

# Data Visualization on Hana

## OBIEE 12c and SAP Hana Integration

Milano, 8 Giugno 2017



# Agenda

- ✓ Technology Reply
- ✓ Obiettivi e contesto
- ✓ Connection pool DB/EE
- ✓ Sorgente DDI
- ✓ Applicazione custom JZEE

- ✓ Strategie a confronto
- ✓ Q&A

American

2



# Technology Reply

## Mission

- Oracle partner from 1996
- Technology Reply is focused on design and implement solutions based on **Oracle** and **Java** technologies.

## Certifications & Specs

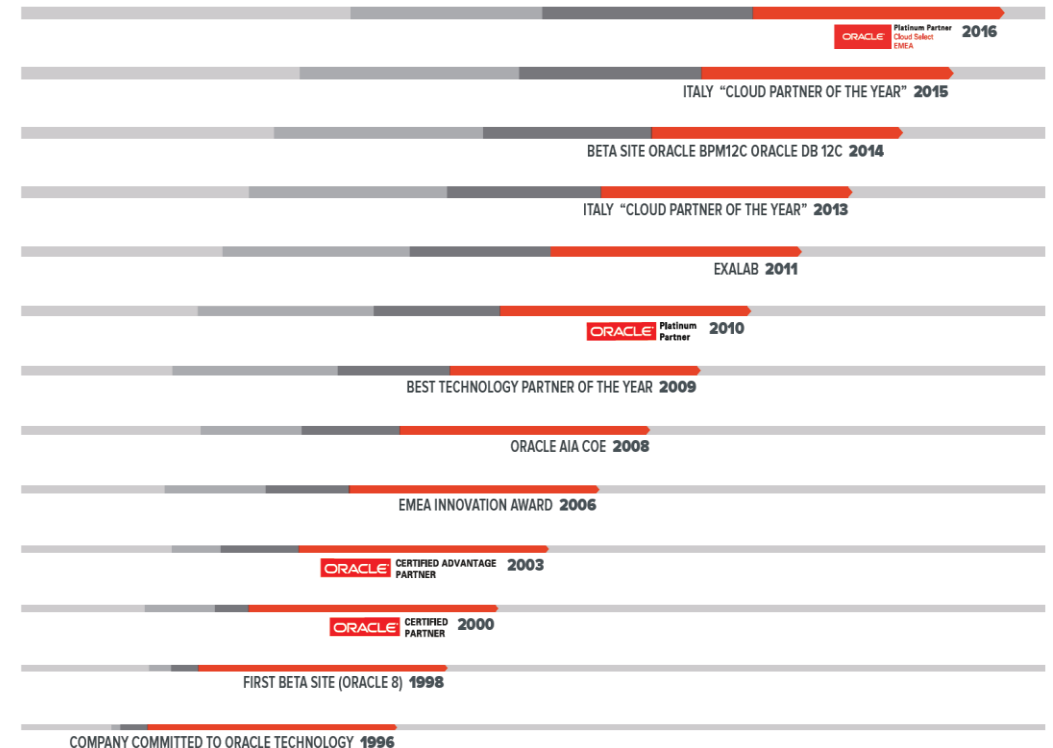
- 27 specializations
- > **170** certified resources
- 2 Advanced Specializations (Java SE7, DB11g)

## Excellence Award Winner

- 2015: Italy “Cloud Partner of the Year”
- 2013: Italy “Cloud Partner of the Year”
- 2009: “Best Technology Partner of the year”

## R&D

- IaaS / PaaS Lab
- Exa Lab
- AIA co-development center of excellence
- Beta Programs Participation



# WE KNOW ORACLE!

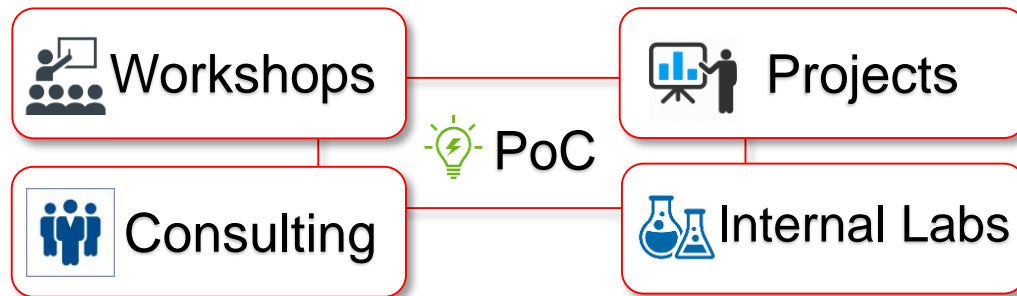
Everyday we design and deliver **Data & Analytics solutions** based on **Oracle** products, in any industry, across our Country.

ORACLE®

Platinum  
Partner

Cloud Standard

## Market Approach



## Expertise growth

Post-graduation  
courses

Reply camps

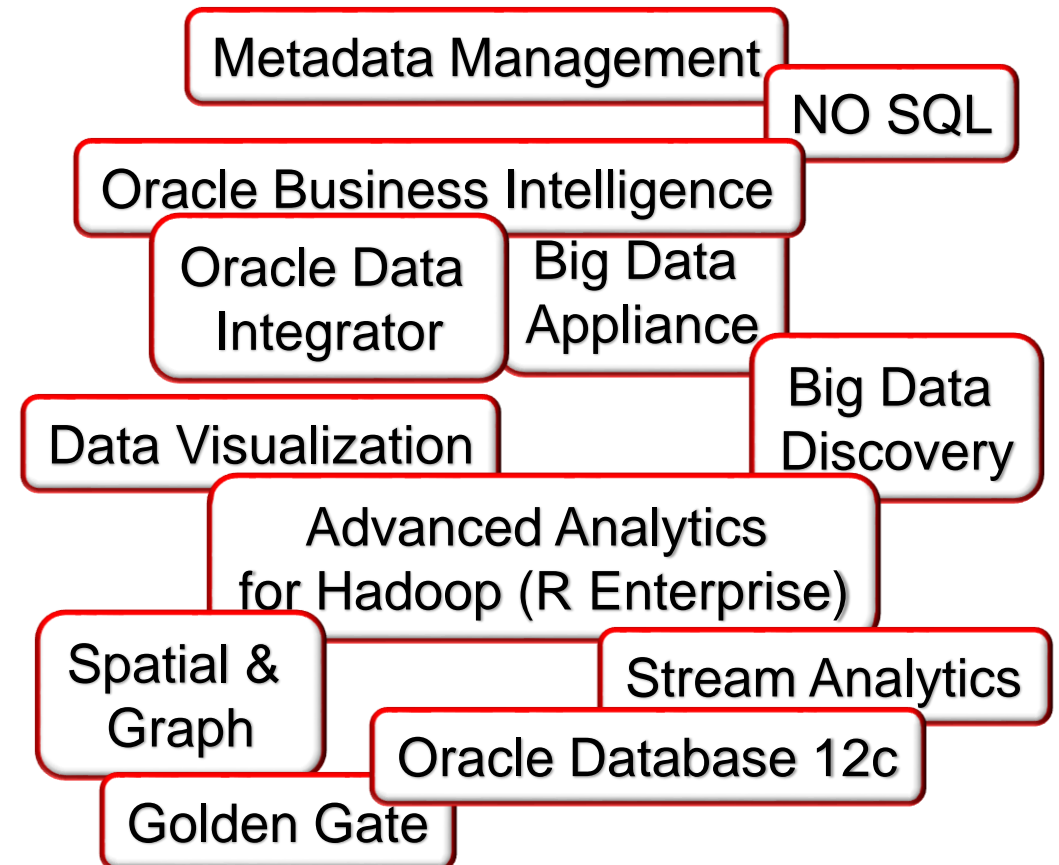
Certifications

CoPs

Hackathons



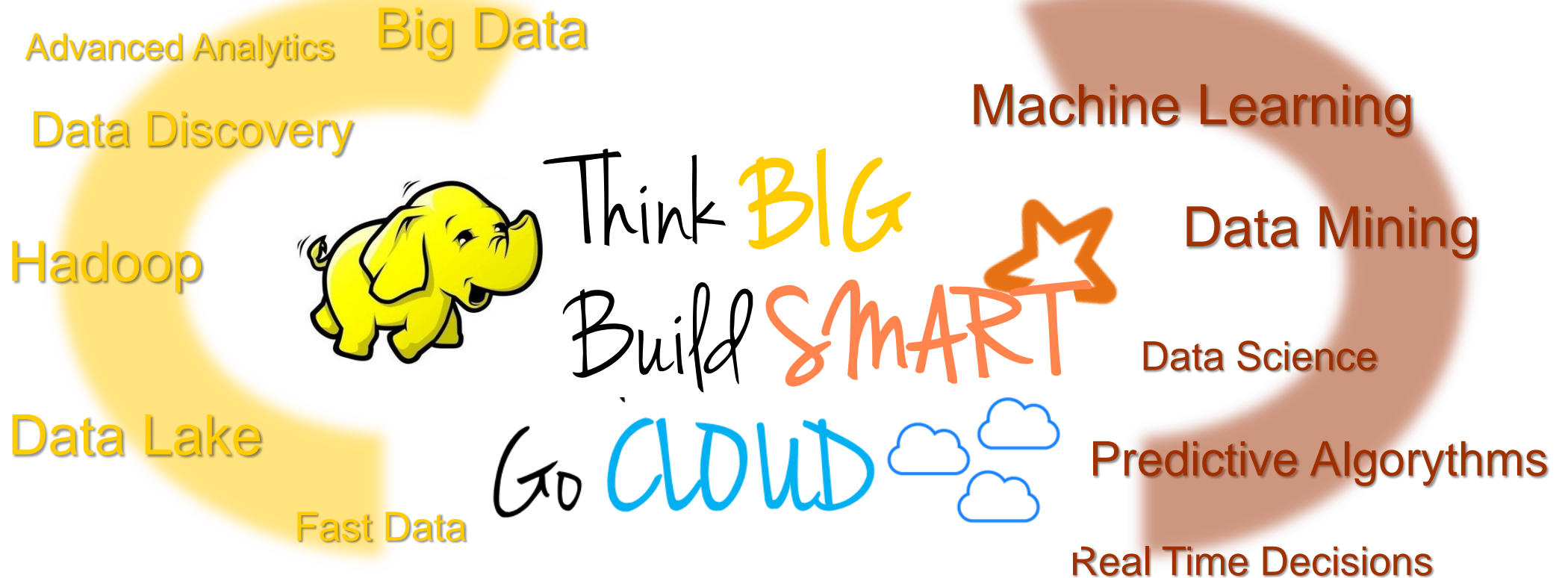
## Products



# Technology Reply

## Big Data & Machine Learning Unit

Focused on design of Big Data / Fast Data architectures, advanced Analytics and Data Mining solutions based on **Oracle** offering (both on-premise and in cloud)





Integrazione HANA - OBIEE



# Scenario & Obiettivi



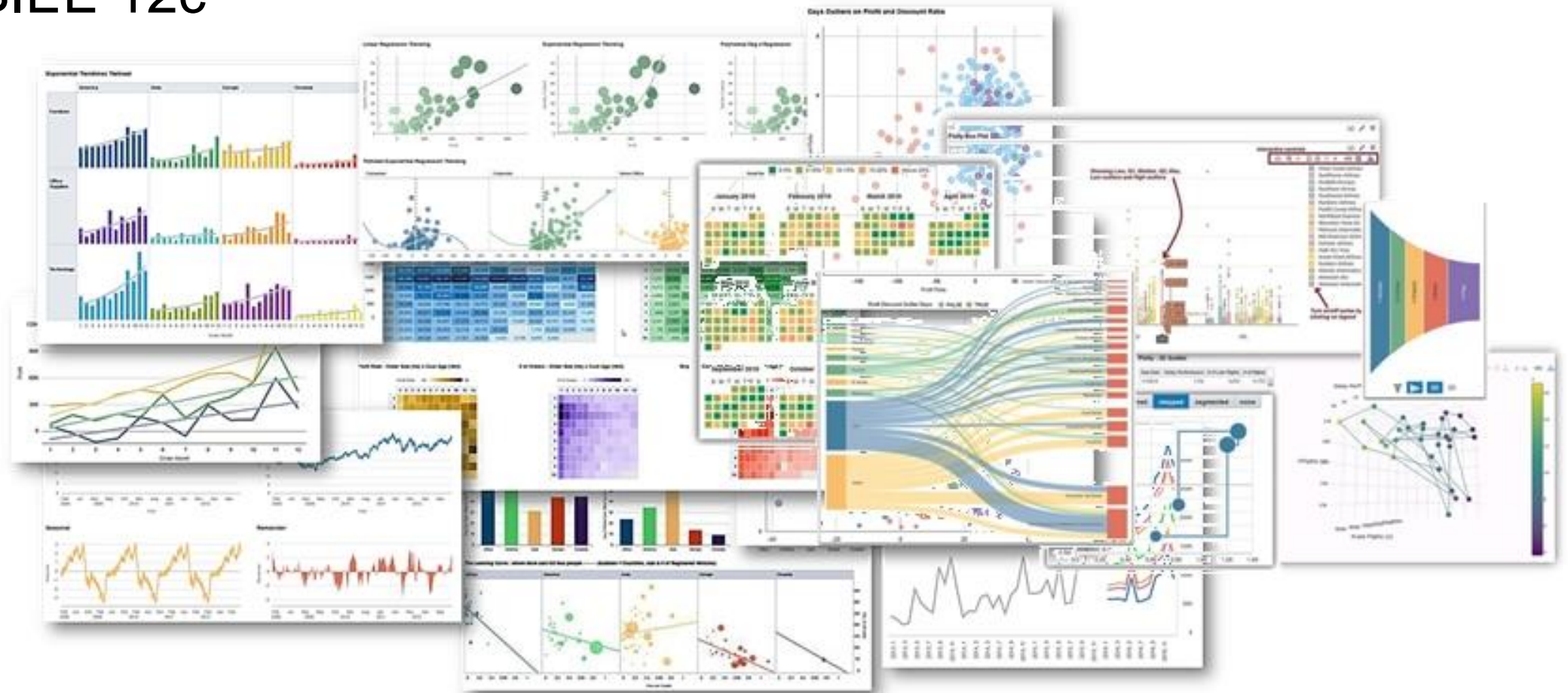
In un contesto di Business Intelligence dove la piattaforma di front-end di riferimento è **Oracle Business Intelligence 12c**, è emersa la necessità di rendere disponibili al business i dati presenti su **SAP HANA**.

Al fine di sfruttare il **know-how** di utilizzo di **OBIEE** molto diffuso tra gli utenti e le potenzialità dello strumento di Data Investigation «**Oracle Data Visualization**» si è deciso di integrare i dati presenti su HANA al fine di renderli fruibili tramite tali strumenti.

Sono state implementate tre **differenti strategie** di integrazione; in base alle esigenze progettuali viene di volta in volta selezionata la migliore strategia.



# OBIEE 12c



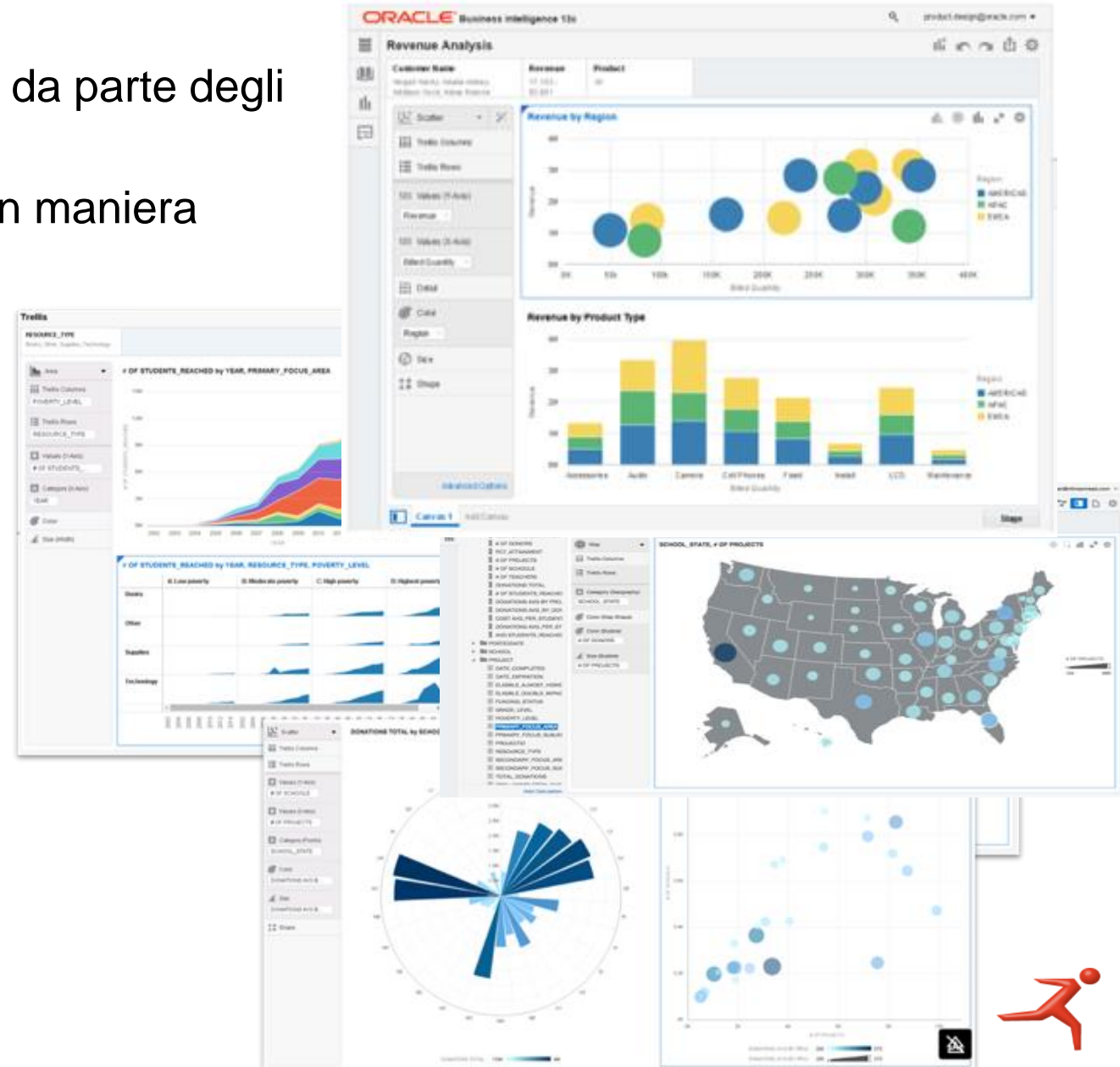
**Look and feel** aggiornato, maggiore **semplicità** di utilizzo.  
Installazione, aggiornamento, **amministrazione** semplificate.  
**Unica piattaforma** che risponde a **molteplici esigenze**, dal Dashboarding all'Advanced Analytics.  
Funzionalità di **data mashup** semplici ed immediate.  
Accesso **mobile** supportato senza necessità di customizzazioni.



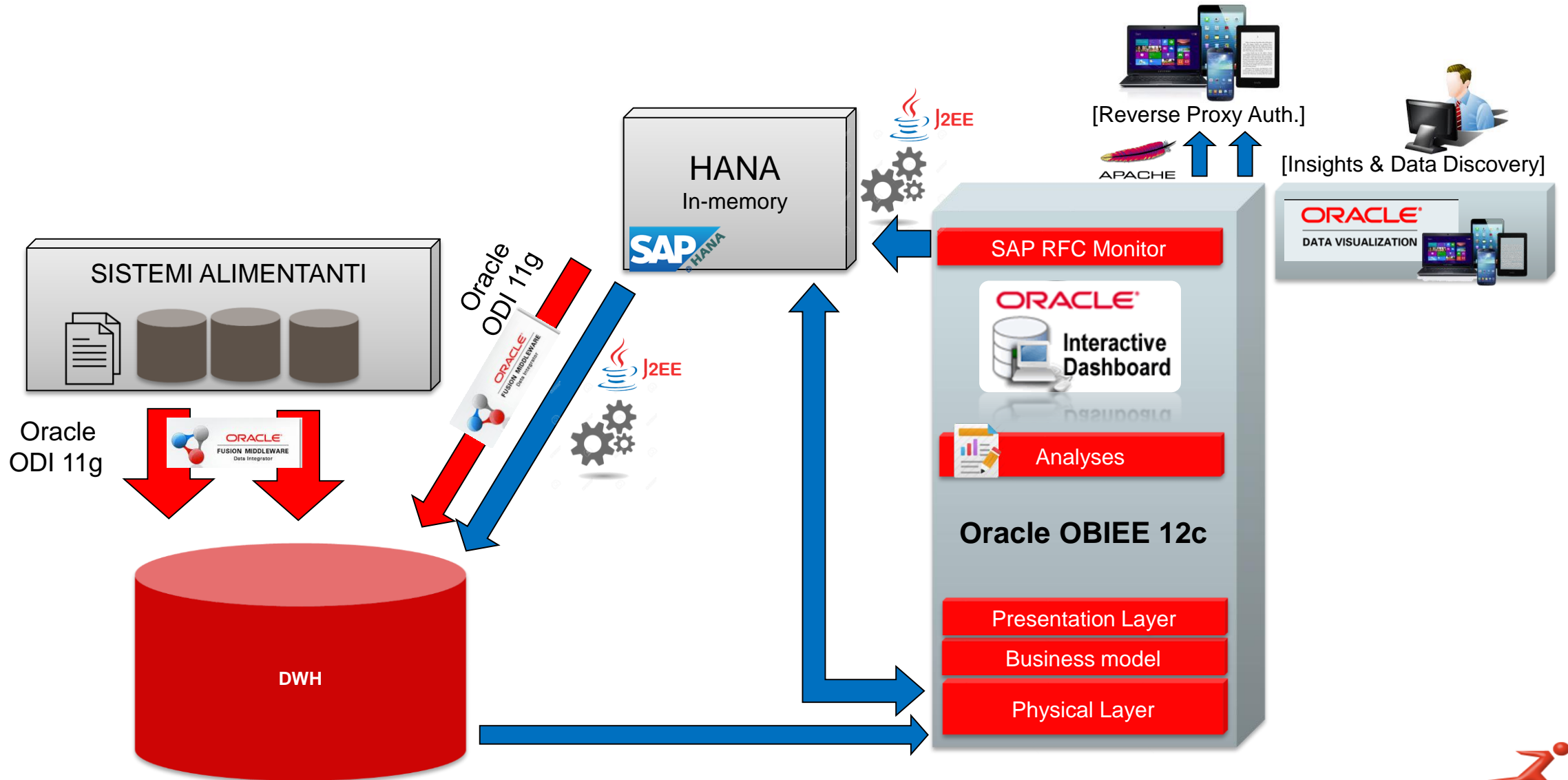


# Oracle Data Visualization

- Tool “Tableau-style” per l’analisi dei dati da parte degli utenti finali.
- Esplorazione e visualizzazione dei dati in maniera semplice e interattiva (point-and-click)
- Usa il BI Repository come sorgente dati (single-version-of-the-truth)
- Consente analisi veloci
  - ▶ Usando le “Subject Area” OBIEE
  - ▶ Usando dati esterni (es. Excel) senza necessità di modifiche al repository.



# Diagramma di contesto

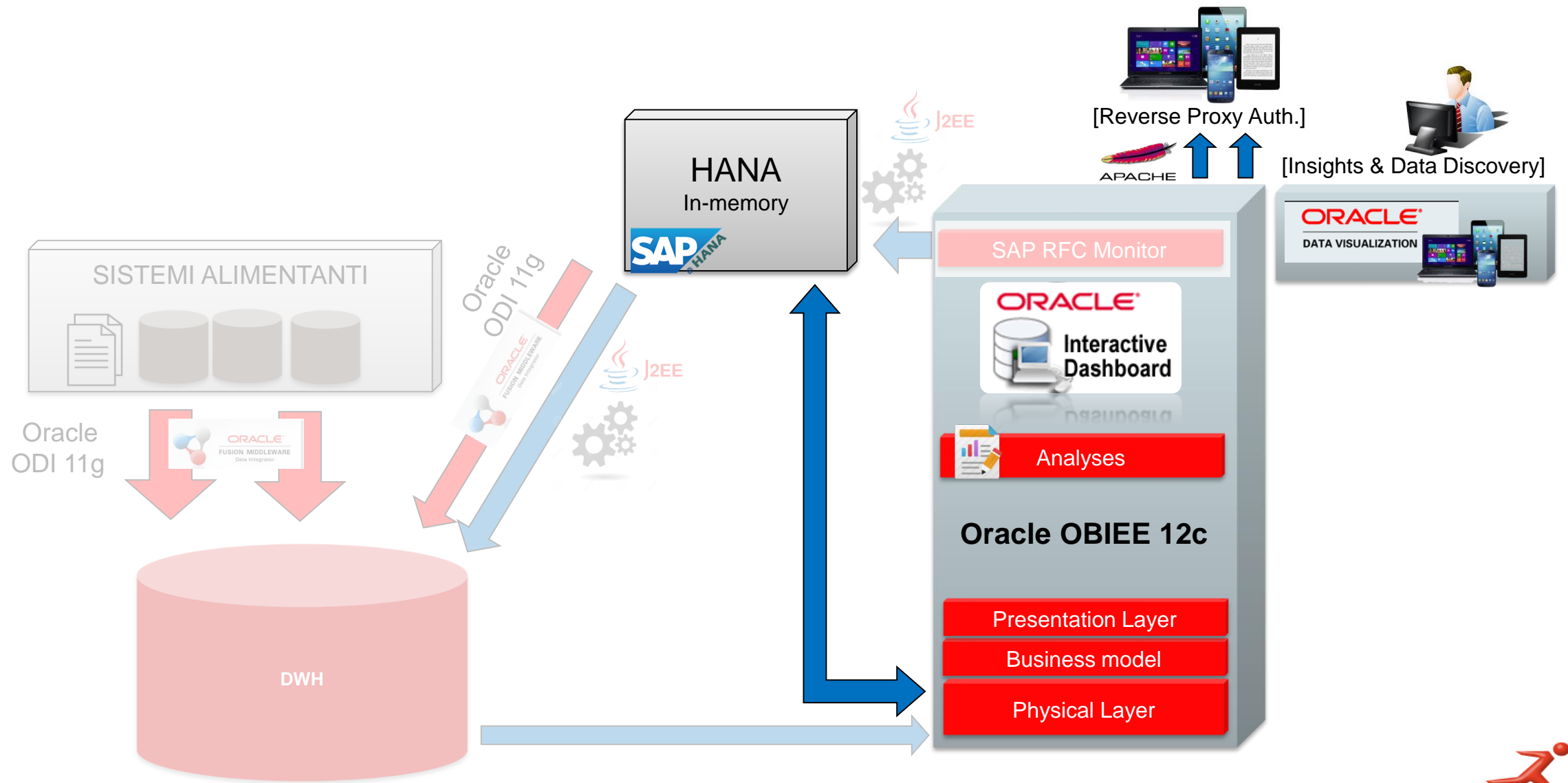


# Modalità di integrazione HANA

- I. Connection pool OBIEE
- II. Sorgente ODI
- III. Applicazione custom J2EE



# I. Connection pool OBIEE



# I. Connection pool OBIEE

Verrà qui di seguito descritto come installare e configurare la connessione su:

- OBIEE 12c Server (Server Linux, nel nostro caso)
- OBIEE Administrator Tool (installato su Macchina Windows, nel nostro caso)



# I. Connection pool OBIEE

Al fine di ottenere la connettività tra HANA e OBIEE è necessario installare i driver odbc specifici.

Tali driver sono disponibili sul SAP ONE SUPPORT LAUNCHPAD:

The screenshot shows the SAP ONE SUPPORT LAUNCHPAD interface for downloading SAP HANA Client 2.0. The page title is "SAP HANA CLIENT 2.0 (SUPPORT PACKAGES AND PATCHES)". There are tabs for "DOWNLOADS", "INFO", and "ECCN INFO". A message box explains multispanning for large packages. Below, a table lists available items for download, filtered by "LINUX ON X86\_64 64BIT".

**SAP HANA CLIENT 2.0 (SUPPORT PACKAGES AND PATCHES)**

DOWNLOADS INFO ECCN INFO

**Items Available to Download (1)** LINUX ON X86\_64 64BIT

*Selected Items (0)*

<input type="checkbox"/>	Name	Patch Level	File Type	File Size	Release Date	Change Date	Related Info
<input type="checkbox"/>	<a href="#">IMDB_CLIENT20_00_74-80002082.SAR</a> Revision 00 for SAP HANA CLIENT 2.0	74	SAR	262468 KB	14.02.2017	14.02.2017	☰

(\*) for validation only



# I. Connection pool OBIEE

I driver, per essere estratti, necessitano di un apposito tool: SAPCAR.exe

Once the file is downloaded, we need to extract it. This file has extension **.SAR** (SAP Archive), to extract it we need SAPCAR utility.

If **SAPCAR** is not downloaded, just search for SAPCAR on SAP service marketplace portal (<http://support.sap.com/swdc>) and execute following command to extract any .SAR file:

```
SAPCAR.exe -xvf AnyArchiveFile.SAR
```

OR

```
SAPCAR -xvf AnyArchiveFile.SAR
```

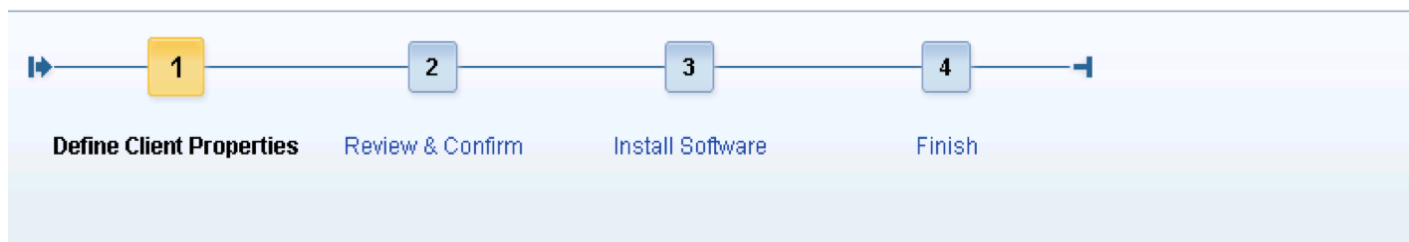


# I. Connection pool OBIEE

Dopo aver estratto i driver sarà necessario eseguire il tool hdbsetup.exe per installare i driver sul client.

## SAP HANA

### Lifecycle Management



Choose an installation to update, or choose a path for a new installation

Update SAP HANA Database Client

Install new SAP HANA Database Client

#### Installation Details

##### To be installed

Version: 2.0.74.1485465563

Mode: 64bit

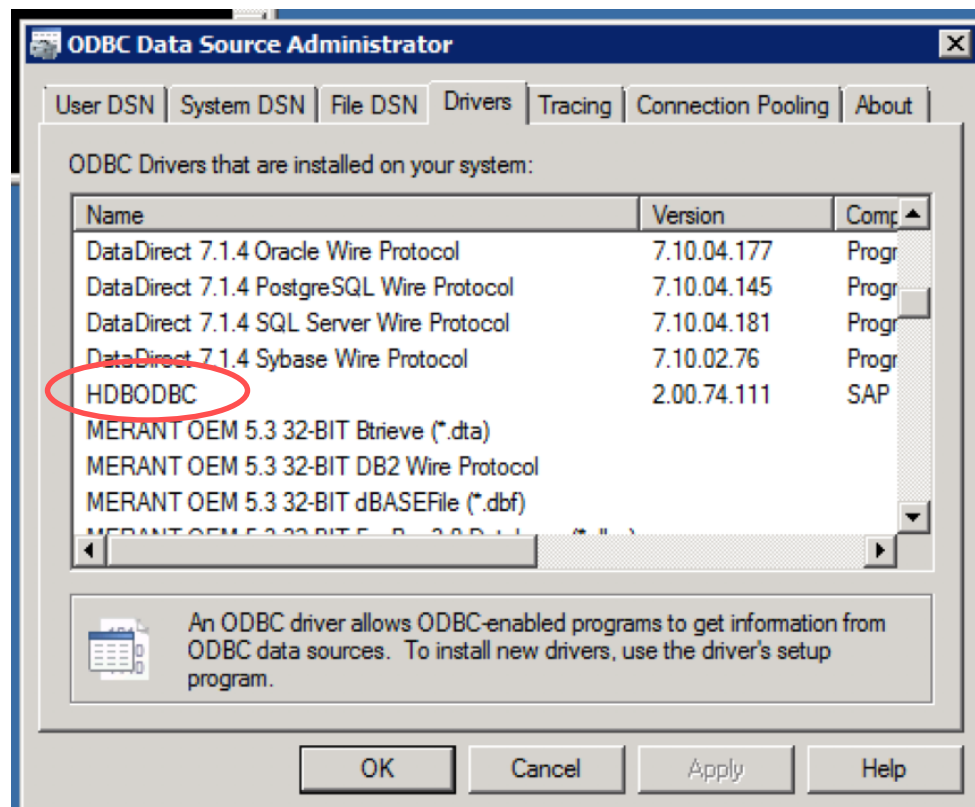
Installation path: C:\Program Files\sap\hdbclient





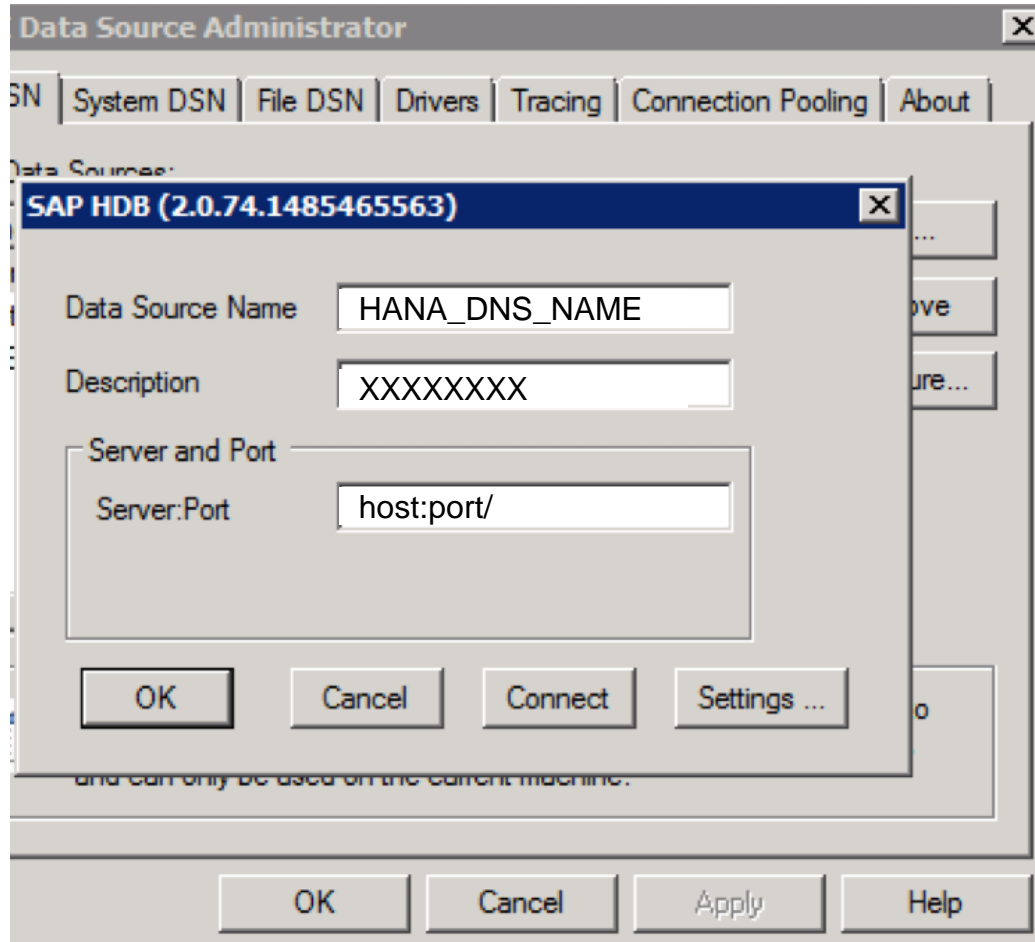
# I. Connection pool OBIEE

A questo punto, i driver installati compariranno tra le Data Sources ODBC:



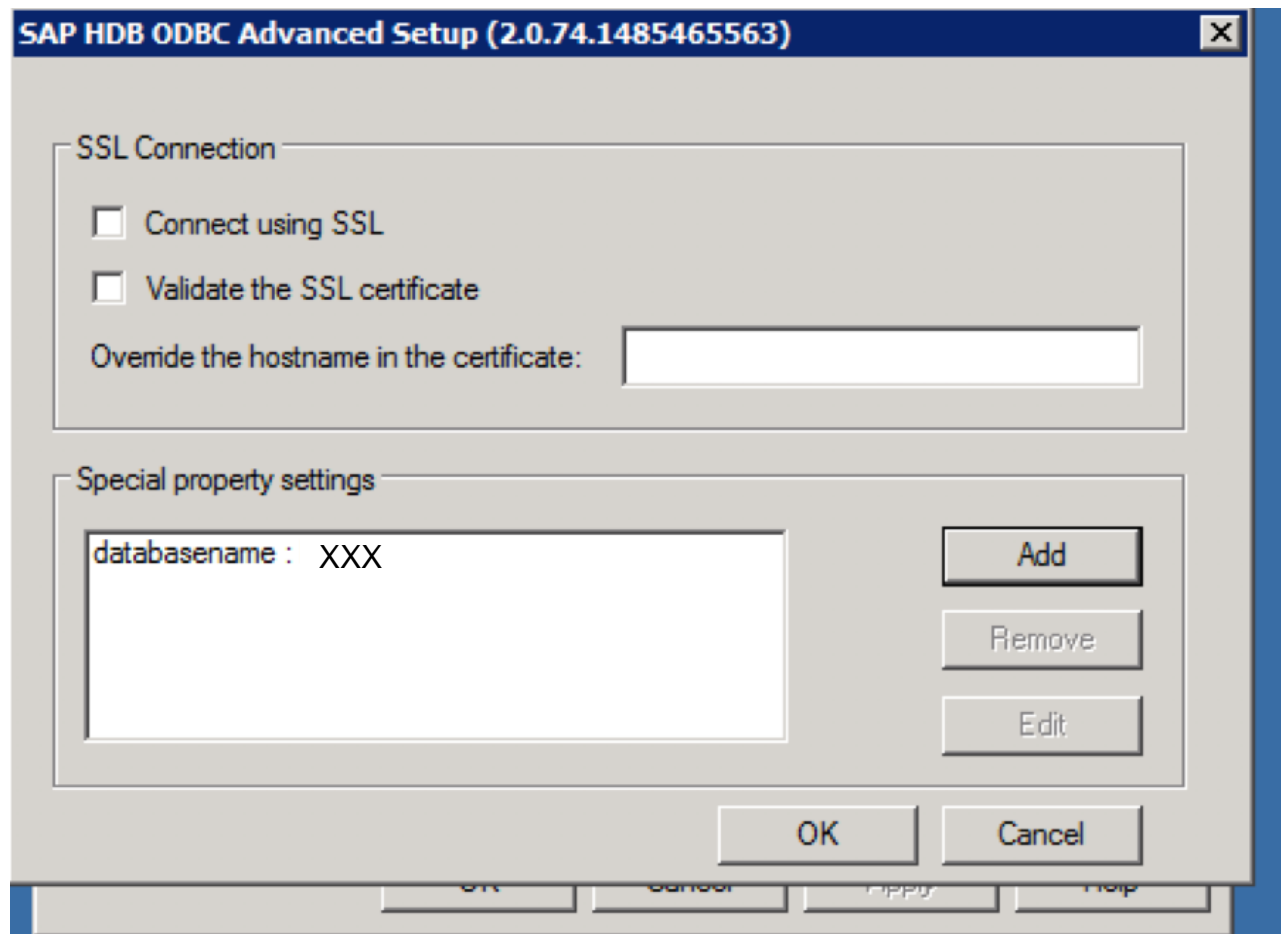
# I. Connection pool OBIEE

Occorre quindi creare il nuovo DSN selezionando il driver installato:



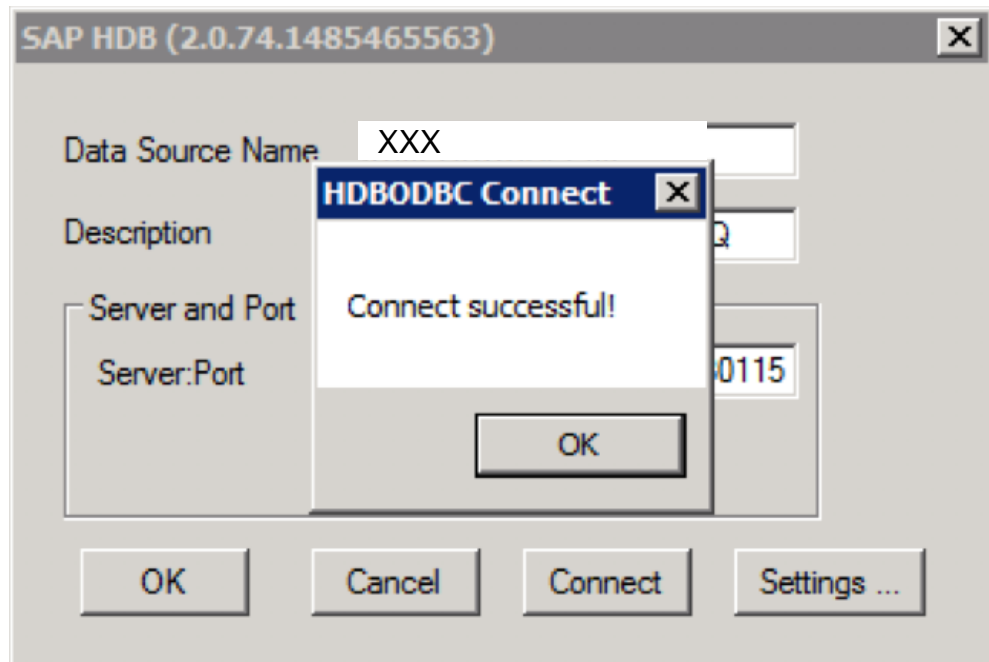
# I. Connection pool OBIEE

Ed aggiungere tra le properties il db name corretto.



# I. Connection pool OBIEE

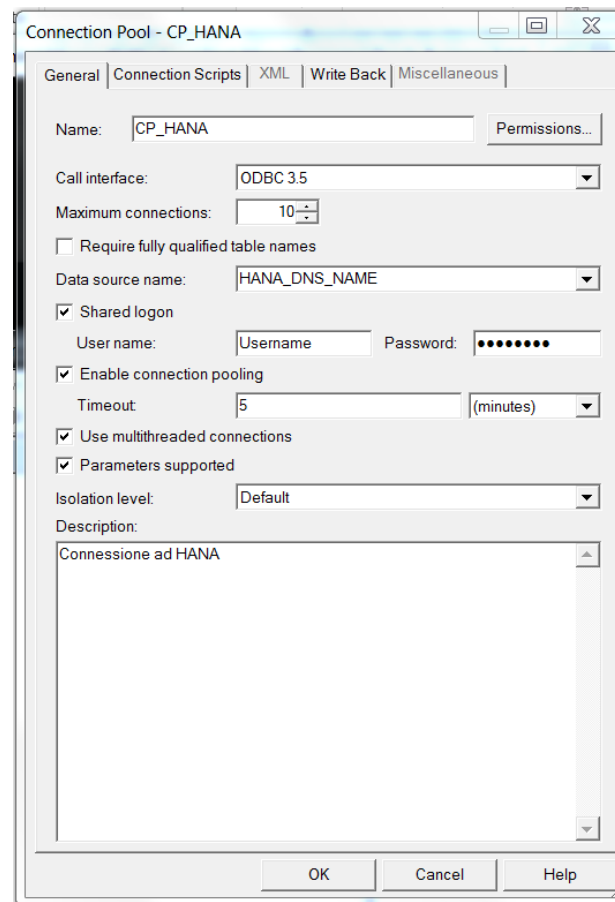
Il test di connessione consente di verificare il buon esito della configurazione.



# I. Connection pool OBIEE

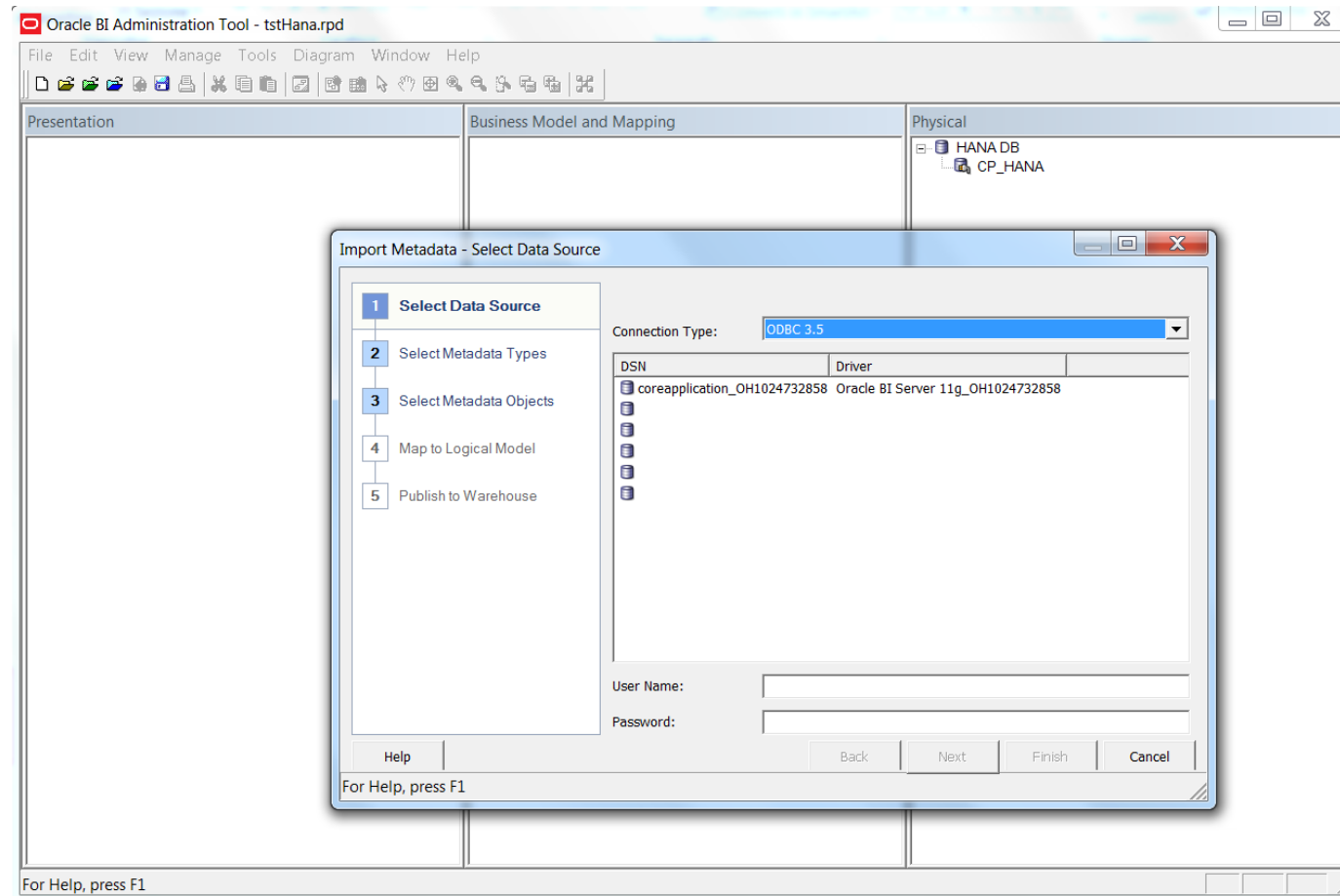
Occorre a questo punto configurare il repository attraverso l'Administration Tool.

All'interno del physical model viene definito il nuovo DB e viene configurato il connection pool opportunamente.



# I. Connection pool OBIEE

Sarà possibile a questo punto importare i metadati direttamente da HANA, tramite la funzionalità apposita dell'Administration Tool.



# I. Connection pool OBIEE

Infine, è necessario installare i driver anche sul Server OBIEE.

```
SAPCAR -xvf IMDB_CLIENT20_00_74-80002082.SAR
```

```
cd /home/oracle/SAP_CLIENT/SAP_HANA_CLIENT
```

```
/home/oracle/SAP_CLIENT/SAP_HANA_CLIENT/hdbsetup  
(in modalità X11)
```



# I. Connection pool OBIEE

La configurazione del Server OBIEE è l'ultimo step che ci consente l'utilizzo di HANA come Data Source.

Tale configurazione consiste in due operazioni su appositi file di configurazione.

1 - aggiungere all'interno del file **obis.properties**  
(\$ORACLE\_HOME/user\_projects/domains/bi/config/fmwconfig/bienv/OBIS) :

`LD_LIBRARY_PATH=/home/oracle/sap/hdbclient` (in base a dove sono stati installati i driver)





# I. Connection pool OBIEE

La configurazione del Server OBIEE è l'ultimo step che ci consente l'utilizzo di HANA come Data Source.

Tale configurazione consiste in due operazioni su appositi file di configurazione.

2 - aggiungere all'interno del file **odbc.ini**:  
(\$ORACLE\_HOME/user\_projects/domains/bi/config/fmwconfig/bienv/core)

```
[QMPHDQSAPPMP]
servernode=lnx478-116it:30115
Driver=/home/oracle/sap/hdbclient/libodbcHDB.so
Description=Accesso a SAP HANA
databasename=XXX_NAME
CHAR_AS_UTF8=1
DriverUnicodeType=1
```



# I. Connection pool OBIEE

Riavviare infine BI Server (OBIS1) da Enterprise Manager.

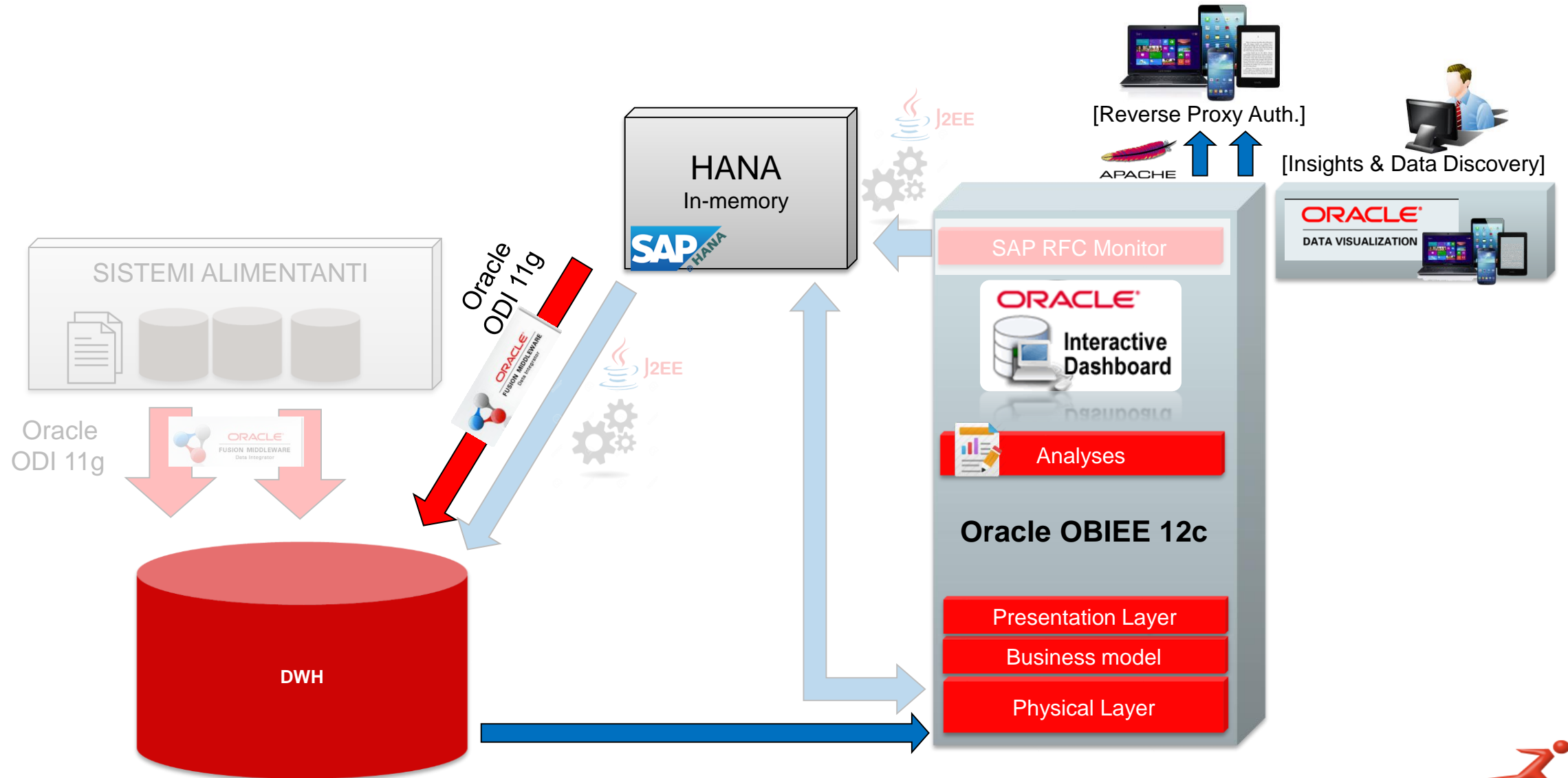
A questo punto è possibile disegnare su repository l'intero modello fisico – logico e di presentazione ed avere così una (o più) subject area dedicate alla fruizione dei dati direttamente su HANA.

Sfruttando queste Subject Area si potranno dunque costruire analisi e dashboard OBIEE come di consueto.

Sarà anche possibile utilizzare queste nuova Subject Area per alimentare un progetto di Data Visualization.



# II. HANA come sorgente ODI



## II. HANA come sorgente ODI

Al fine di ottenere la connettività tra HANA e ODI risulta necessario installare in questo caso gli appositi driver jdbc.

Verrà qui di seguito descritto come installare e configurare la connessione su:

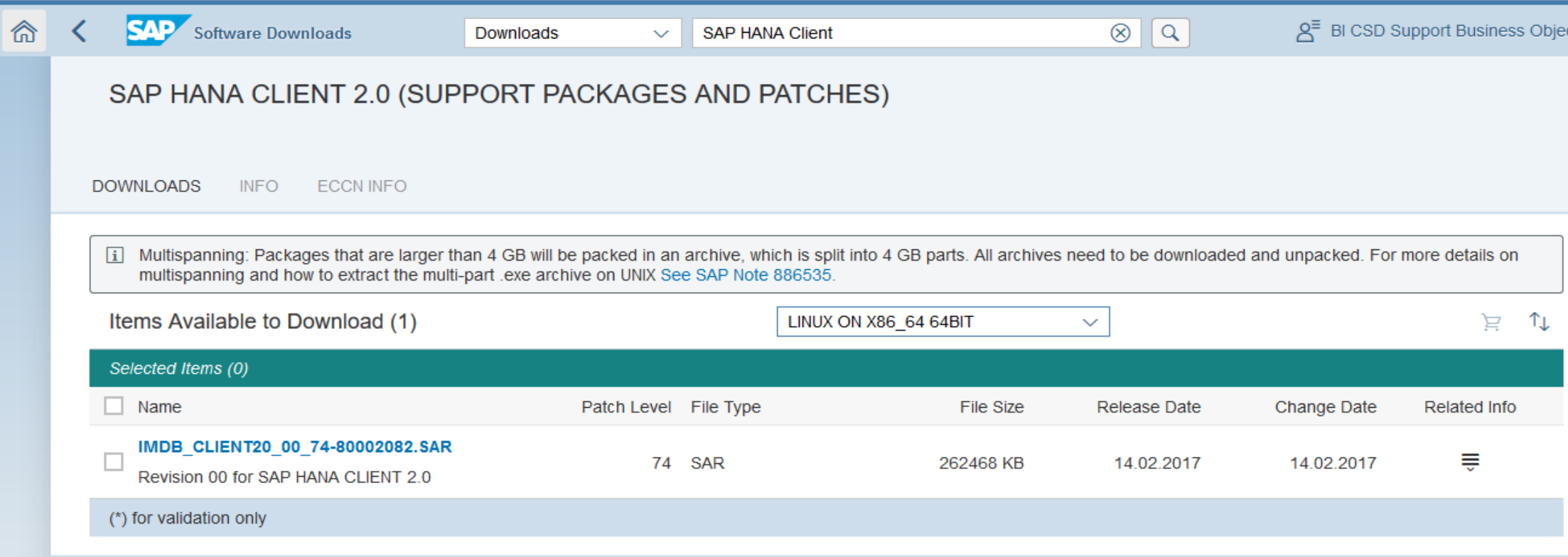
- ODI Studio (Macchina Windows, nel nostro caso)
- ODI Server (Server Linux, nel nostro caso)

L'utilizzo di SAP Java Connectors, è soluzione standard; non sono necessari Knowledge module custom.



## II. HANA come sorgente ODI

Anche in questo caso i driver necessari sono disponibili sul SAP ONE SUPPORT LAUNCHPAD:



SAP HANA CLIENT 2.0 (SUPPORT PACKAGES AND PATCHES)

DOWNLOADS INFO ECCN INFO

**i** Multispanning: Packages that are larger than 4 GB will be packed in an archive, which is split into 4 GB parts. All archives need to be downloaded and unpacked. For more details on multispanning and how to extract the multi-part .exe archive on UNIX See [SAP Note 886535](#).

Items Available to Download (1) LINUX ON X86\_64 64BIT

*Selected Items (0)*

<input type="checkbox"/>	Name	Patch Level	File Type	File Size	Release Date	Change Date	Related Info
<input type="checkbox"/>	<b>IMDB_CLIENT20_00_74-80002082.SAR</b> Revision 00 for SAP HANA CLIENT 2.0	74	SAR	262468 KB	14.02.2017	14.02.2017	☰

(\*) for validation only



## II. HANA come sorgente ODI

Anch'essi per essere estratti all'interno del client necessitano di un apposito tool: SAPCAR.exe

Once the file is downloaded, we need to extract it. This file has extension **.SAR**(SAP Archive), to extract it we need SAPCAR utility. If **SAPCAR** is not downloaded, just search for SAPCAR on SAP service marketplace portal(<http://support.sap.com/swdc>) and execute following command to extract any .SAR file:

```
SAPCAR.exe -xvf AnyArchiveFile.SAR
```

OR

```
SAPCAR -xvf AnyArchiveFile.SAR
```



## II. HANA come sorgente ODI

La configurazione del driver JDBC sul server ODI avviene copiando il file ngdbc.jar, disponibile nella cartella di estrazione dei driver, nel seguente percorso:

```
ODI_HOME/oracledi/agent/drivers/
```

L'url di connessione in ODI dovrà essere così formato:

```
URL= "jdbc:sap://hostname:port/SAPPMP?databaseName=NAME"
```



## II. HANA come sorgente ODI

A questo punto sarà possibile sfruttare HANA come sorgente dati per ODI e, attraverso i Knowledge Modules standard, si potranno realizzare (come nel nostro caso) i mapping di alimentazione di apposite tabelle su DWH.

Tali tabelle, modellate all'interno del repository OBIEE, serviranno come esposizione dei dati HANA all'interno della piattaforma Oracle Business Intelligence.

Come nel caso precedente, sarà quindi possibile disegnare su repository l'intero modello fisico – logico e di presentazione ed avere così una (o più) subject area dedicate alla fruizione dei dati integrati su DWH da HANA.

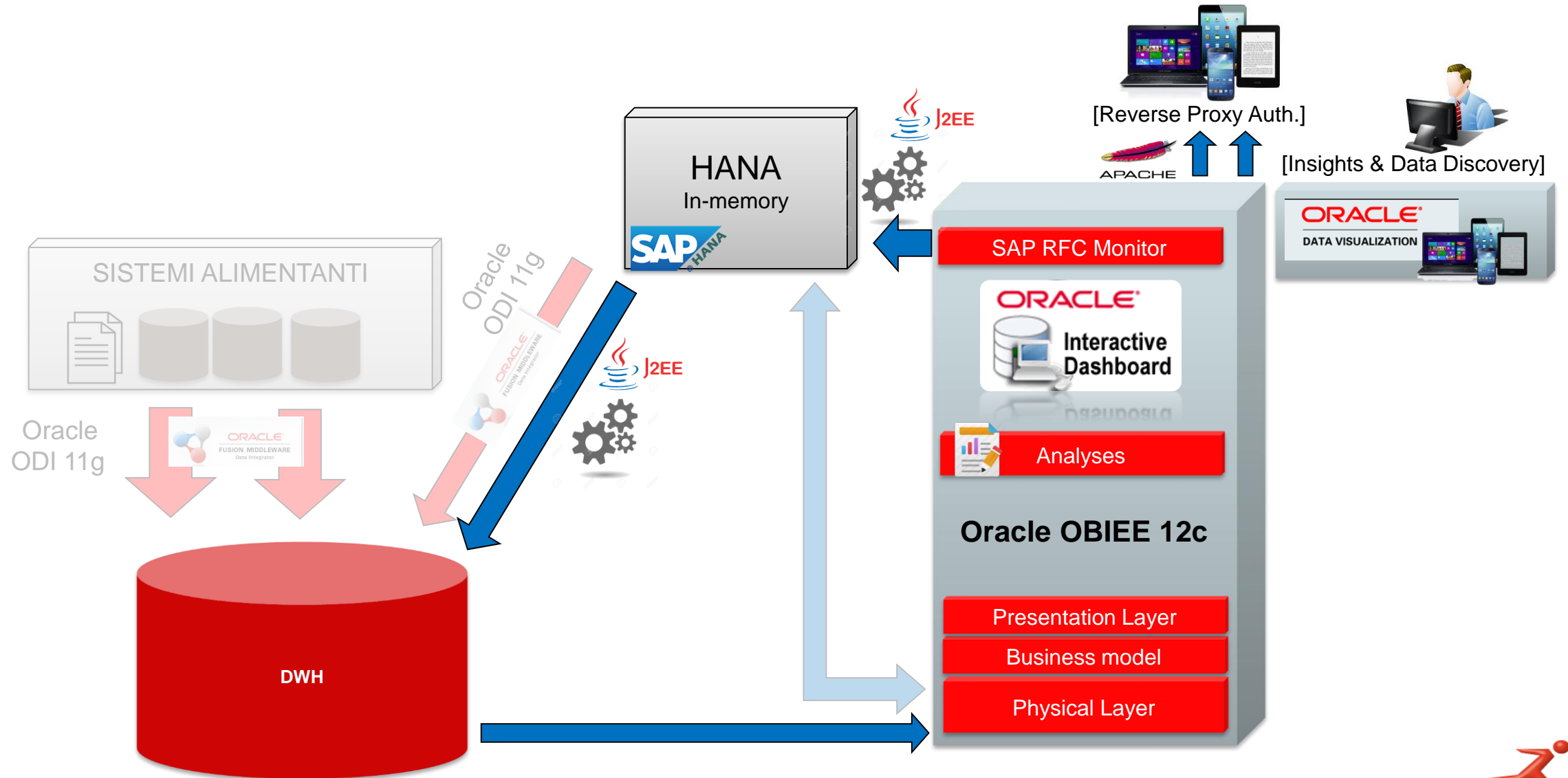
Sfruttando queste Subject Area si potranno dunque costruire analisi e dashboard OBIEE come di consueto.

Sarà anche possibile utilizzare queste nuova Subject Area per alimentare un progetto di Data Visualization.





# III. Applicazione custom J2EE



### III. Applicazione custom J2EE

Nell'ambito dell'integrazione sullo stack OBIEE dei dati provenienti da HANA, è emersa inoltre la necessità di eseguire **chiamate «on demand»** che eseguissero, sulla base di determinati parametri, specifici programmi su SAP.

L'output di tali elaborazioni voleva poi essere reso disponibile agli utenti su OBIEE.

Si è così realizzata una applicazione J2EE che si occupasse di eseguire i programmi (passando i dovuti parametri) e di memorizzare il risultato dell'esecuzione su apposite strutture su DWH.



# III. Applicazione custom J2EE

## Elementi in gioco

**JCO:** the **SAP Java Connector** is a middleware component that allows a JAVA application to call or communicate to any SAP systems and vice versa.

**RFC: Remote Function Call** is the standard SAP interface for communication between SAP systems. The RFC calls a function to be executed in a remote system.



### III. Applicazione custom J2EE

Un ulteriore passo di integrazione è stato rappresentato dall'inserimento, all'interno di una apposita Dashboard Page, di un cruscottino di controllo che consentisse agli utenti di:

- Eseguire le RFC (tra quelle preconfigurate) sulla base di determinati parametri.
- Monitorarne lo status (in corso, terminato con successo, in errore) e il relativo storico
- Visualizzare lo storico delle esecuzioni
- Decidere se rendere il risultato disponibile a tutti gli utenti o soltanto alla propria user.



# III. Applicazione custom J2EE

## Rfc Execution

\* RFC Description  \* View

\* Bank Area  \* Payer Value Date From  \* Payer Value Date To



## Rfc Execution History

Execution ID	View Description	RFC Description	RFC Parameter	Status	End Timestamp
487	Private	XXX	Bank Area: Area 001, PERIOD: 01/01/1900 – 07/06/2017	Finished	06/06/2017 8.23.54
486	Private			7 ; Finished	06/06/2017 7.21.58
485	Private			7 ; Finished	31/05/2017 20.35.03
484	Private			Finished	31/05/2017 20.33.31
483	Private			Finished with Error	31/05/2017 20.32.11
482	Private			Finished with Error	31/05/2017 20.31.35
481	Private			Finished	31/05/2017 20.24.52
480	Private			Finished	31/05/2017 20.14.05
479	Private			Finished	31/05/2017 15.44.08
478	Private			Finished	31/05/2017 15.35.28

Rows 1 - 10



### III. Applicazione custom J2EE

Al termine del processo di esecuzione della RFC si ricade nella soluzione II, ove i dati resi disponibili su DWH vengono poi integrati su repository OBIEE e consumati accedendo alle relative subject area (in questo caso una differente subject area per ogni RFC configurata).



# Le tre strategie a confronto

Strategia	Punti di Forza	Punti di Debolezza
I – Connection pool OBIEE	Accesso diretto al dato	Performance legate al comportamento del driver ODBC
II – Sorgente ODI	Robustezza, migliori performance	Esecuzione batch, ridondanza del dato
III – Applicazione custom J2EE	Possibilità di eseguire RFC parametriche	Maggiore complessità di implementazione







# THANK YOU!

Francesco Coccia  
[f.coccia@reply.it](mailto:f.coccia@reply.it)

