


IMS Explorer



IMS Application Development (AD) Challenges



Shrinking knowledge base around IMS & hierarchical data base model



Difficult to find DLI programmers



Fewer experienced COBOL and PL/I programmers



Lack of integrated development solutions and tools



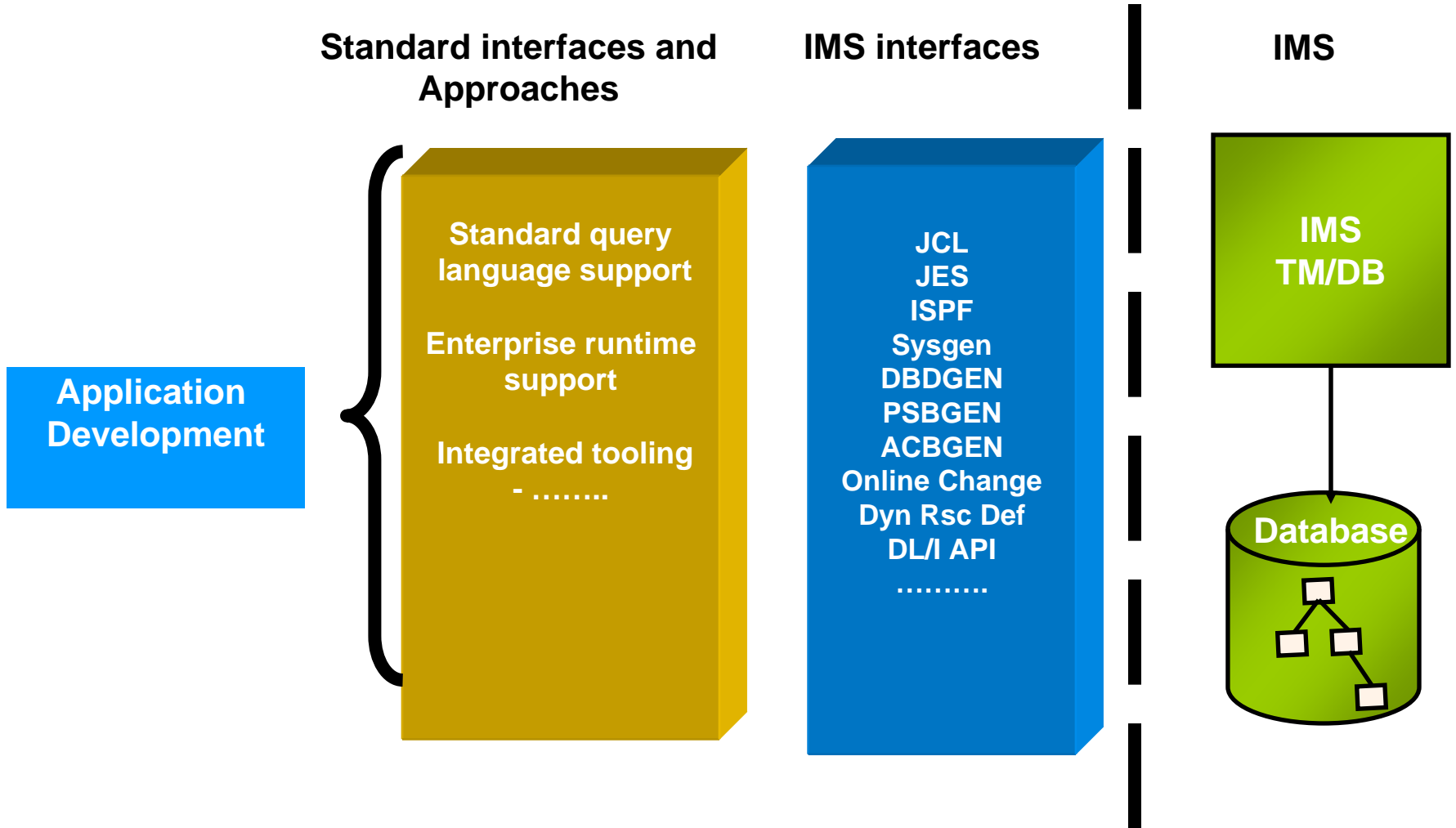
Difficult to test and deploy applications

IMS AD Simplification Strategy

Goal: Reduce programming effort and skills required to create and enhance IMS applications

- ✓ Using industry standards
- ✓ Task automation
- ✓ Consistent look and feel

IMS AD Simplification Strategy



What are the tools supporting our Strategy ?

Eclipse is a software development environment comprising an **integrated development environment (IDE)**



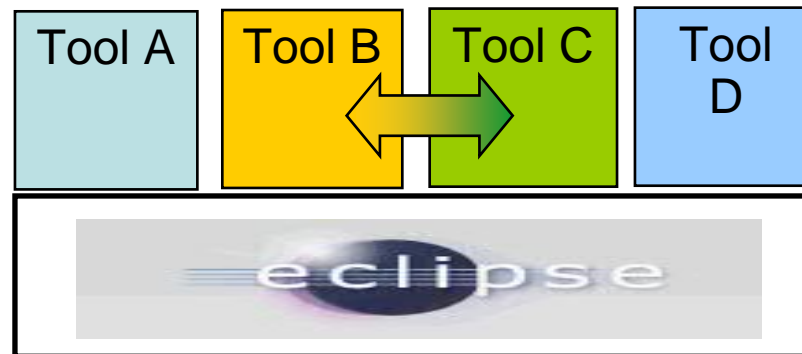
And Plug-in and Shell-Sharing are...



Plug-in:

A **software module** that adds a specific **feature** or service to a larger system.

- Shell sharing:



Introducing IMS Explorer...

Simplifying IMS application development !



Easy-to-use interface



Same look and feel as other IBM products



Complements end-to-end IMS A/D task flow

IMS Explorer Tech Preview

- New Face of IMS
 - GUI-based framework for consistent and integrated tools across environment
- Easier visualization and editing of IMS Database and Program Definitions
 - Graphical display of IMS segment hierarchy and database structure
 - Graphical editors to display/create IMS PSBs
 - Graphical editors to edit/add fields on a DBDs
 - Generation of DBD and PSB source
- Ability to easily access IMS data using SQL statements

IMS Explorer Future Requirements

- Simplification of Application Development
 - Assistance for IMS Application Unit Test, by graphically helping the user to schedule and IMS App program
- Host Connectivity
 - Graphical assistance to FTP file to/from the Host PDS
- Cobol/PLI copybook/strut importers
 - Requires RDz

DBD Graphical Editor

IMS Explorer - DEMOIOD/database/autodb/AUTODB.dbd - Eclipse SDK

File Edit Navigate Search Project Diagram Services Samples Run Window Help

Project Explorer

- allbds
- autpsb1
- ComplexDB
- Demo
- DEMOIOD
 - autpsb11
 - JRE System Library [Java60]
 - imsjava.jar - C:\\$cc71\marilene_Workber
 - database
 - autodb
 - Source
 - AUTODB.dbd
 - autoldb
 - empdb2
 - empldb2
 - index.22
 - program
 - autpsb11
 - Source
 - AUTPSB11.psb
 - test
 - dummy
 - Hospital
 - incompletPSB
 - LX2

Database name: AUTODB Database access type: (HDAM,OSAM)

DBD name: AUTODB

Logical relationships between databases

DEALER
Has 2nd Indexes
Total length: 61
DLRNO
DLRNAME
CITY
ZIP
PHONE
NEWFIELD

MODEL
Total length: 37
MODKEY

SALES
Has Logical Parent
Total length: 85
SALENUM

STOCK
Total length: 46
STKVIN

STOCKSALE
Has Logical Parent

SALESINF
Total length: 15

SALESPER
Has Logical Parent
Total length: 6
EMPNO

EMPSAL
Has Logical Parent
Total length: VLC
DLRNO

EMPL
Total length: 56
EMPNO

EMPLINFO
Total length: 61
STATE

Properties

Property	Value
.Segment statement	
Length (BYTES):	61
Parent segment (PARENT):	0
Segment name (NAME):	DEALER
Source segment (SOURCE):	
List of fields	

Additional properties of a Segment or Field

PSB editor to edit Segment and Fields Sensitivity

The screenshot shows the IMS Explorer application with the PSB editor open for the file `DEMOIOD/program/autpsb11/AUTPSB11.psb`. The interface includes a Project Explorer on the left, a Data Project Explorer, and a main editor area displaying a hierarchical tree of segments and fields. A yellow arrow points to the `Source` folder in the Project Explorer, labeled "Updated PSB source". Another yellow arrow points to the `DEALER` segment in the tree, labeled "Select sensitivity of segments and fields".

The tree structure is as follows:

- DEALER** (Total length: 61)
 - DLRNO
 - DLRNAME
 - CITY
 - ZIP
 - PHONE
 - NEWFIELD
- MODEL** (Total length: 37)
 - MODTYPE
 - MODKEY
 - MAKE
 - MODEL
 - YEAR
 - MSRP
 - COUNT
- SALES** (Total length: 131)
 - STKVIN
 - COLOR
 - PRICE
 - LOT
 - WRNTY
 - SALENUM
- STOCK** (Total length: 46)
 - STKVIN
 - COLOR
 - PRICE
 - LOT
 - WRNTY
- SALESIF** (Total length: 15)
 - QUOTA
 - SALESYTD
 - COMSSION
- EMPLINFO** (Total length: 61)
 - ADDRESS
 - STREET
 - CITY
 - STATE
 - ZIP
- ORDER** (Total length: 74)
 - ORDNBR
 - LASTNAME
 - FIRSTNAME
 - DATE
 - TIME
- SALES** (Total length: 131)
 - STKVIN
 - COLOR
 - PRICE
 - LOT
 - WRNTY
 - SALENUM
- SALES** (Total length: 131)
 - STKVIN
 - COLOR
 - PRICE
 - LOT
 - WRNTY
 - SALENUM
- SALES** (Total length: 131)
 - STKVIN
 - COLOR
 - PRICE
 - LOT
 - WRNTY
 - SALENUM
- SALES** (Total length: 131)
 - STKVIN
 - COLOR
 - PRICE
 - LOT
 - WRNTY
 - SALENUM

The Properties window at the bottom shows the following details for the selected segment:

Property	Value
.Segment statement	
Length (BYTES):	122
Parent segment (PARENT):	SALESIF

SQL Access to IMS Database

The screenshot shows the Eclipse IDE interface for SQL access to an IMS database. The main window displays a SQL statement: `SELECT PCB01.HOSPITAL.HOSPNAME, PCB01.PATIENT.PATNAME, PCB01.HOSPITAL.HOSPCODE FROM PCB01.HOSPITAL, PCB01.PATIENT`. A yellow callout box points to this statement, stating "SQL statement automatically generated".

Below the statement is the SQL Builder dialog. It shows two tables: **HOSPITAL** and **PATIENT**. In the **HOSPITAL** table, the **HOSPCODE** and **HOSPNAME** columns are selected. In the **PATIENT** table, the **PATNAME** column is selected. A yellow callout box points to this dialog, stating "SQL Builder: Point and click to generate a SQL statement".

Below the dialog is the **Columns** tab, which contains a table with the following columns: **Column**, **Alias**, **Output**, **Sort Type**, and **Sort Order**. The table contains three rows:

Column	Alias	Output	Sort Type	Sort Order
PCB01.HOSPITAL.HOSPNAME		<input checked="" type="checkbox"/>		
PCB01.PATIENT.PATNAME		<input checked="" type="checkbox"/>		
PCB01.HOSPITAL.HOSPCODE		<input checked="" type="checkbox"/>		

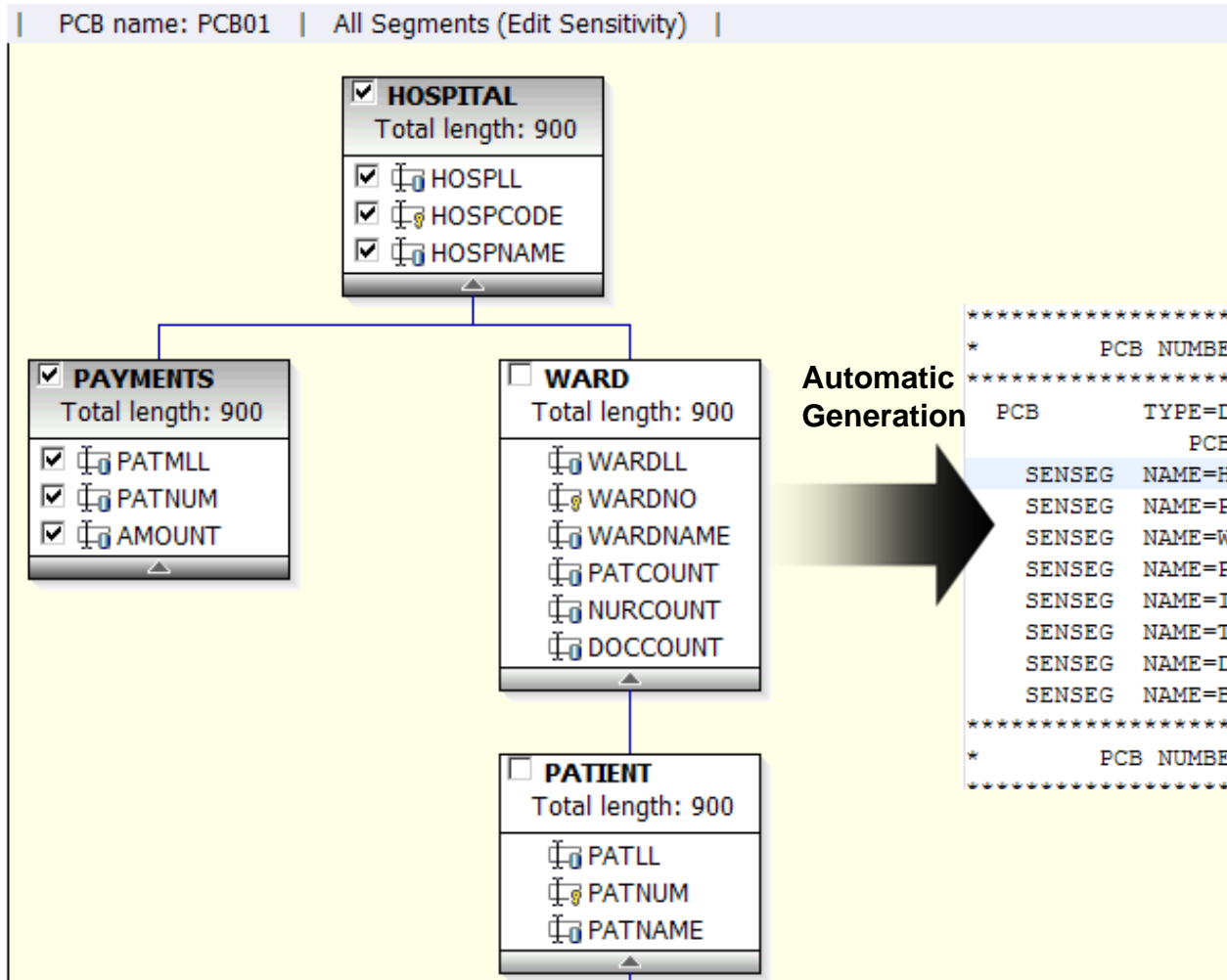
Below the columns table is the **SQL Results** window, which shows a table of results from the SQL statement. A yellow callout box points to this window, stating "Results from the SQL statement". The results table has the following columns: **HOSPLL**, **HOSPCODE**, **HOSPNAME**, **HOSPITAL_HOSPCODE**, and **WAR**. The results are as follows:

HOSPLL	HOSPCODE	HOSPNAME	HOSPITAL_HOSPCODE	WAR
1	R.121001000...	GOOD SAMA...	R.1210010000A	0004
2	R.121001000...	GOOD SAMA...	R.1210010000A	0004
3	R.121001000...	GOOD SAMA...	R.1210010000A	0004
4	R.121001000...	GOOD SAMA...	R.1210010000A	0007
5	R.121002000...	SANTA TERE...	R.1210020000A	0002
6	R.121004000...	NEW ENGLA...	R.1210040000A	0011
7	R.121004000...	NEW ENGLA...	R.1210040000A	0011
8	R.121004000...	NEW ENGLA...	R.1210040000A	0011
9	R.121004000...	NEW ENGLA...	R.1210040000A	0070
10	R.121004000...	NEW ENGLA...	R.1210040000A	0070

The Data Source Explorer on the left shows the database structure, including the **PCB01** schema and its tables: **BILLING**, **DOCTOR**, **HOSPITAL**, **ILLNESS**, **PATIENT**, **PAYMENTS**, and **TREATMNT**.

IMS Explorer – in practice

IMS Explorer.... In practice



Automatic Generation



```

*****
*          PCB NUMBER 5          DB  DEDBJN21
*****
PCB          TYPE=DB, DBDNAME=DEDBJN21, POS=M, PROCOPT=A, KEYLEN=
          PCBNAME=PCB01
SENSEG      NAME=HOSPITAL, PARENT=0
SENSEG      NAME=PAYMENTS, PARENT=HOSPITAL,
SENSEG      NAME=WARD, PARENT=HOSPITAL
SENSEG      NAME=PATIENT, PARENT=WARD
SENSEG      NAME=ILLNESS, PARENT=PATIENT
SENSEG      NAME=TREATMNT, PARENT=ILLNESS
SENSEG      NAME=DOCTOR, PARENT=TREATMNT
SENSEG      NAME=BILLING, PARENT=PATIENT
*****
*          PCB NUMBER 6          DB  IVPDB1
*****
  
```

Generated PSB source

```

=1 Alt=0
*****
*****
,KEYLEN=26, C
  
```

```

REPEAT      6 ADD
RIGHT      12 JOIN
  
```

IMS Explorer.... In practice



Data - Demo/Script3.sql - Eclipse SDK

File Edit Navigate Search Project SQL Diagram Services Samples Run Window Help

Data Project Explorer

- Demo (IMS Hospital:jdbcims://ecdv13.vmec.svl.ibm.com)
 - SQL Scripts
 - Script1.sql
 - Script2.sql
 - Script3.sql
 - XML

SQL Editor

```
SELECT PCB01.HOSPITAL.HOSPNAME, PCB01.PATIENT.PATNAME, PCB01.HOSPITAL.HOSPCODE
FROM PCB01.HOSPITAL, PCB01.PATIENT
```

Table Selection

HOSPITAL

- HOSPCODE
- HOSPLL
- HOSPNAME

PATIENT

- HOSPITAL_HOSPC
- WARD_WARDNO
- PATNUM
- PATLL
- PATNAME

DISTINCT

Columns Conditions Groups Group Conditions

Column	Alias	Output	Sort Type	Sort Order
PCB01.HOSPITAL.HOSPNAME		<input checked="" type="checkbox"/>		
PCB01.PATIENT.PATNAME		<input checked="" type="checkbox"/>		
PCB01.HOSPITAL.HOSPCODE		<input checked="" type="checkbox"/>		

Properties SQL Results

Type query expression here

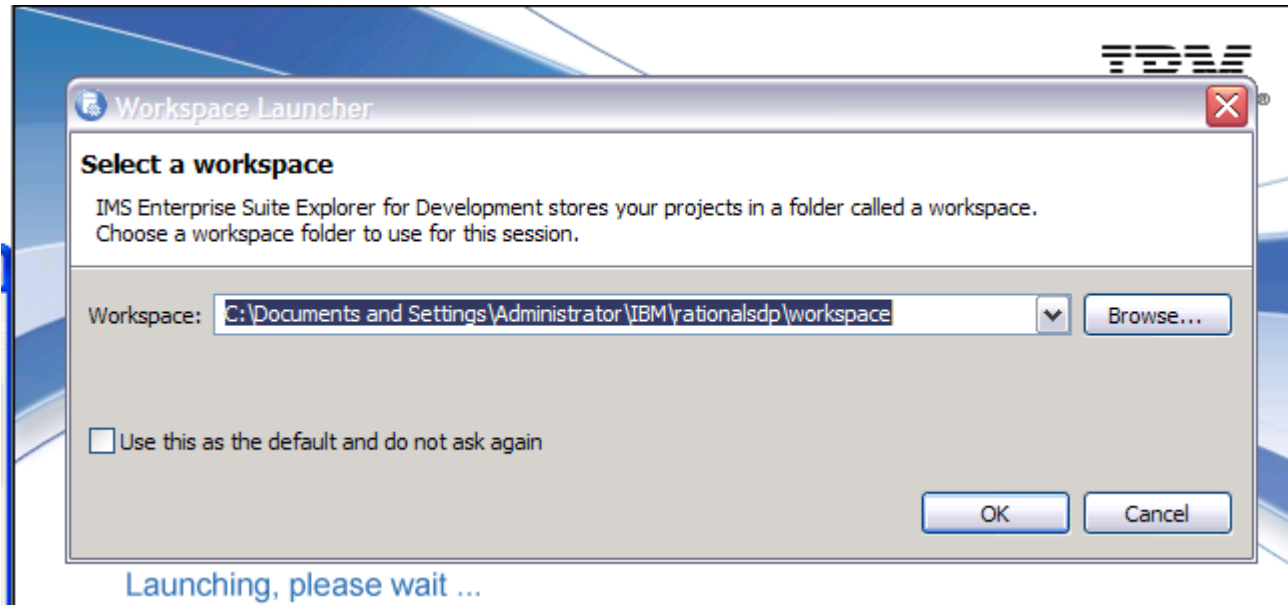
Status	Operation	Date	Connecto...
✓	Succeed Return All R...	10/14/10 11...	IMS Hospital
✓	Succeed "PCB01".H...	10/14/10 11...	IMS Hospital
✓	Succeed "PCB01".H...	10/14/10 11...	IMS Hospital
✓	Succeed SELECT PCB...	10/14/10 11...	IMS Hospital
✓	Succeed SELECT PCB...	10/14/10 11...	IMS Hospital
✓	Succeed SELECT PCB...	10/14/10 11...	IMS Hospital
✗	Failed SELECT PCB...	10/14/10 11...	IMS Hospital
✓	Succeed SELECT PCB...	10/14/10 11...	IMS Hospital

	HOSPLL	HOSPCODE	HOSPNAME	HOSPITAL_HOSPCODE	WAR
1		R121001000...	GOOD SAMA...	R1210010000A	0004
2		R121001000...	GOOD SAMA...	R1210010000A	0004
3		R121001000...	GOOD SAMA...	R1210010000A	0004
4		R121001000...	GOOD SAMA...	R1210010000A	0007
5		R121002000...	SANTA TERE...	R1210020000A	0002
6		R121004000...	NEW ENGLA...	R1210040000A	0011
7		R121004000...	NEW ENGLA...	R1210040000A	0011
8		R121004000...	NEW ENGLA...	R1210040000A	0011
9		R121004000...	NEW ENGLA...	R1210040000A	0070
10		R121004000...	NEW ENGLA...	R1210040000A	0070

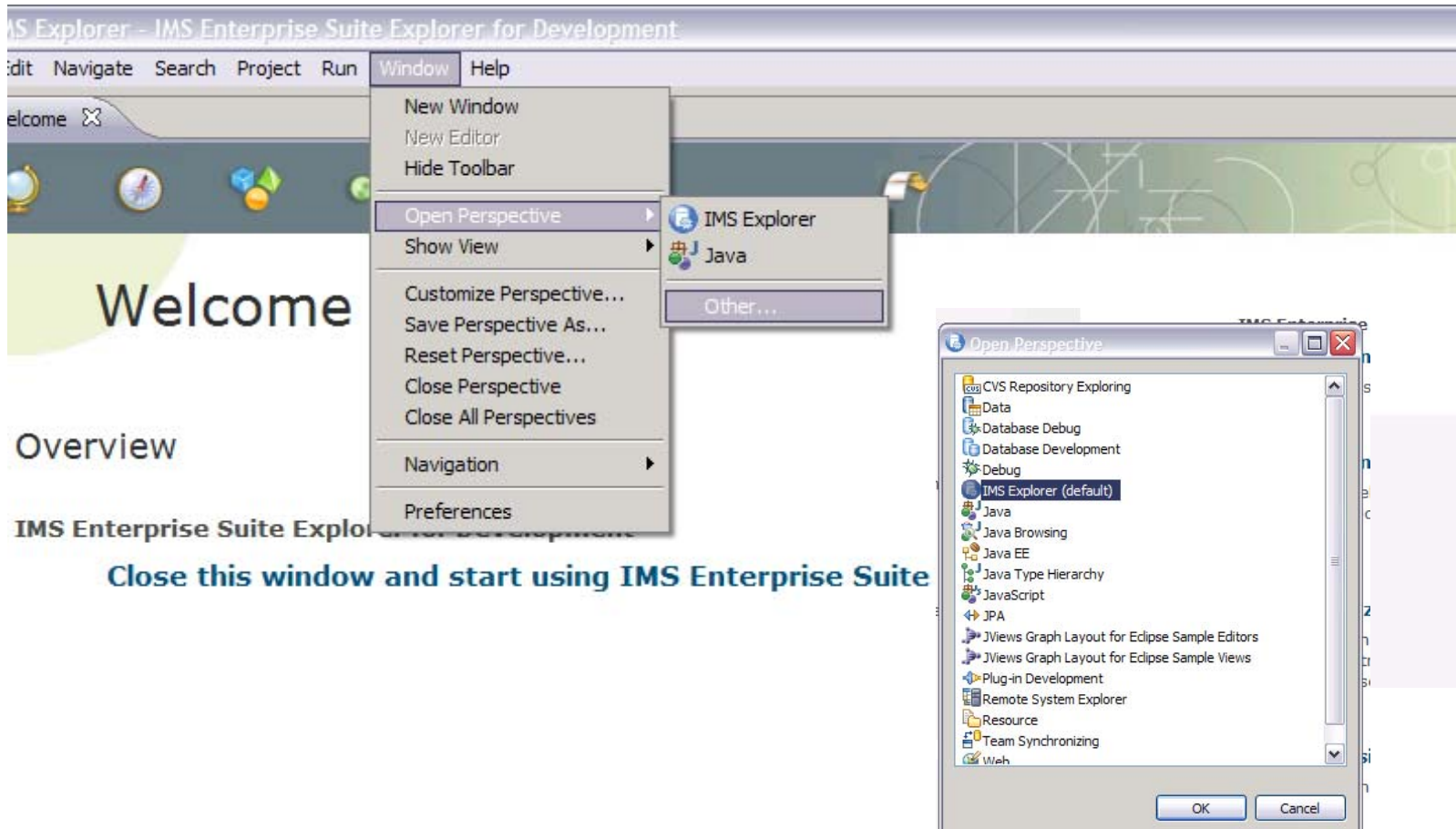
Database type: IMS_V11, Current profile: IMS Hospital, Database: IMS Hospital, connected

Start to Finish

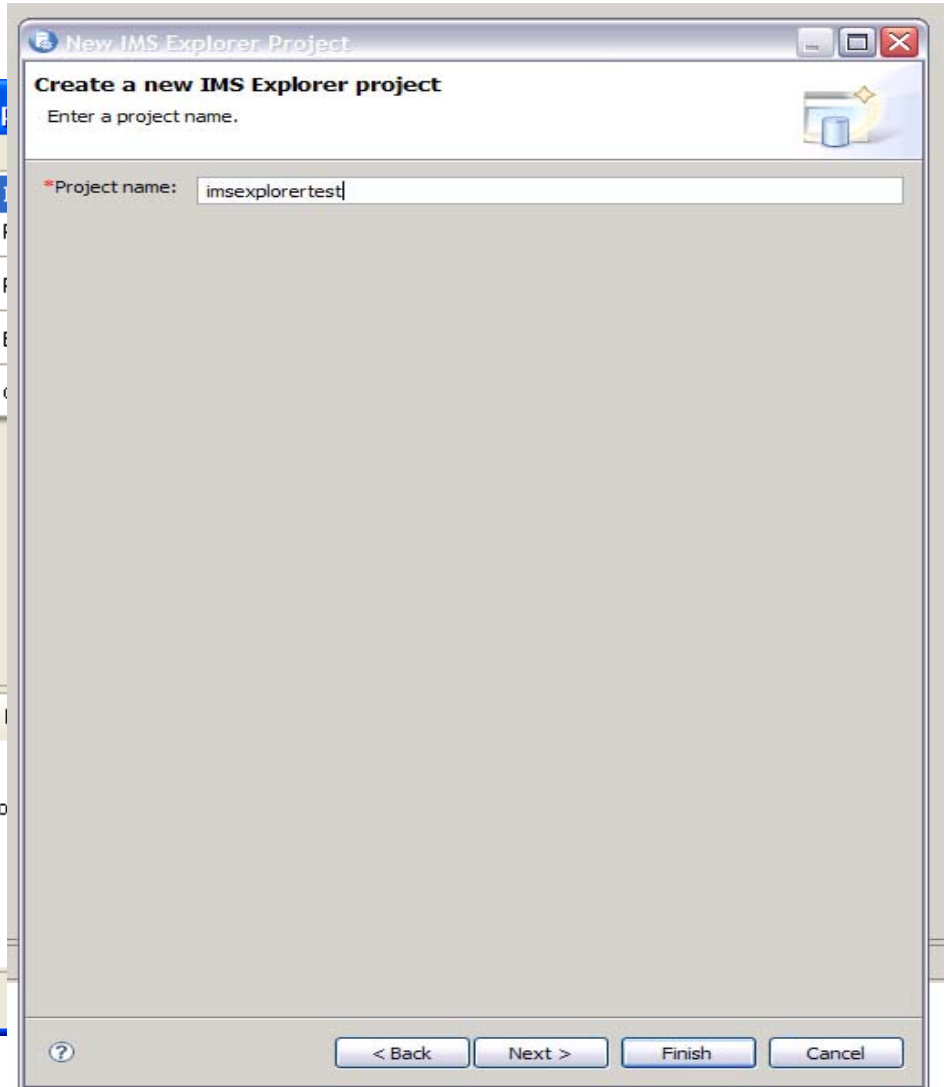
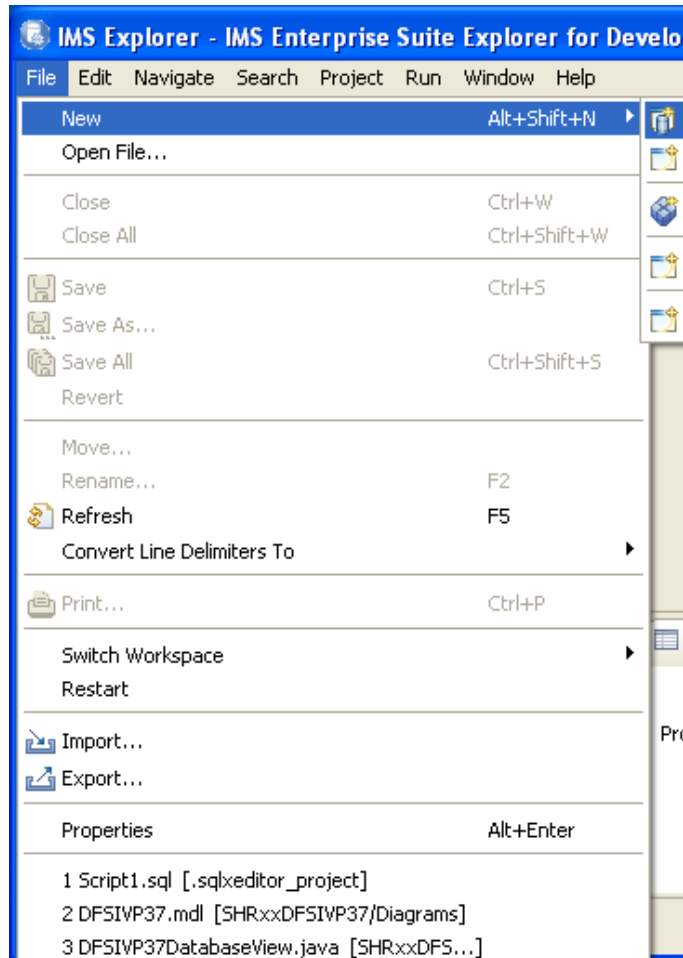
Start the IMS Explorer



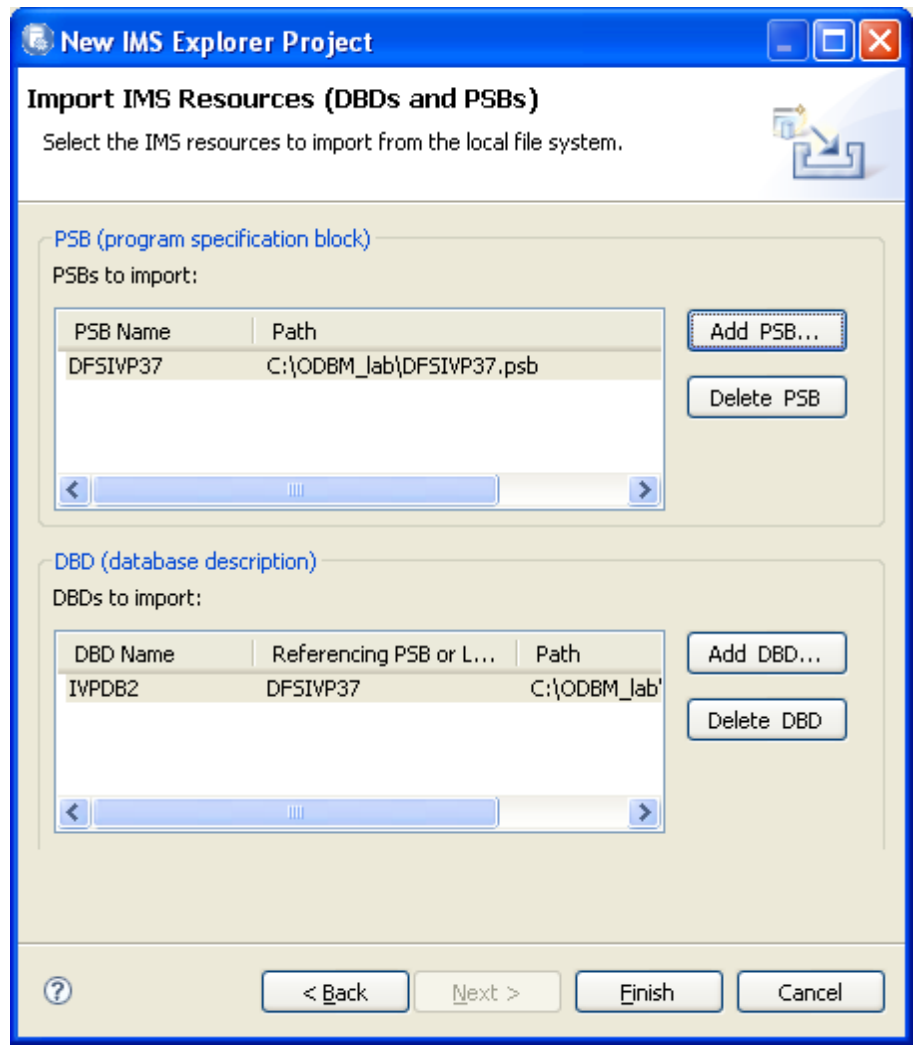
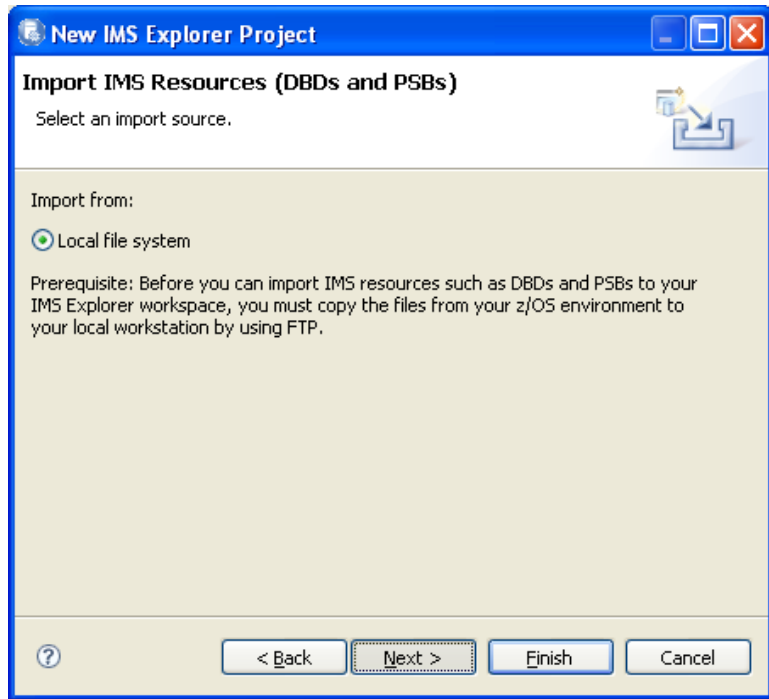
Open the IMS Explorer Perspective



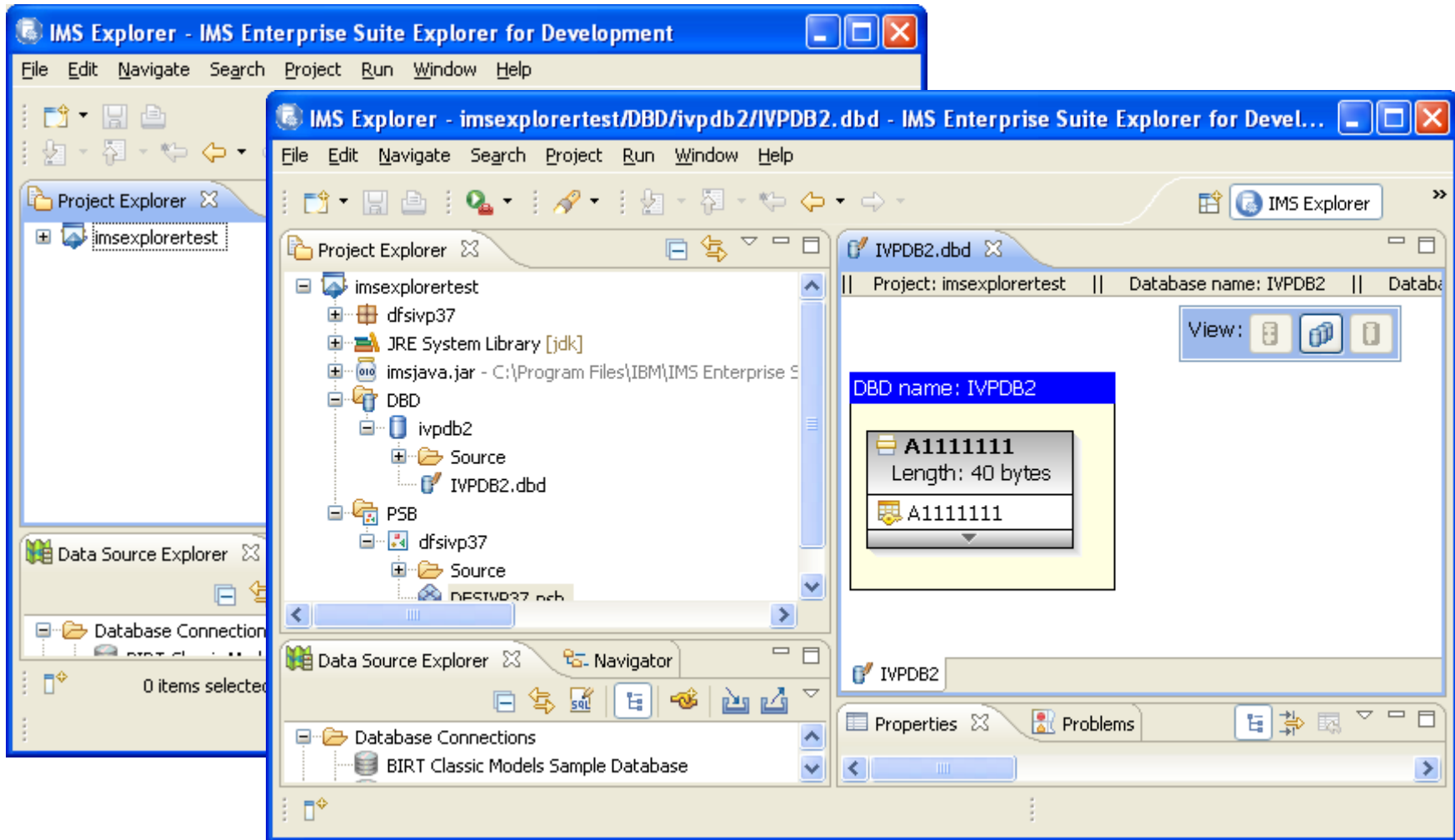
Create a New IMS Explorer Project



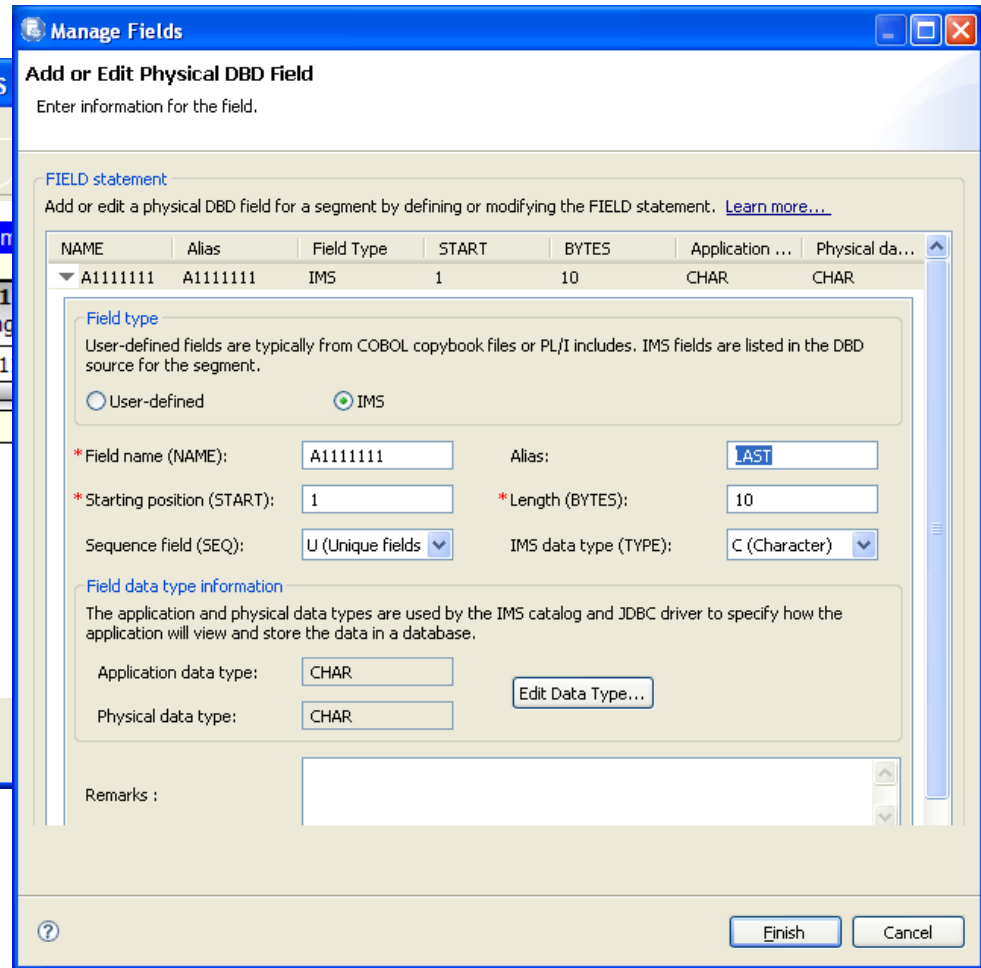
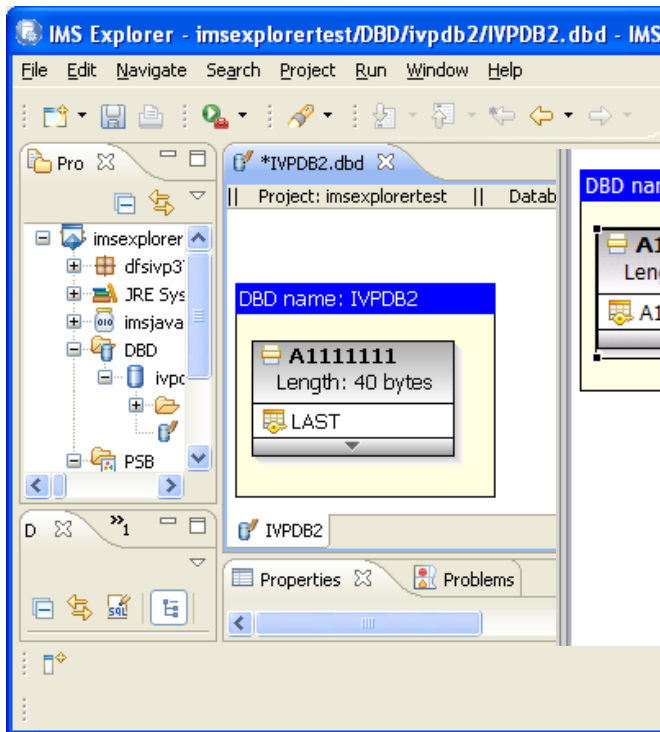
Import the IMS resources from a Local File system (assumes you previously retrieved the PSB and DBD sources from the mainframe)



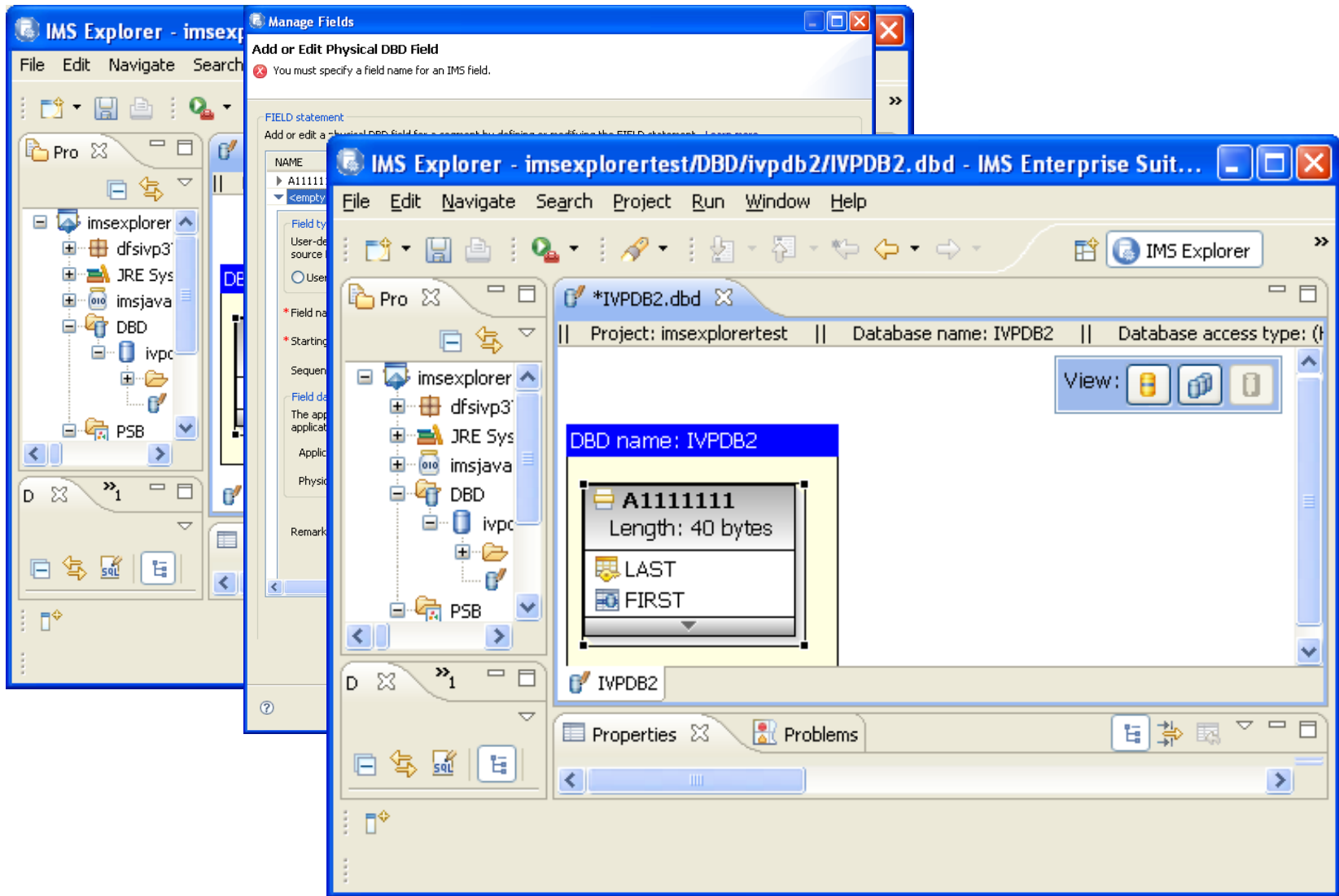
Open up the project you created and navigate to the DBD



You can edit the information, e.g., rename an alias



You can add fields to the segment that were not originally defined in the dbd



You can also define fields up to the segment size

DBD name: IVPDB2

A1111111
Length: 40 bytes

- LAST
- FIRST
- PHONE
- ZIP
- FILLER

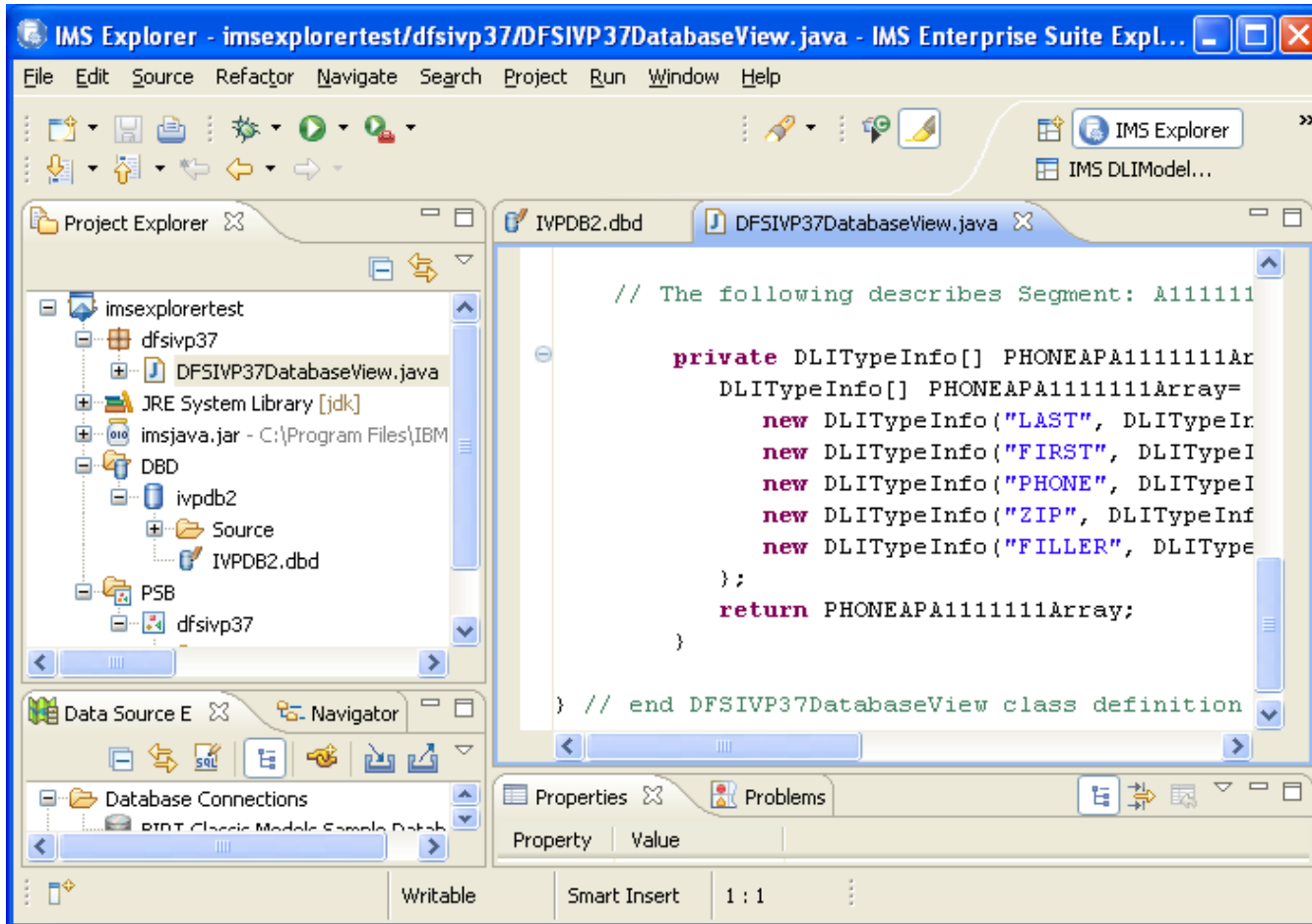
Manage Fields
Add or Edit Physical DBD Field
Enter information for the field.

FIELD statement
Add or edit a physical DBD field for a segment by defining or modifying the FIELD statement. [Learn more...](#)

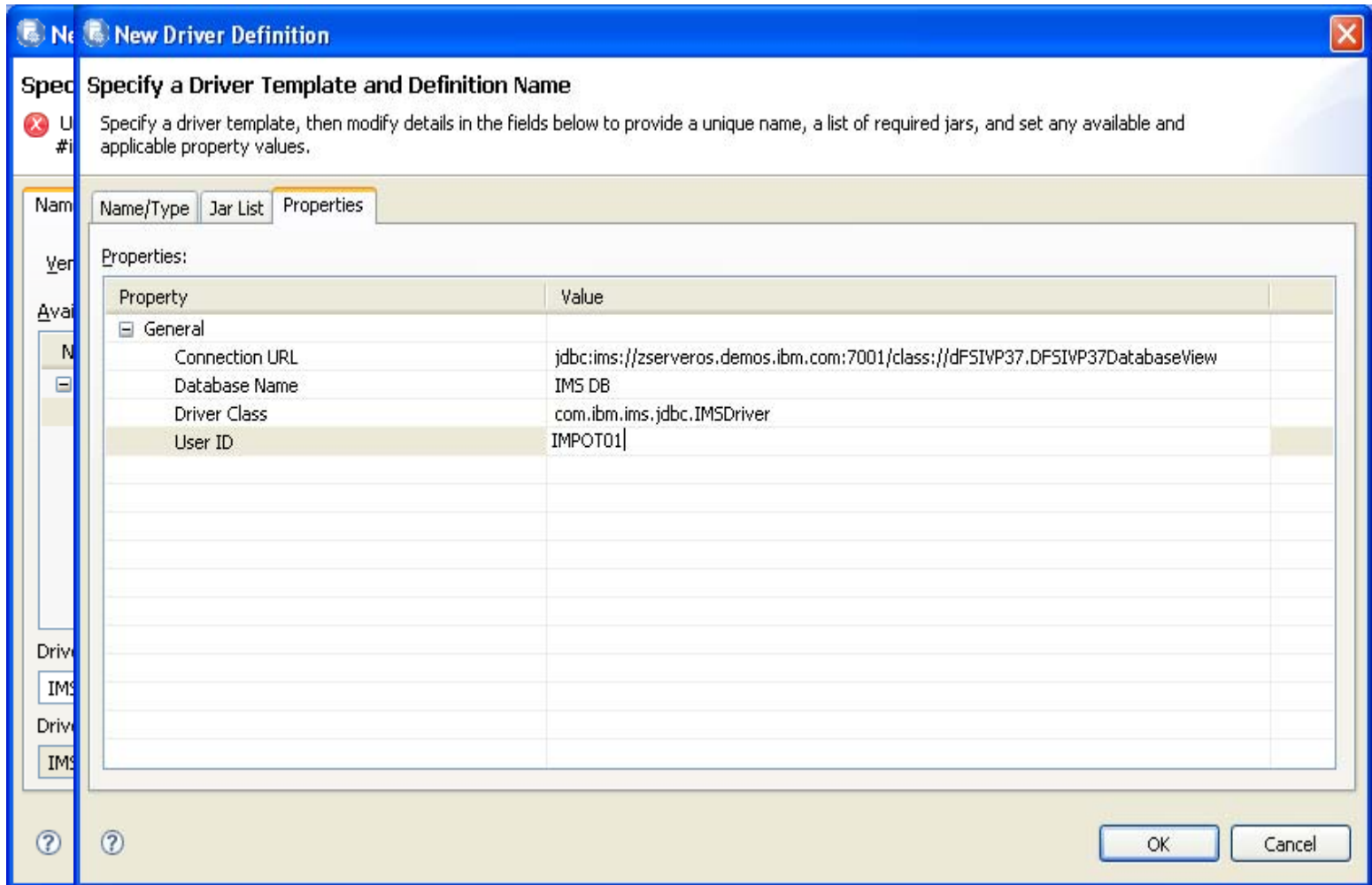
NAME	Alias	Field Type	START	BYTES	Application ...	Physical da...
▶ A1111111	LAST	IMS	1	10	CHAR	CHAR
▶ FIRST	FIRST	IMS	11	10		
▶ PHONE	PHONE	IMS	21	10		
▶ ZIP	ZIP	IMS	31	7		
▶ FILLER	FILLER	IMS	38	3		

Buttons: Add FIELD Statement, Delete FIELD Statement, Finish, Cancel

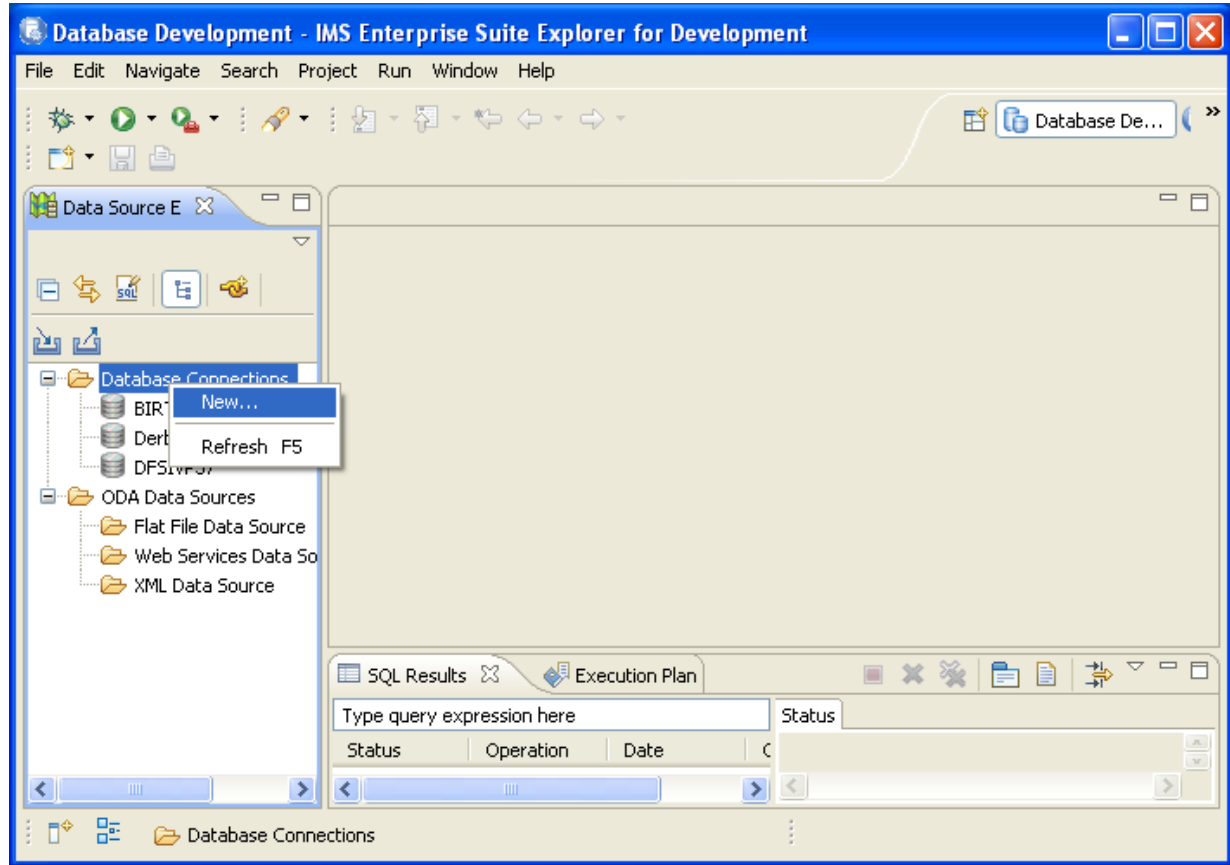
Which automatically generates the metadata



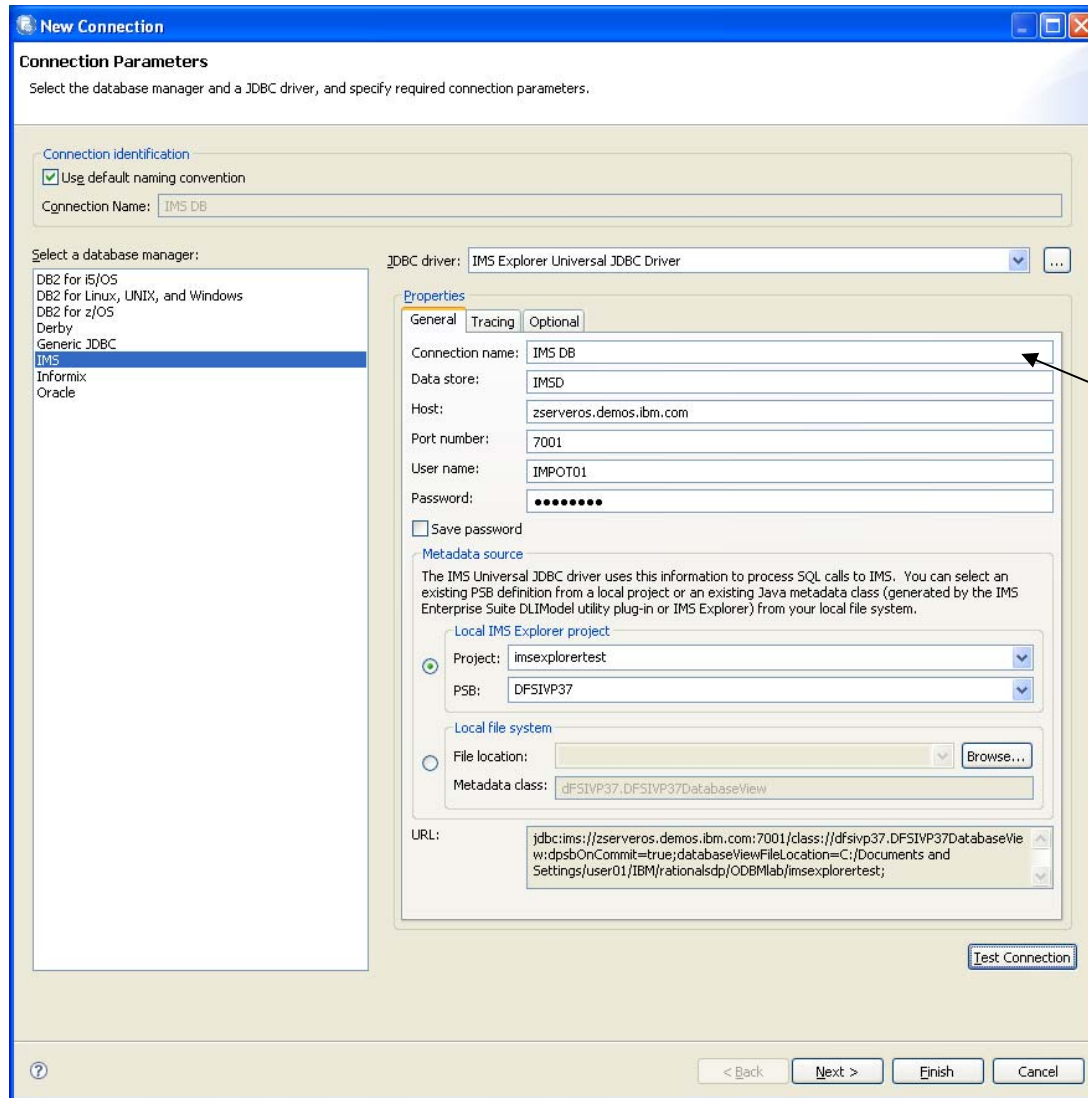
Now you need to add the IMS universal driver



And connect to the IMS database



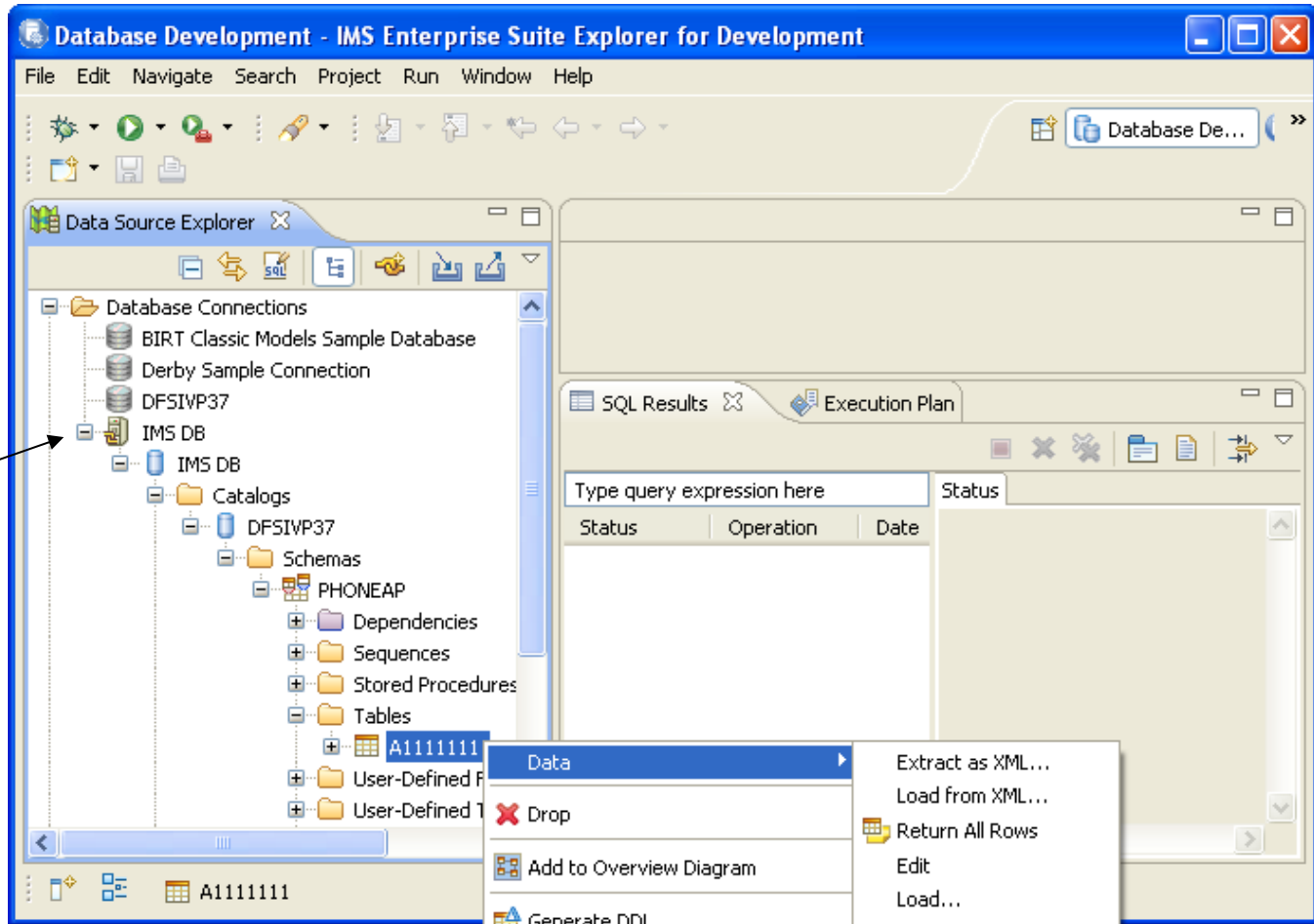
Select IMS as the database manager and choose the driver you defined



Note the Connection name that you specify

The connection is shown in the Data Source Explorer (DSE) window and In this example the specific example

Connection name that you specified earlier

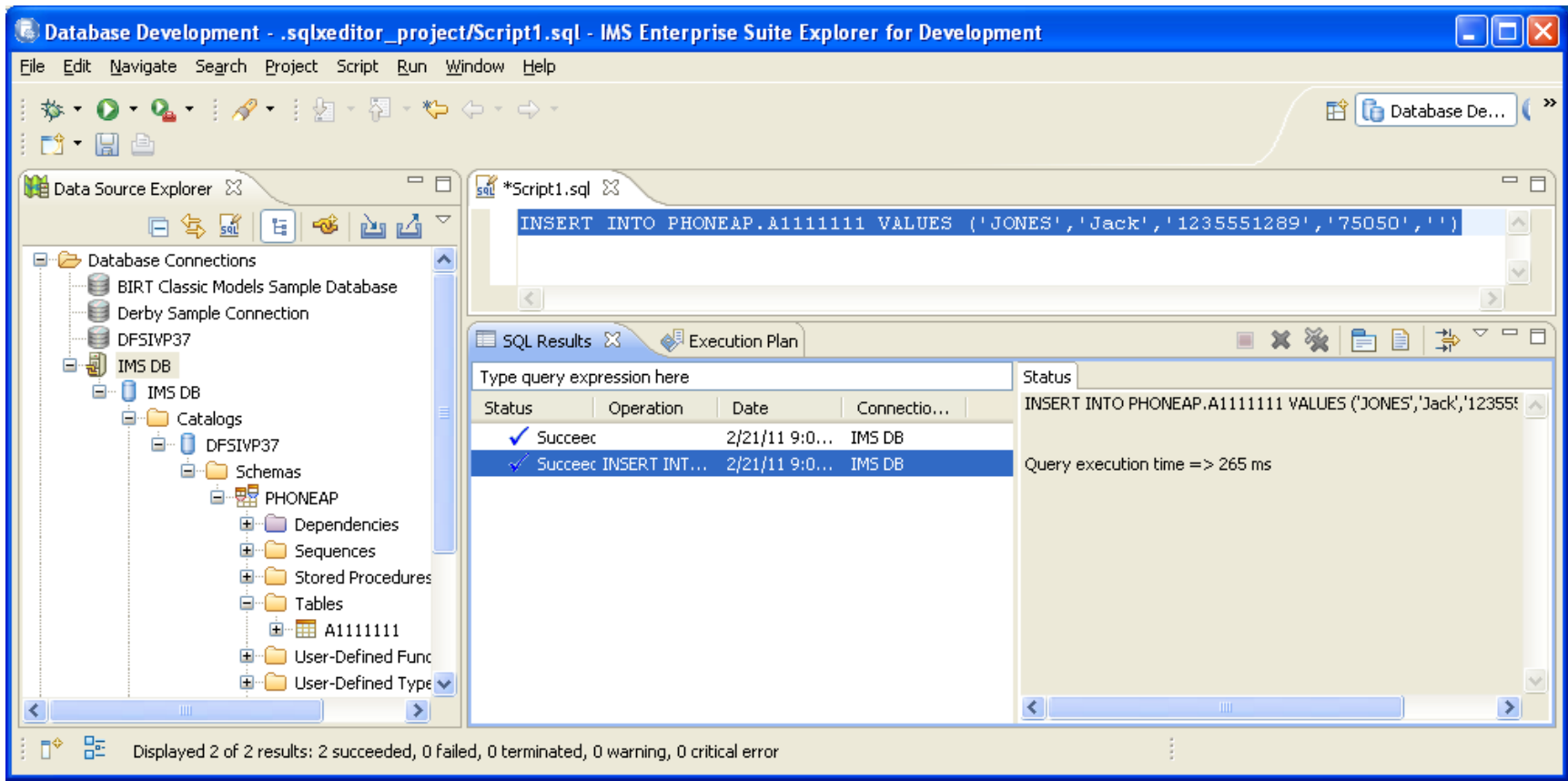


Select “sample contents”

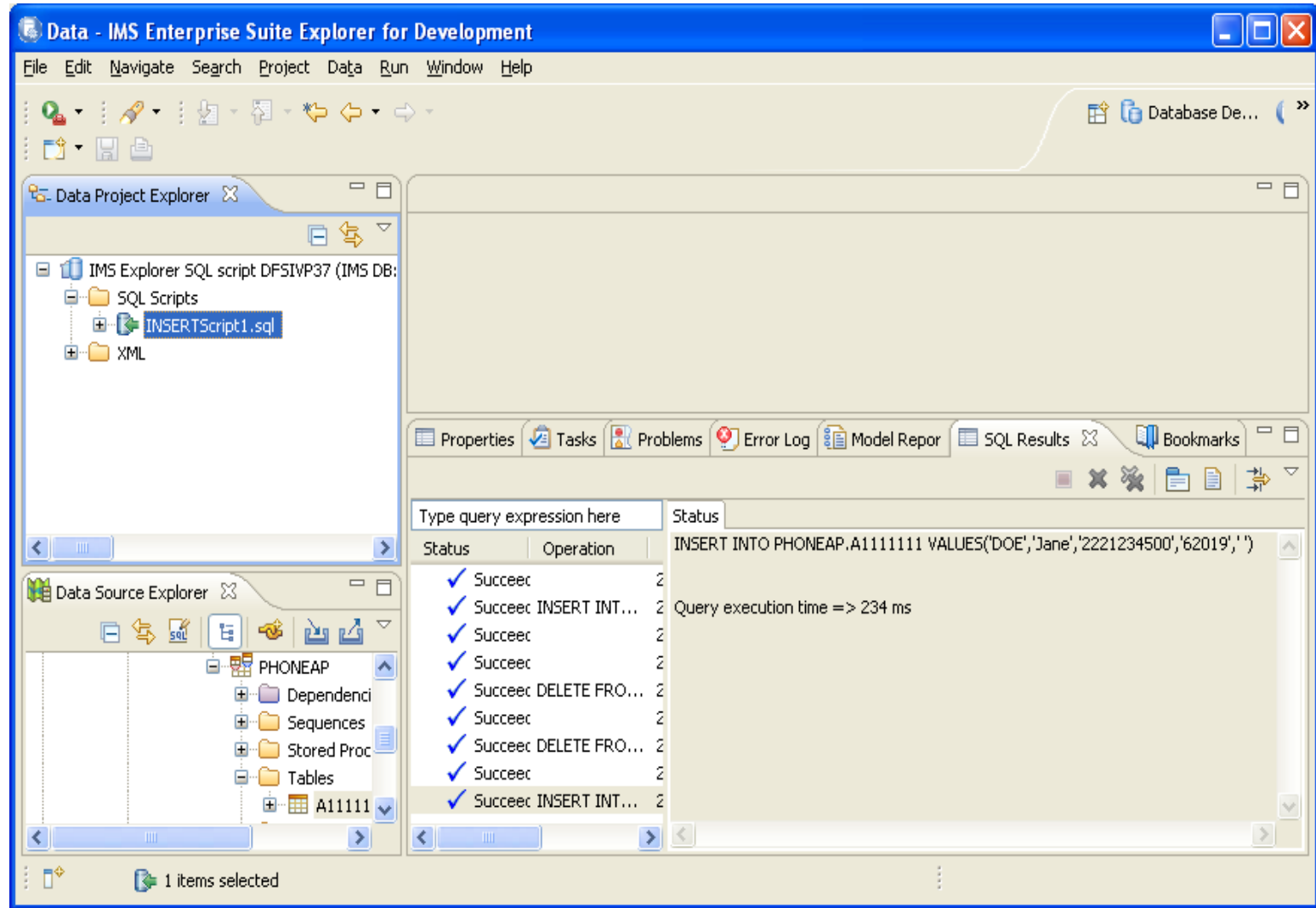
The screenshot displays the IBM Database Development interface. On the left, the Data Source Explorer shows a tree view of database connections, including 'IMS DB' and 'Catalogs'. The main window shows the 'SQL Results' tab with a table of data. The table has columns: LAST, FIRST, PHONE, ZIP, and FILL. The data rows are numbered 1 through 11. The status bar at the bottom indicates 'Displayed 1 of 1 results: 1 succeeded, 0 failed, 0 terminated, 0 warning, 0 critical error'.

Status	Operation	LAST	FIRST	PHONE	ZIP	FILL
✓ Succeed	2	LAST6	FIRST6	8-11...	D...	
		THOMAS	TEETHU	1111...	1...	
		JEFFERS D	ENISE	8565...	0...	
		LAST1	FIRST1	8-11...	D...	
		Simon	Rich	3096...	6...	
		CLANTON	STEVE	5244...	7...	
		NEWMAN	JOHN	9999...	6...	
		LAST2	FIRST2	8-11...	D...	
		TESTLast	FIRST	1113...	7...	
		LAST3	FIRST3	8-11...	D...	
		LAST5	FIRST5	8-11...	D...	

highlight the database connection, right mouse click and select **New SQL Script**.
When the window opens, key in a request, e.g.
INSERT INTO PHONEAP.A1111111 VALUES ('xxxxx','yyyyy','1234567890','11111','')



You can also create a data explorer project and create scripts that are saved



Data - IMS Enterprise Suite Explorer for Development

File Edit Navigate Search Project Data Run Window Help

Database De... >>

Data Project Explorer

- IMS Explorer SQL script DFSIVP37 (IMS DB:
 - SQL Scripts
 - INSERTScript1.sql
 - QUERYScript.sql
 - XML

Data Source Explorer

- PHONEAP
 - Dependenci
 - Sequences
 - Stored Proc
 - Tables
 - A11111

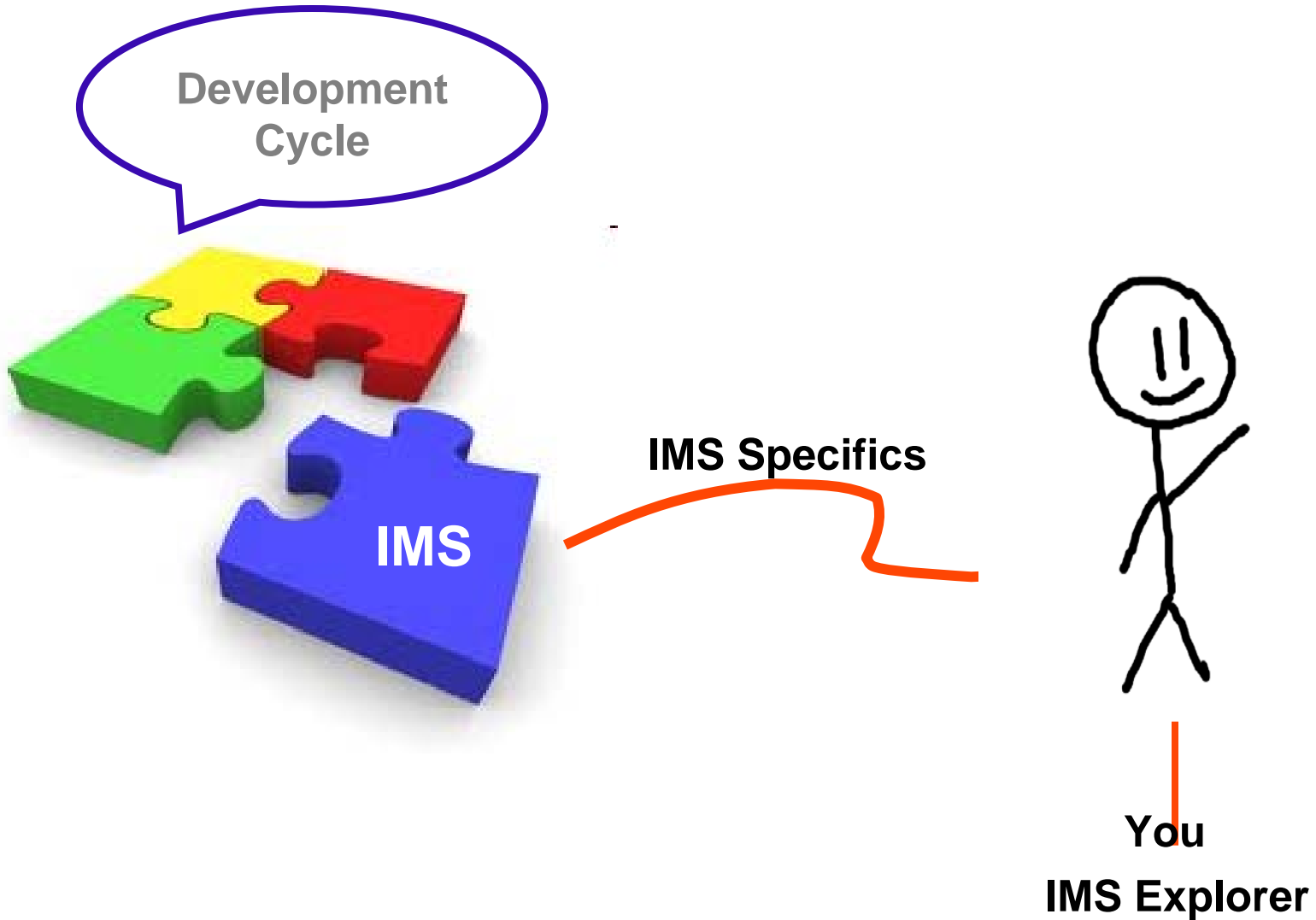
Properties Tasks Problems Error Log Model Repor SQL Results Bookmarks

Type query expressic	Status	Result1				
		LAST	FIRST	PHONE	ZIP	FILLER
✓ Succeed	4	LAST1	FIRST...	8-111-...	D0...	
✓ Succeed IN	5	Simon	Rich	30966...	61...	
✓ Succeed	6	JONES	Jack	12355...	75...	
✓ Succeed	7	NEWMAN	JOHN ...	99999...	60...	
✓ Succeed	8	LAST2	FIRST...	8-111-...	D0...	
✓ Succeed DE	9	TESTLast	FIRST...	11133...	75...	
✓ Succeed	10	LAST3	FIRST...	8-111-...	D0...	
✓ Succeed DE	11	LAST5	FIRST...	8-111-...	D0...	
✓ Succeed	12	DOE	Jane	22212...	62...	
✓ Succeed	13	LAST4	FIRST...	8-111-...	D0...	
✓ Succeed IN	14	jeffers d	ENISE...	85653...	08...	

Total 14 records shown

Displayed 10 of 10 results: 10 succeeded, 0 failed, 0 terminated, 0 warning, 0 critical error

IMS Explorer in the bigger AD picture!



IMS Explorer ... learn more?



Sign up for the IMS Explorer Technical Preview

www.ibm.com/ims

→ Click on IMS Enterprise Suite

Home Solutions Services Products Support & downloads My IBM Welcome [IBM Sign in] [Register]

IBM Software > Information Management > IMS family >

IMS Enterprise Suite

The IMS™ Enterprise Suite, part of the [IMS SOA Integration Suite](#), is a set of components that support open integration technologies to enable new application development and extend access to IMS transactions and data. The IMS Enterprise Suite provides user-friendly standard interfaces, simplifies IMS metadata generation, and enables IMS business event data and monitoring. The IMS Enterprise Suite also eases and expands IMS development (including Java™ and XML), administration, and access. Graphical user interfaces and standards-based programming models are provided through tooling support from the WebSphere® and Rational® product families.

The IMS Enterprise Suite is available for both z/OS® and distributed platforms, and is a no-cost product for unlimited installs. The IMS Enterprise Suite components are designed to complement [IMS 11](#), but also support [IMS 10](#). The IMS Enterprise Suite V1.1 is upward compatible from IMS SOAP Gateway Version 10 and the IMS Version 10 DLIModel utility plug-in, allowing existing applications and data to be used without change. Migration and coexistence support is provided for IMS Version 10 users. Review the Preventative Service Planning (PSP) information for the details.

Downloads

- [IMS Enterprise Suite](#)
- [IMS Explorer Technology Preview](#)
- [Rational Developer for System z V7.6 \(IMS Edition\)](#)

Documentation

- [Release notes with latest release and APARs information](#)
- [IMS Enterprise Suite](#)

Related links

- Business Partners
- Developers
- System z