partioning/Fragmentation

No Storage Management/Page size configuration

No Database/schema Changes

No Application Changes

No Summary Tables/Materialized Views

No Buying more expensive Hardware

No Change of expectations

**Power of Simplicity!**

Advanced DataTools

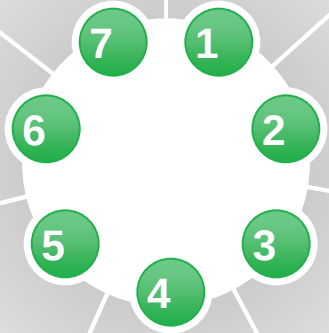
# Breakthrough technologies for performance

**Extreme Compression**
Required because RAM is the limiting factor.

**Row & Columnar Database**
Row format within IDS for transactional workloads and columnar data access via accelerator for OLAP queries.

**Multi-core and Vector Optimized Algorithms**
Avoiding locking or synchronization

**In Memory Database**
3rd generation database technology avoids I/O. Compression allows huge databases to be completely memory resident




**Predicate evaluation on compressed data**
Often scans w/o decompression during evaluation

**Frequency Partitioning**
Enabler for the effective parallel access of the compressed data for scanning. Horizontal and Vertical Partition Elimination.

**Massive Parallelism**
All cores are used within cpus for queries

# Questions

# Next Webcast

- Date: October 1, 2013
- Time: 2:00pm EST

- ***New Informix Warehouse Accelerator 12.10 Benchmarks***

# Fastest Informix DBA Contest 2013

- This year the Fastest Informix DBA Contest will be held over the web so anyone, anywhere in the world can participate from July 9 to September 30, 2013.
- The challenge will be a combination of OLTP and batch.
- The winner will be the DBA who can get the most transactions per minute in an OLTP benchmark and at the same time run a batch billing job and generate the most bills in a 10 minute run.

# Fastest Informix DBA Contest 2013 Schedule

- July 9 - Register for contest and download code
- July 9 - Fastest Informix DBA Contest Webcast - 2:00pm EST - View Replay!
- July 22 - Start submitting your code and onconfig entries, you will be emailed back the results
- August 20 - Webcast - Latest status update on the contest
- September 30 - All final entries must be submitted and contest closes
- October 15 - Webcast to Announce Winners

# Informix Training in 2013

- September 9-12, 2013
  - **Informix for Database Administrators**
- October 28-31, 2013 –
  - **Advanced Informix Performance Tuning**
- All courses can be taken online on the web from your desk or at our training center in Virginia.
- We guarantee to *NEVER* cancel a course and will teach a course as long as one student is registered!

# Thank You


## Mike Walker
## Advanced DataTools Corporation

mike@advancedatatools.com


For more information:

http://www.advancedatatools.com