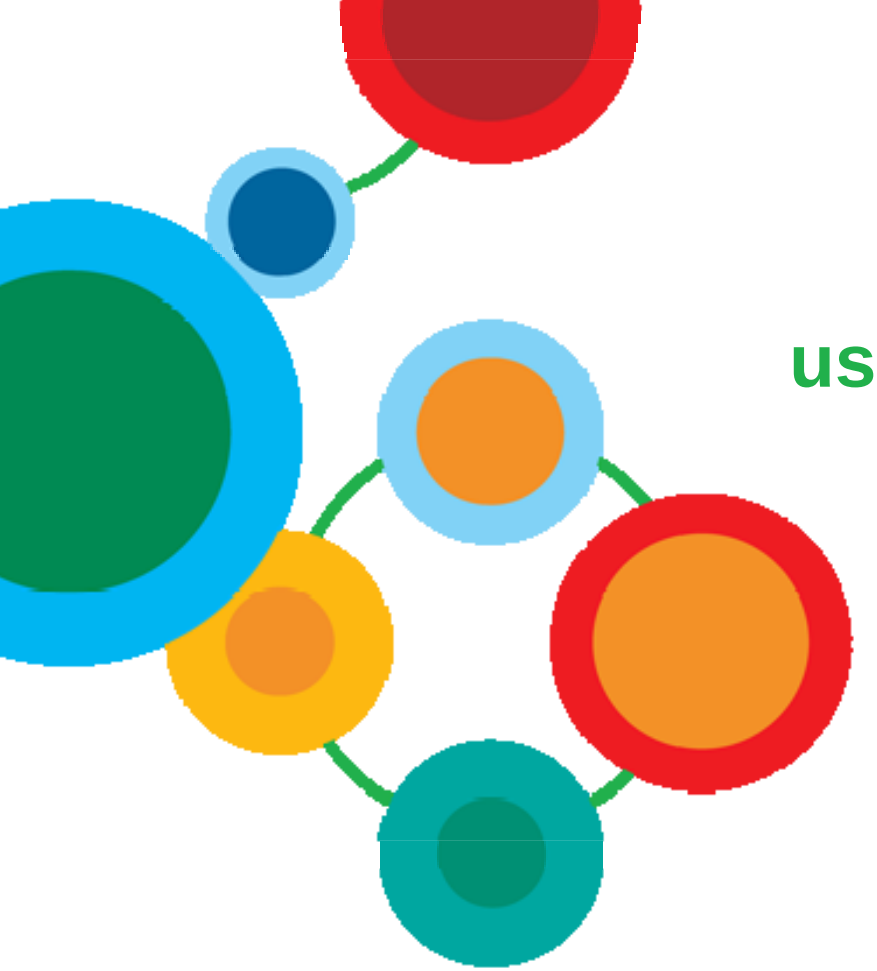


Caterpillar and IMS: Application Modernization using the IMS SOAP Gateway

Session Number 2890A

Steve Clanton, Caterpillar
Josh Newell, IBM

IBM Software
Information On Demand 2011

How Did We Get Here? (A little bit of history)

A quote from "IBM 100 – ICONS of Progress – Information Management System"

'In 1966, Watts and the 12-member IBM team, along with 10 members of Rockwell and three members from Caterpillar Inc.—whose engines would supply the power for communications between the spacecraft and all of the NASA tracking stations around the world—went to work designing and developing the system that was called the ICS/DL/I." ... "The Apollo 11 Moon mission was one of the first major projects for which computerized information management tools were essential to its success. Following that achievement, ICS was renamed the Information Management System and was released commercially for IBM mainframes, starting with the IBM System/360."

The Challenge Today.

- Coordinate the tracking of manufacturing materials between applications based in IMS and applications based on Distributed Platforms.
  - A full function, 40 year old, legacy IMS application, tracking materials in many factories and warehouses around the world
  - Many distributed applications, both purchased and in-house developed, tracking materials in a single or small group of factories.
  - Legacy FTP of files to communicate between the systems
  - Manual data entry to update systems and process exceptions

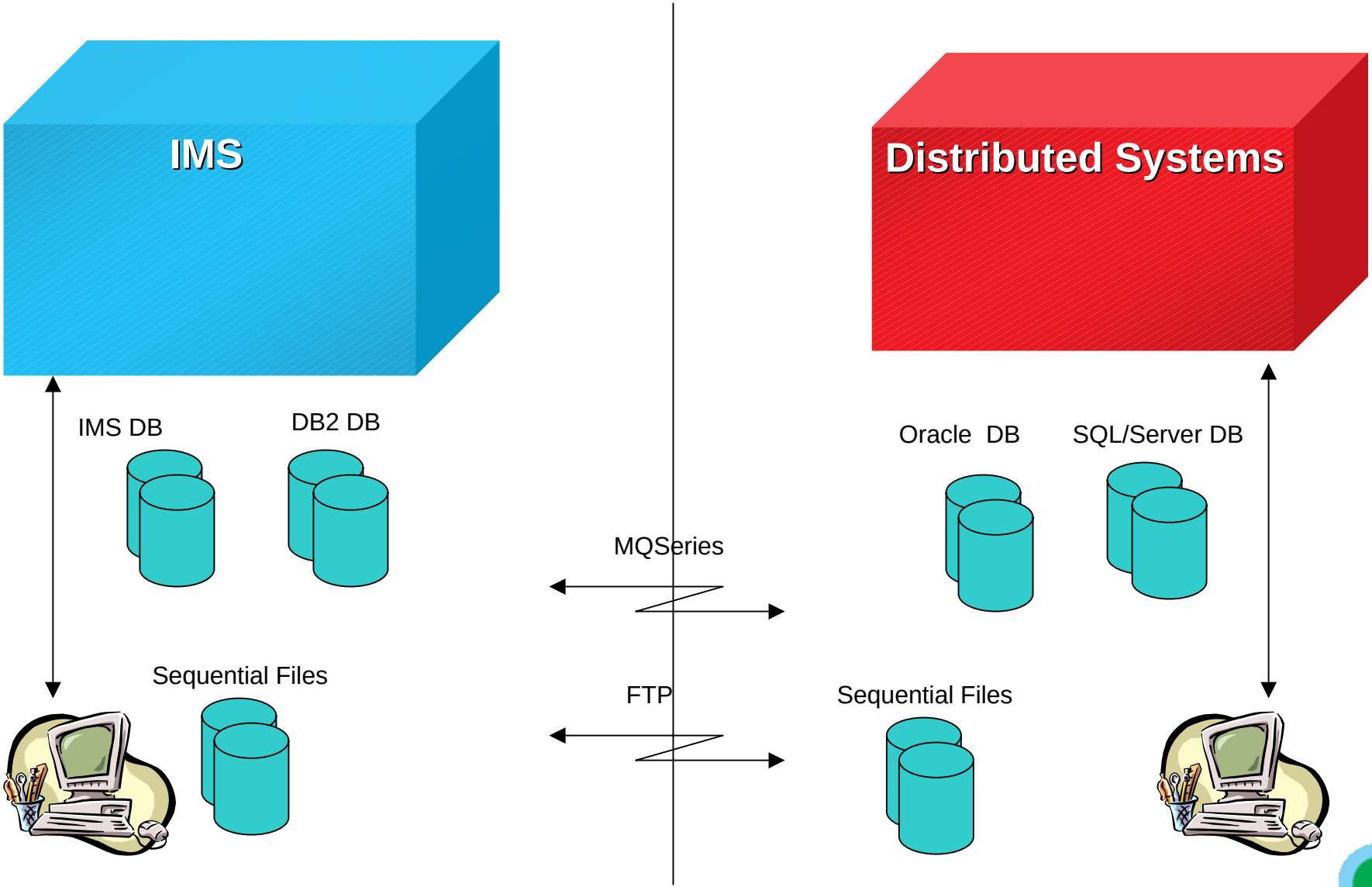
Today's Environment.

Caterpillar currently has seven Production IMS systems, four are DB/DC and three are DBCTL. We also have five application development and test systems, two systems quality environments, and four "sandbox" systems. They are split between North America and Europe. All DB/DC systems also use DB2 and MQSeries subsystems, and use APPC and IMS Connect.

Production Statistics

Databases	Over 10,000
Programs	Over 40,000
Transactions	Over 75,000
Terminals	Over 150,000
Printers	Over 20,000
Yearly Transaction Volumes	1.3 billion (2010)

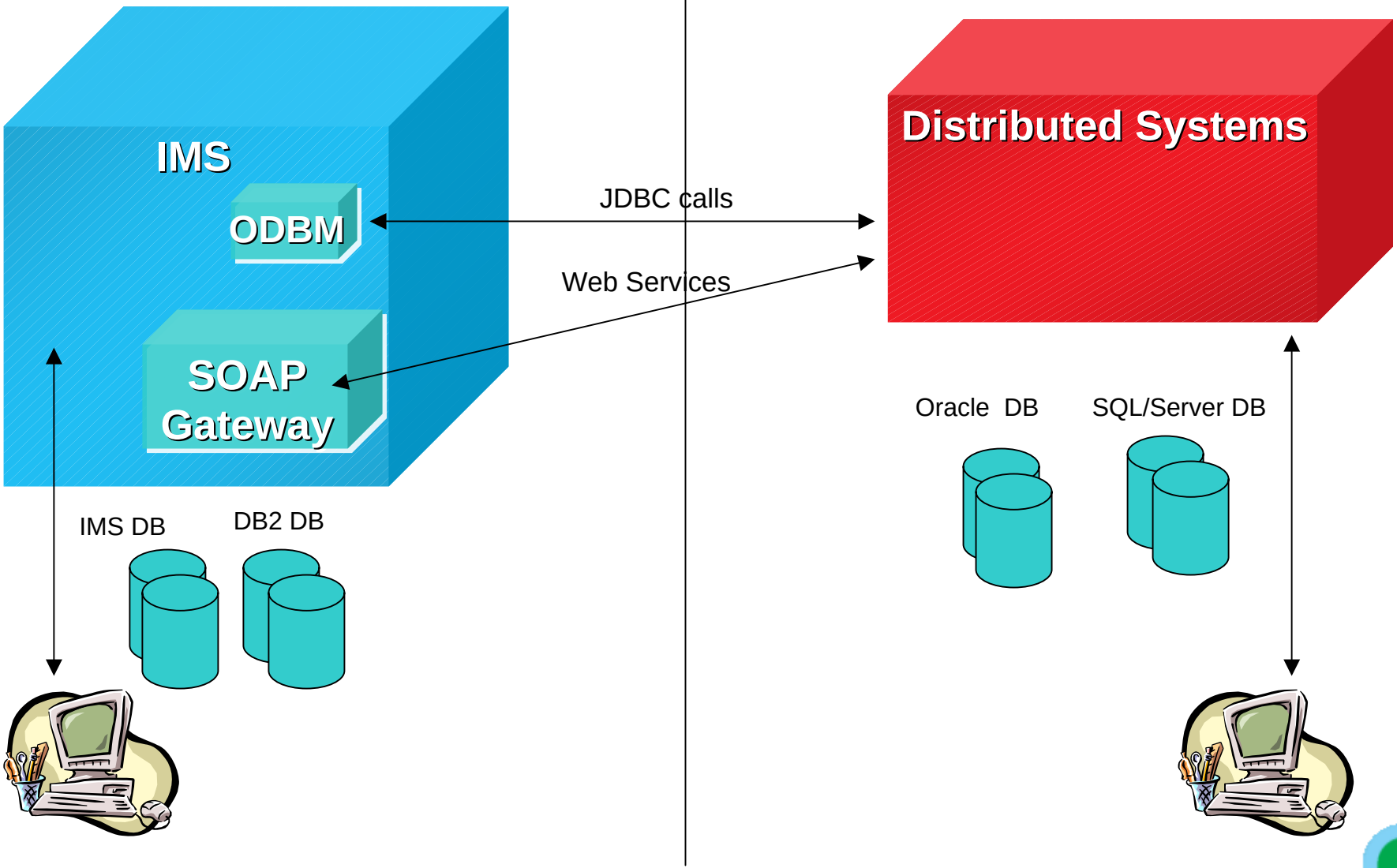
Old Configuration



The Solution(s)

- A Service Oriented Architecture
  - Implement the IMS SOAP Gateway on z/OS with Web Security.
  - Expose IMS Transactions as Web Services
  - Implement both asynchronous and synchronous Callout from IMS programs.
  - Implement IMS V11 With ODBM.
  - Use RDz and the DLIMODEL utility as the tooling to implement the solutions

New Solution



IMS Transaction as a Web Service

- Most IMS Transactions can be exposed as a Web Service with no change to the IMS program.

- The tooling for doing this is provided by Rational Application Developer for z/OS (RDz) and the IMS SOAP Gateway. IMS Versions 10 and 11 provide 2 limited licenses of RDz.

New Solution

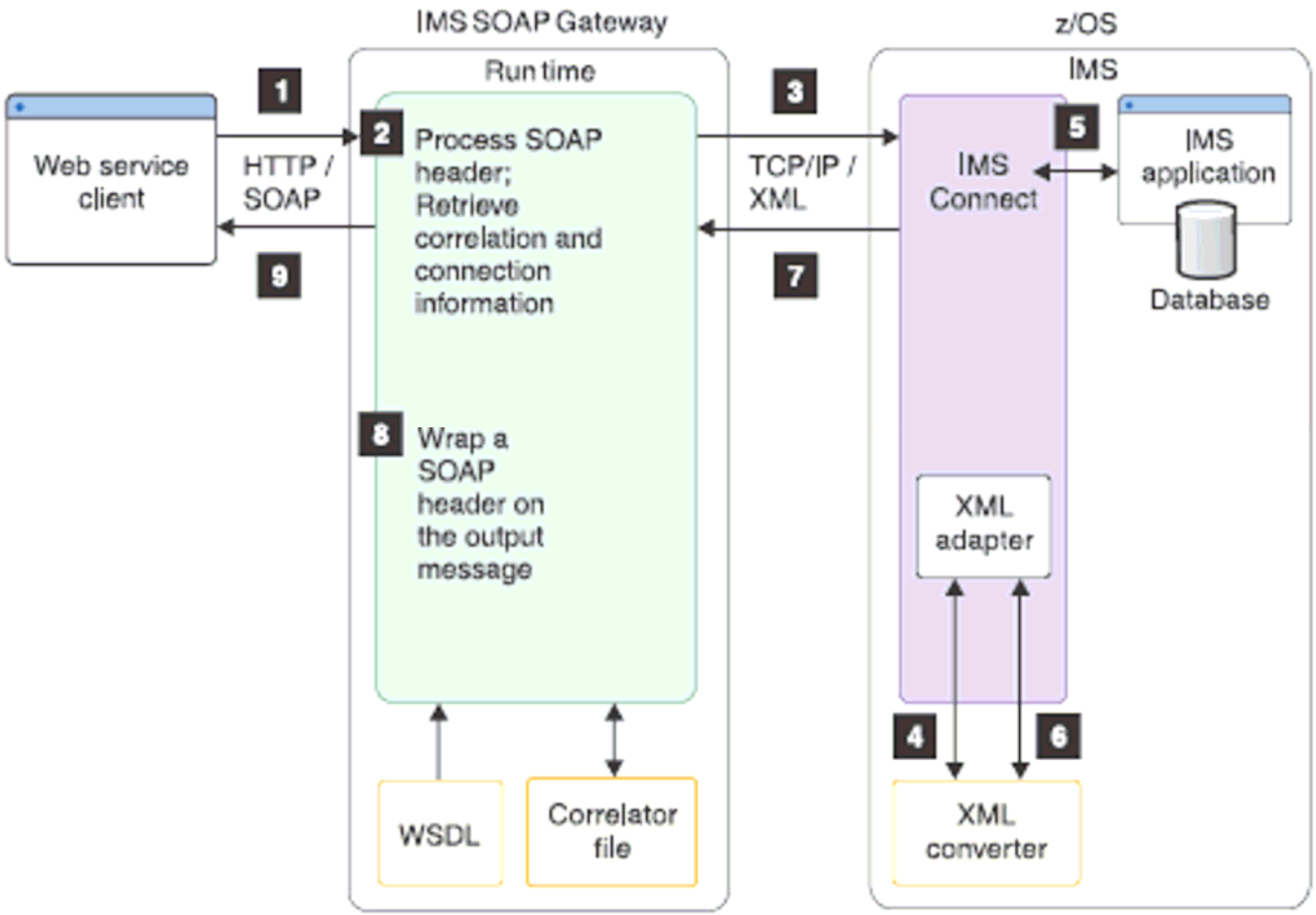


Figure 1. IMS SOAP Gateway runtime environment for the IMS applications as Web service providers scenario

IMS Transaction as a Web Service

IMS Transaction Program

```
01 I-AREA.
  05 I-LENGTH   PIC S999 COMP SYNC.
  05 I-ZZ       PIC S999 COMP SYNC VALUE +0.
  05  TRANCODE  PIC X(8).
  05  A         PIC S9(8).
  05  B         PIC S9(8).

01 O-AREA.
  05 O-LENGTH   PIC S999 COMP SYNC VALUE +12.
  05 O-ZZ       PIC S999 COMP SYNC.
  05 C          PIC S9(8).
```

IMS Transaction as a Web Service

Web Service WSDL File

```
<complexType name="IAREA">
        <sequence>
          <element form="qualified" name="a">
            <simpleType>
              <restriction base="int">
                <minInclusive value="-99999999"/>
                <maxInclusive value="99999999"/>
              </restriction>
            </simpleType>
          </element>
          <element form="qualified" name="b">
            <simpleType>
              <restriction base="int">
                <minInclusive value="-99999999"/>
                <maxInclusive value="99999999"/>
              </restriction>
            </simpleType>
          </element>
        </sequence>
      </complexType>
```

IMS Transaction as a Web Service

Web Service WSDL File

```
<complexType name="OAREA">
        <sequence>
          <element form="qualified" name="c">
            <simpleType>
              <restriction base="int">
                <minInclusive value="-99999999"/>
                <maxInclusive value="99999999"/>
              </restriction>
            </simpleType>
          </element>
        </sequence>
      </complexType>
```

IMS Transaction as a Web Service Consumer

- IMS Supports both Asynchronous and Synchronous "Callout" to Web Services

- The tooling for doing this is also provided by Rational Application Developer for z/OS (RDz) and the IMS Enterprise Suite SOAP Gateway. IMS Versions 10, 11, and 12 provide 2 limited licenses of RDz.

IMS Transaction as a Web Service Consumer (CALLOUT)

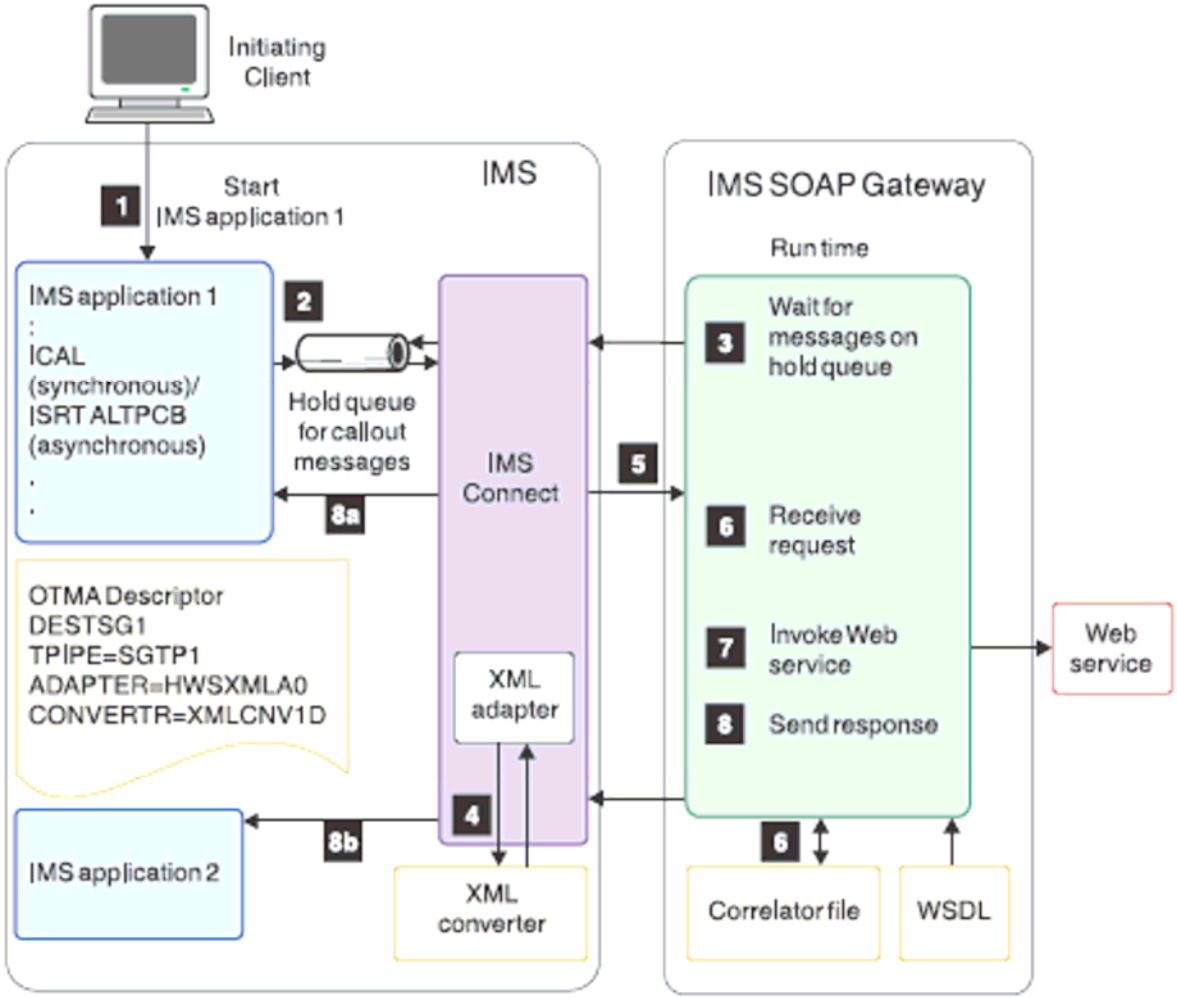


Figure 2. IMS SOAP Gateway runtime environment for the IMS applications as Web service consumers scenario (asynchronous callout)

IMS Transaction as a Web Service Consumer (ASYNC)

COBOL Copylib

```
01  INPUT-AREA.
  05  I-LENGTH  PIC S999 COMP SYNC VALUE +12.
  05  I-ZZ      PIC S999 COMP SYNC.
  05  INPUT1    PIC 9(08).
  05  INPUT2    PIC 9(08).
01  OUTPUT-AREA.
  05  O-LENGTH  PIC S999 COMP SYNC VALUE +12.
  05  O-ZZ      PIC S999 COMP SYNC.
  05  TRANCODE  PIC X(08).
  05  FILLER1   PIC X(01).
  05  OUTPUT2   PIC 9(08).
```

IMS Call

CALL 'CBLTDLI' USING 'ISRT' ALTIO-PCB INPUT-AREA.

OTMA Descriptor

```
D KR01ASYN TYPE=IMSCON TMEMBER=TIMSTOC1 TPIPE=KR01ASYN
D KR01ASYN ADAPTER=HWSXMLA0 CONVERTR=BCPASYND
```

IMS Transaction as a Web Service Consumer (ASYNC)

SOAP Gateway Correlator File

```
Adapter Type                          : IBM XML Adapter
Converter Name                        : BCPASYND
Connection Bundle Name                : timssxc
Socket Timeout                        : 0
Execution Timeout                     : 0
Lterm Name                            :
Transaction Code                      : KR01SPL1
Callout Connection Bundle Name        : timssxcout
Callout WSDL                          : BCPASYNCMIN.wsdl
Callout Web Services Timeout          : 7500
Correlator Service Name               : Service1
Correlator Operation Name             : Compare
```

IMS Transaction as a Web Service Consumer (ASYNC)

Connection Bundle

```
Connection Bundle Name                          : timssxc
Host Name                                       : tcpipsyt.cis.cat.com
Port Number                                     : 10115
Datastore                                       : TIMS
IMS User ID                                     : z1jgims1
IMS User ID Password                            : ********
Group Name                                      :
Callout SSL Truststore Name                     :
Callout SSL Truststore Password                 :
Callout SSL User Authentication Name            :
Callout SSL User Authentication Password        :
Callout SSL Keystore Name                       :
Callout SSL Keystore Password                   :
Callout TPipes                                  : KR01ASYN, NPS39001
```

IMS Transaction as a Web Service Consumer (ASYNC)

Web Service WSDL File

```
<s:element name="Compare">
<s:complexType>
<s:sequence>
  <s:element minOccurs="0" maxOccurs="1" name="o" type="tns:BCPMath" />
  </s:sequence>
  </s:complexType>
  </s:element>
<s:complexType name="BCPMath">
<s:sequence>
  <s:element minOccurs="1" maxOccurs="1" name="a" type="s:int" />
  <s:element minOccurs="1" maxOccurs="1" name="b" type="s:int" />
  </s:sequence>
  </s:complexType>
<s:element name="CompareResponse">
<s:complexType>
<s:sequence>
  <s:element minOccurs="1" maxOccurs="1" name="CompareResult" type="s:int" />
  </s:sequence>
  </s:complexType>
  </s:element>
  </s:schema>
```

IMS Transaction as a Web Service Consumer (SYNC)

COBOL Copylib

```
01  CALLOUT-REQUEST.
  05  INPUT1     PIC 9(08).
  05  INPUT2     PIC 9(08).
01  CALLOUT-RESPONSE.
  05  OUTPUT2    PIC 9(08).
```

IMS Call

MOVE 'SENDRECV' TO AIBSFUNC
MOVE 'KR01MNOX' TO AIBRSNM1
MOVE +16 TO AIBOALEN
MOVE +8 TO AIBOAUSE
MOVE +10000 TO AIBRSFLD
CALL 'AIBTDLI' USING '**ICAL**' AIB CALLOUT-REQUEST CALLOUT-RESPONSE.

OTMA Descriptor

```
D KR01MNOX TYPE=IMSCON TMEMBER=TIMSTOC1 TPIPE=KR01MNOX
D KR01MNOX ADAPTER=HWSXMLA0 CONVERTR=KR01MNOD
```

Where are we today?.

Installations:

We have 20 Transactions as Web Services, 3 ASYNC Callouts, and 1 JDBC(ODBM) client in production. Several additional applications are in development.

Maintenance:

Currently, in both development and production, we are running IMS V11 at RSU 1104+, and the IMS Enterprise Suite SOAP Gateway on z/OS, with RDz 7.6.1.

For the Enterprise Suite 1.1 SOAP Gateway, we are moving to Fixpak 3. That is PM38023(UK68596) and PM38024(UK68597). It also requires moving to RDz 8.0.1.

For ODBM, we have applied PM11977(UK67096), and PM37073(UK70572).

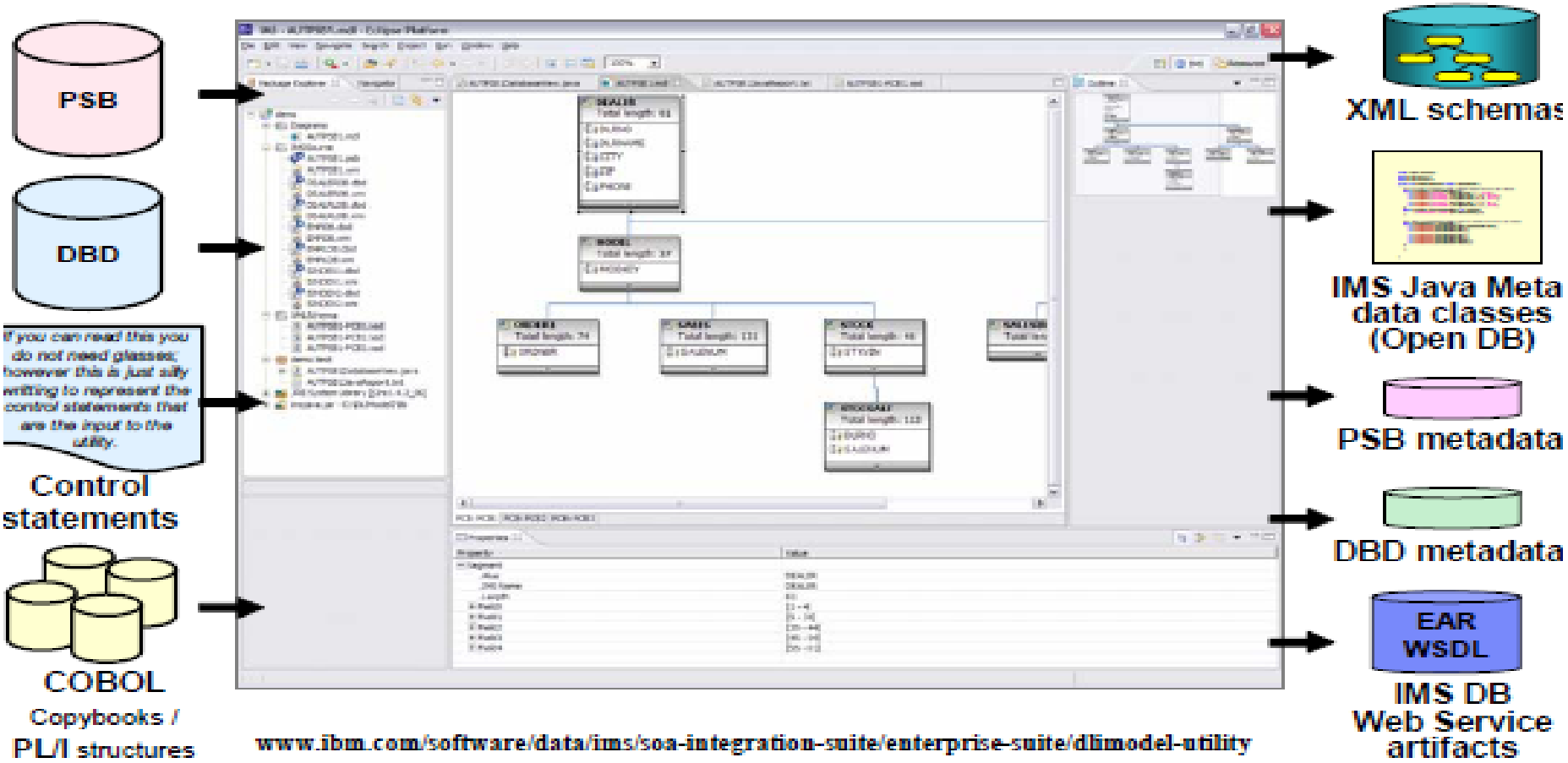
IMS V11 with Open DataBase Manager (ODBM)

- ODBM provides a way for distributed java applications to make database calls directly to IMS, using the IMS Universal Drivers, in much the the same way that they have accessed DB2 tables in the past.

- The tooling for doing this is provided by the IMS DLIMODEL Utility, running under Eclipse, and optionally the Rational Application Developer for z/OS (RDz). IMS Versions 10, 11, and 12 provide 2 limited licenses of RDz.

IMS V11 – ODBM

IMS Enterprise Suite DLIModel Utility Plug-In



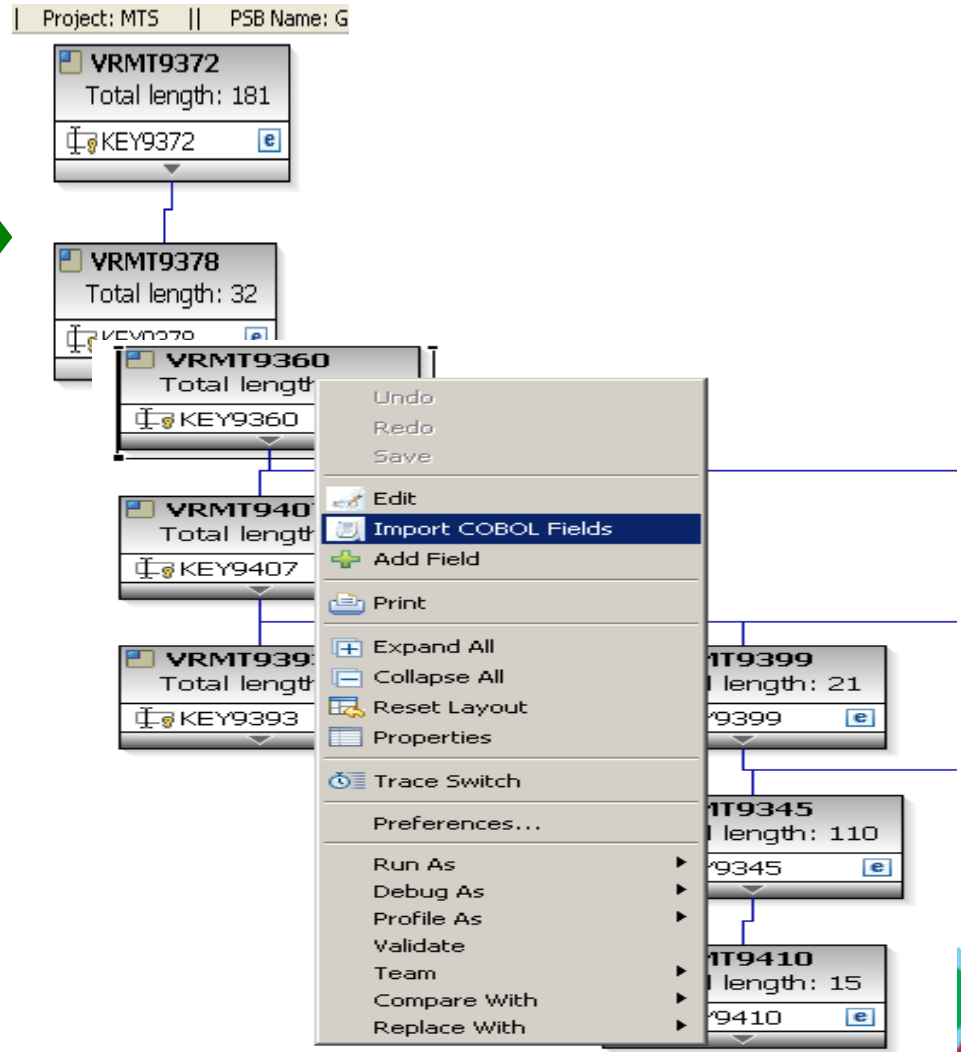
www.ibm.com/software/data/ims/soa-integration-suite/enterprise-suite/dlimodel-utility

IMS V11 – DL/I Model Utility PSBs – DBDs

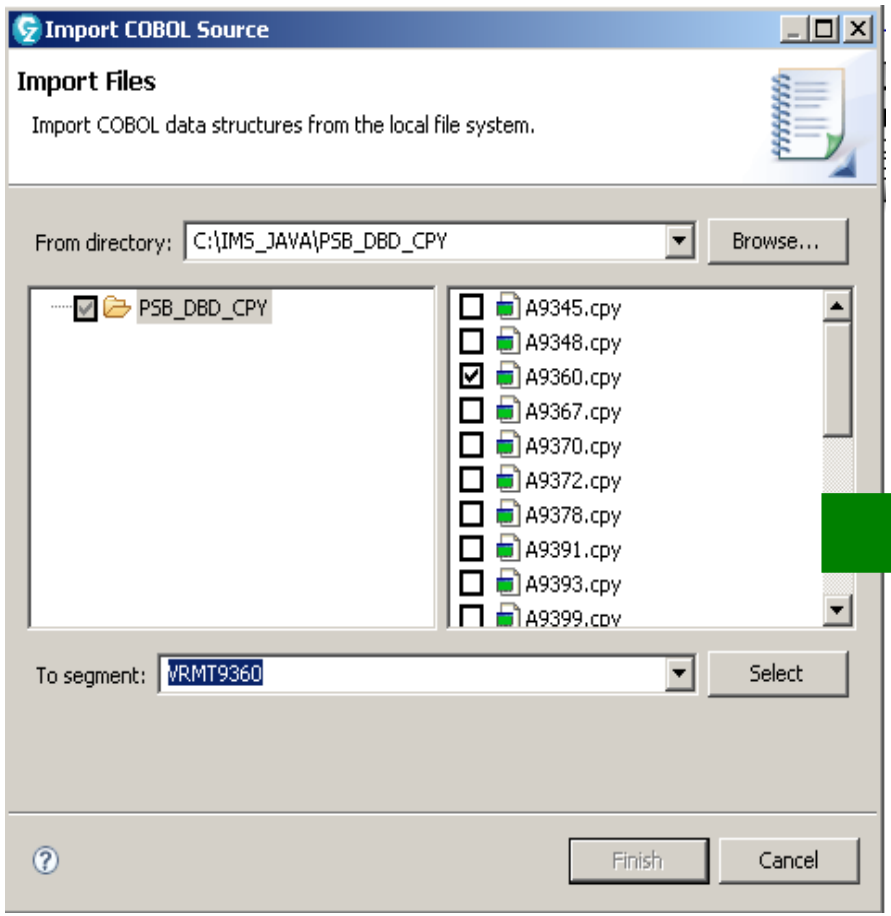
```
 PRINT   NOGEN
 PCB9NA0 PCB    TYPE=DB,DBDNAME=VRC19NA0,PROCOPT=GOT,KEYLEN=200
     SENSEG  NAME=VRMT9360,PARENT=0
     SENSEG  NAME=VRMT9407,PARENT=VRMT9360
     SENSEG  NAME=VRMT9393,PARENT=VRMT9407
     SENSEG  NAME=VRMT9399,PARENT=VRMT9407
     SENSEG  NAME=VRMT9345,PARENT=VRMT9399
```

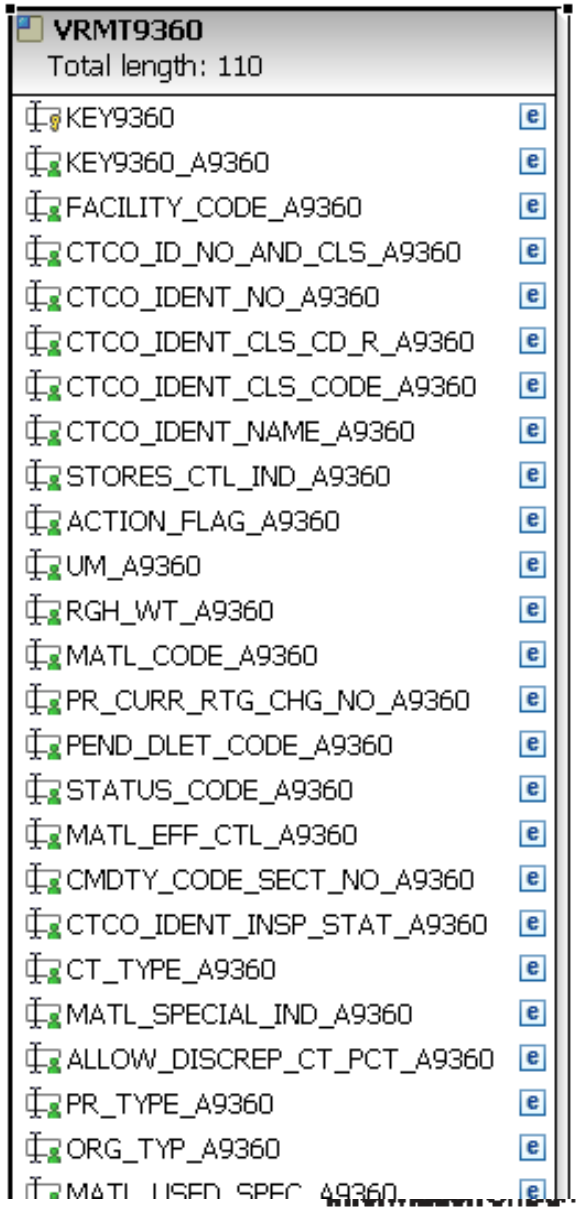
```
 PRINT   NOGEN
     DBD     NAME=VRLW9NA0,                          C
           ACCESS=(HDAM,OSAM),
           RMNAME=(DFSHDC40,10,17850,1024)
*                    *----------> 150 CYLS.
*
DSG1   DATASET
DD1=VRW19NA0,DEVICE=3380,SIZE=6144,SCAN=0,FRSPC=(0,10)
     SEGM    NAME=VRMT9360,BYTES=110,                C
           COMPRTN=COMPRSS1
      FIELD  NAME=(KEY9360,SEQ,U),BYTES=13,START=1
     SEGM    NAME=VRMT9407,PARENT=VRMT9360,BYTES=13
      FIELD  NAME=(KEY9407,SEQ,U),BYTES=6,START=1
     SEGM    NAME=VRMT9393,PARENT=VRMT9407,BYTES=73,         C
           COMPRTN=COMPRSS1
      FIELD  NAME=(KEY9393,SEQ,M),BYTES=1,START=1
      FIELD  NAME=(KEY9429,SEQ,U),BYTES=15,START=1
     DBDGEN
     FINISH



     END
```

DL/I Model utility



IMS V11 – DL/I Model Utility Copybooks





IMS V11 – ODBM – Data Mapping

Copybook format	DLIType info data type	Java data types
PIC X(25)	CHAR	java.lang.String
PIC S9 (1-4 figures) COMP	SMALLINT (2 Bytes)	short
PIC S9 (5-9 figures) COMP-4	INTEGER (4 Bytes)	int
PIC S9 (10-18 figures) BINARY	BIGINT (8 Bytes)	long
COMP-1	FLOAT	float
COMP-2	DOUBLE	double
PIC S9(06)V99 COMP-3	PACKEDDECIMAL	java.math.BigDecimal
PIC S9(06)V99	ZONEDECIMAL	java.math.BigDecimal
PIC 9(06).99	ZONEDECIMAL	java.math.BigDecimal
PIC 9 DISPLAY	ZONEDECIMAL	java.math.BigDecimal

IMS V11 – ODBM – Generated Metadata Class

```
package Z1MXODBM;
. . .
public class Z1MXODBMDatabaseView extends DLIDatabaseView {
// The following describes Segment: MX1701 ("MX1701") in PCB: MX1701 ("MX1701")
    static DLITypeInfo[] MX1701MX1701Array= {
        new DLITypeInfo("PARMKEY", DLITypeInfo.CHAR, 1, 37, "PARMKEY", DLITypeInfo.UNIQUE_KEY),
        new DLITypeInfo("PARAMETER_KEY_MX1701", DLITypeInfo.CHAR, 1, 37),
        new DLITypeInfo("RCD_TYPE_MX1701", DLITypeInfo.CHAR, 1, 3),
        new DLITypeInfo("SEGMENT_KEY_MX1701", DLITypeInfo.CHAR, 4, 34),
        new DLITypeInfo("VARI_DATA_MX1701", DLITypeInfo.CHAR, 38, 339)
    };
    static DLISegment MX1701MX1701Segment= new DLISegment
        ("MX1701","MX1701",MX1701MX1701Array,376);
    // An array of DLISegmentInfo objects follows to describe the view for PCB: MX1701 ("MX1701")
    static DLISegmentInfo[] MX1701array = {
        new DLISegmentInfo(MX1701MX1701Segment,DLIDatabaseView.ROOT)
    };
// Constructor
    public Z1MXODBMDatabaseView() {
        super("2.0.3","Z1MXODBM", "MX1701", "MX1701", MX1701array, "G");
    } // end Z1MXODBMDatabaseView constructor
} // end Z1MXODBMDatabaseView class definition
```

IMS V11 – ODBM – Required files for Client

- The jar file produced from the DLIModel Utility for the PSB containing the Database Views.

- The IMS Universal drivers. It is part of IMS V11(FMID JMK1106). The JDBC and DL/I drivers are shipped in a single jar file, **imsudb.jar**.
  - This jar file is part of IMS and is an OMVS file usually mounted as /usr/lpp/ims/ims11/imsjava/ . Be sure to FTP the IMSUDB.jar file as binary.
- Both of these files must be added to the build path so that they are available to the Java Client program.

IMS V11 – ODBM JDBC Call

Segment field = Table field

PCB = DB Schema

Segment = Table

```
public static void displayDealer() throws SQLException( //15
    String sql="SELECT * FROM PCB9NAO.VRMT9378 " +
       "WHERE CODE7431 = 'W3146LO'";
    System.out.println("\nSQL Command: "+sql+"\n");
    ResultSet rs = st.executeQuery(sql); //16
    ResultSetMetaData rsmd = rs.getMetaData(); //17
    int numColumns = rsmd.getColumnCount();
    for (int i=1; i<=numColumns; i++) {
    System.out.print(rsmd.getTableName(i)+"."+rsmd.getColumnName(i)+" | ");
    }
    System.out.println("\n-----------------------------------"+
    "-----------------------------------------");
    while (rs.next()) { //18
    for (int i=1; i<=numColumns; i++) {
    System.out.print(rs.getString(i) + " | "); //19
    }
    System.out.println();
    }
    rs.close(); //20
}
```

Caterpillar and IMS: Application Modernization using the IMS SOAP Gateway

IBM Redbook – "IMS Open Database" – SG24-7856-00.

Questions???

Steve Clanton - Caterpillar Inc.

Email: Clanton_Steven_E@cat.com

Thank You!
Your Feedback is Important to Us

- Access your personal session survey list and complete via SmartSite
  - Your smart phone or web browser at: iodsmartsite.com
  - Any SmartSite kiosk onsite
  - Each completed session survey increases your chance to win an Apple iPod Touch with daily drawing sponsored by Alliance Tech