E42

Modernize MFS Transactions with XML and Web Services

Shyh-Mei F. Ho

Senior Technical Staff Member

IMS e-business Architecture, SVL, IBM

shyhmei@us.ibm.com

IMS

technical conference

Las Vegas, NV **September 15 – September 18, 2003**

© IBM Corporation 2003

RETURN TO INDEX

Agenda

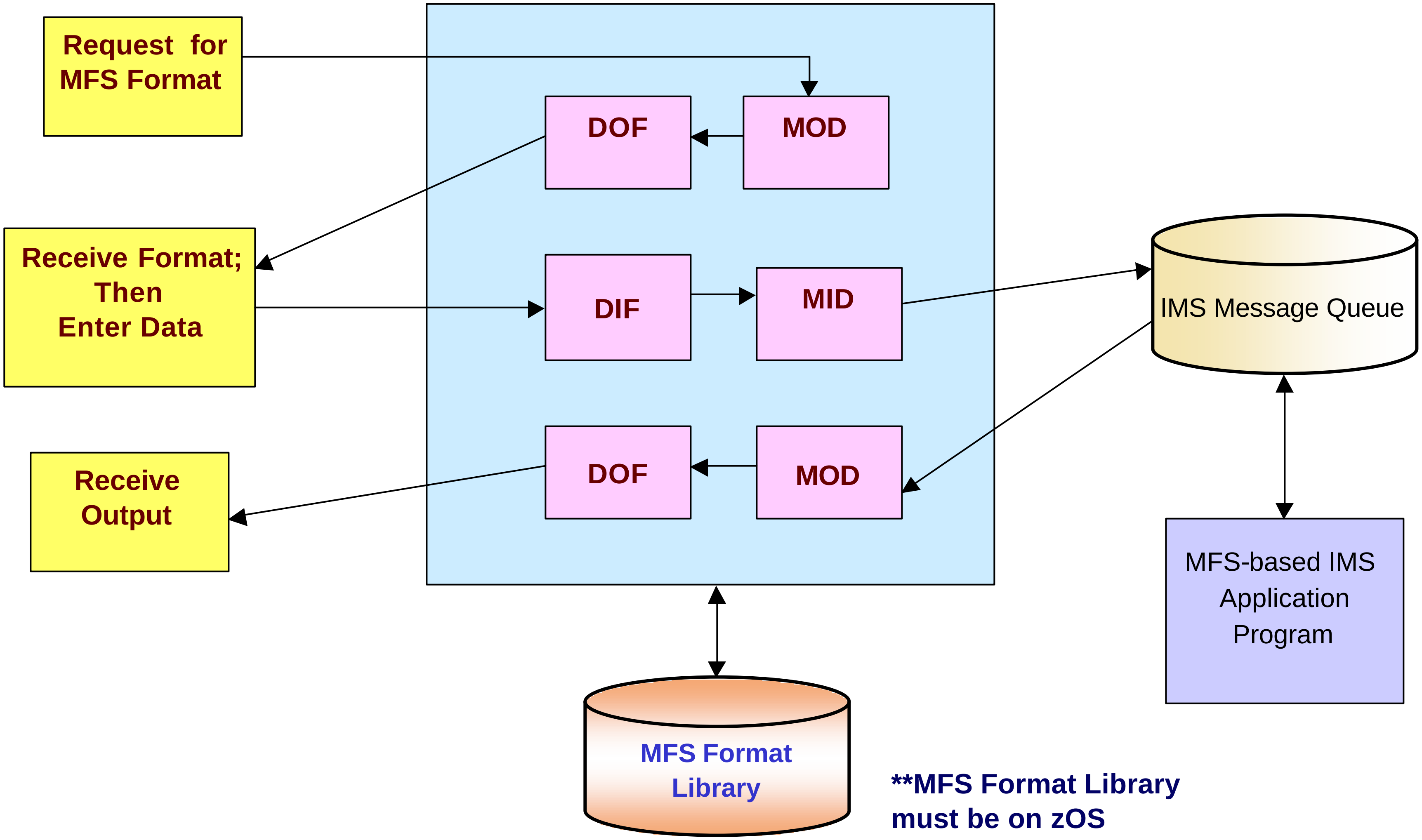
- What is MFS and traditional MFS online processing
- Business Challenge
- Modernize MFS Transactions
- MFS Web Services
- MFS Future Requirements

MFS (Message Format Services)

- MFS formats messages to and from terminal devices
- MFS language utility compiles MFS source and generates MFS control blocks, known as MID/MOD and DIF/DOF, and places them in an IMS Format Library
- IMS application program I/O data structures do not fully describe the end user interaction with existing MFS-based IMS application. Information buried within various MFS statements are processed by the MFS online processor
  - 3270 screen attribute bytes
  - PF Key input data
  - And etc.

© IBM Corporation 2003

Tradition MFS online processing



****MFS Format Library must be on zOS**

Business Challenge

- As business processes are updated to exploit new B2B technologies, there is a need to support B2B interchange
  - A nonproprietary industry-wide standard method is needed to represent existing MFS control blocks
  - XML is growing in acceptance as the universal data format which can be the input and output for any application
  - Represent traditional MFS control blocks in XML
- Retarget MFS-based IMS transactions to support B2B XML communication and Web services without changing existing IMS applications

© IBM Corporation 2003

Modernize MFS Transactions

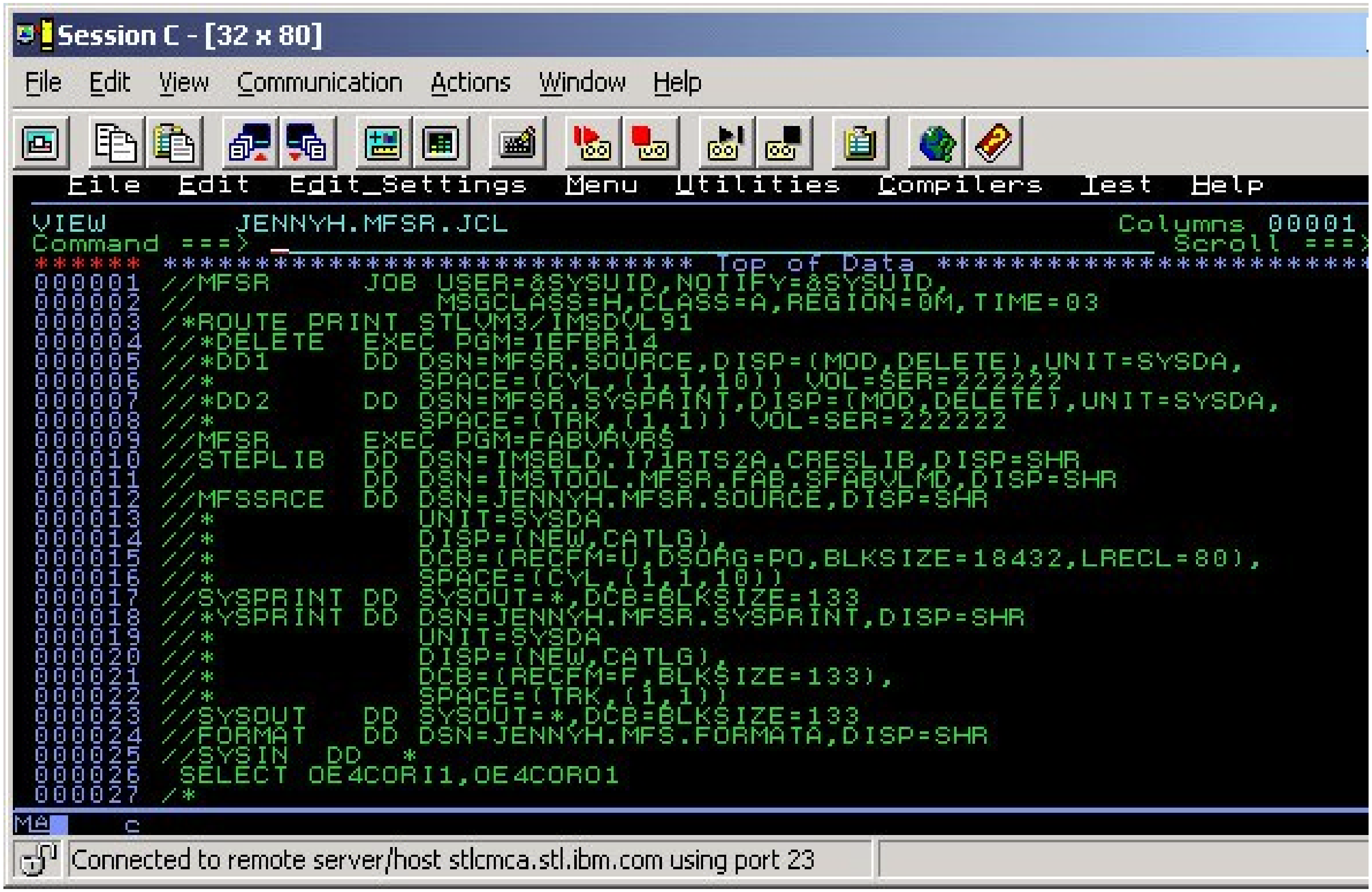
- MFS metamodel
  - Preserve existing IMS applications by modeling MFS source
- MFS importer
  - Parse and convert MFS statements into XMI (XML metadata interchange)
- MFS transformer
  - Translate XMI back into an equivalent MFS online processing; and vice verse

MFS Metamodel

- Preserve existing IMS applications by modeling MFS sources into metamodel
  - A nonproprietary industry-wide standard way to represent MFS source information
    - One of the CAM (Common Application Metamodel) models
    - OMG EAI (Enterprise Application Integration) marketplace standards
      - Non-normative metamodel
  - Use IMS MFS Reversal Utility to re-create MFS source from MFS Format Library, if MFS sources are not available
  - Support certain services/functions provided by MFS
    - PF keys
    - Logical pages
    - Predefined literals
    - Attribute bytes
  - Do not describe 3270 data streams
    - Capability of 3270 data streams is supported by HOD (Host On Demand) and WebSphere HATS (Host Access Transformation Server)

© IBM Corporation 2003

IMS MFS Reversal Utility – sample JCL

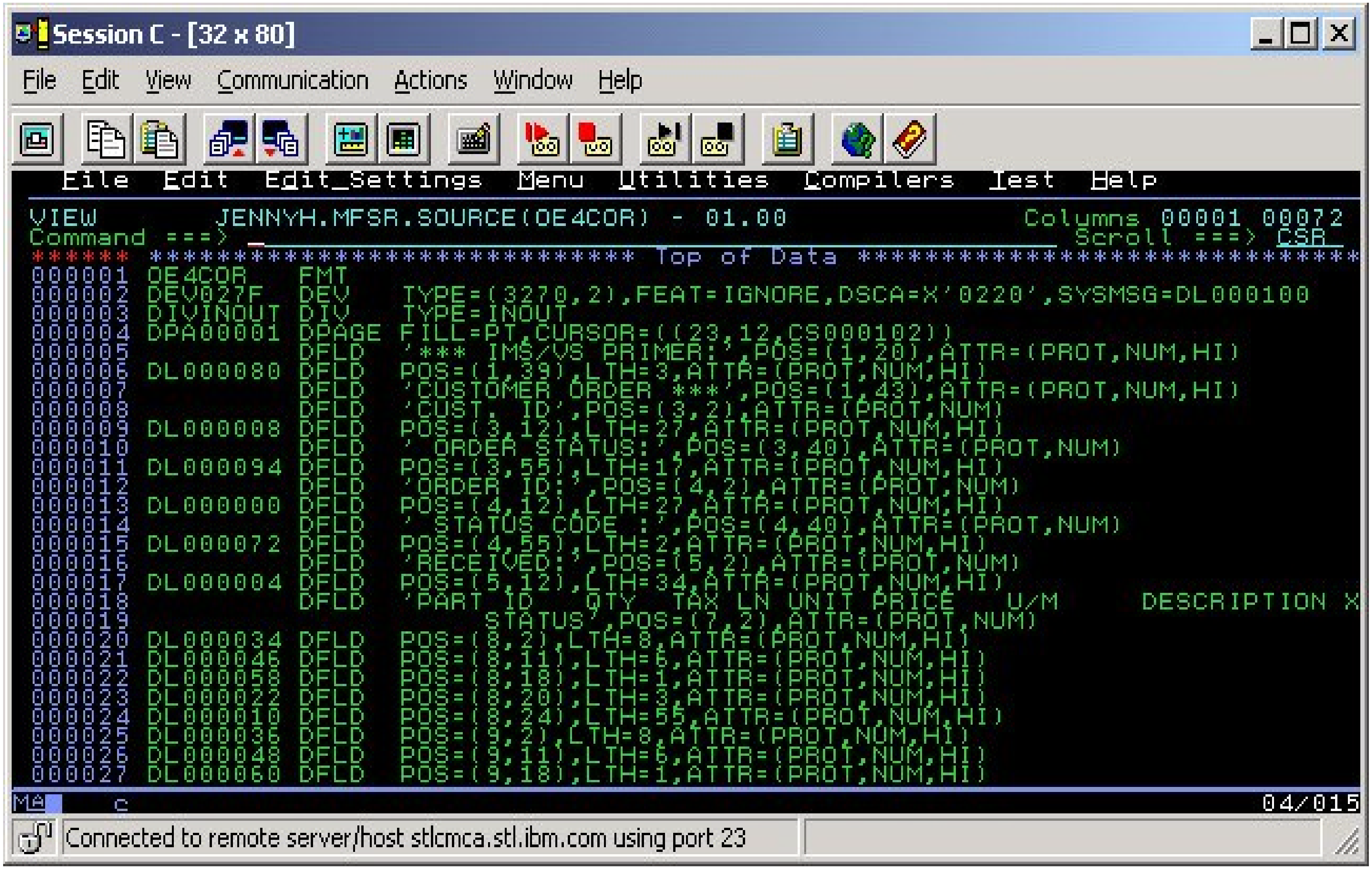

Session C - [32 x 80]
File Edit View Communication Actions Window Help
File Edit Edit_Settings Menu Utilities Compilers Test Help
MA c

```
VIEW       JENNYH.MFSR.JCL                                  Columns 00001
Command ===> _                                                 Scroll ===>
****** *************************** Top of Data **************************
//MFSR     JOB USER=&SYSUID,NOTIFY=&SYSUID,
//             MSGCLASS=H,CLASS=A,REGION=0M,TIME=03
/*ROUTE PRINT STLVM3/IMSDVL91
//*DELETE  EXEC PGM=IEFBR14
//*DD1     DD DSN=MFSR.SOURCE,DISP=(MOD,DELETE),UNIT=SYSDA,
//*           SPACE=(CYL,(1,1,10)) VOL=SER=222222
//*DD2     DD DSN=MFSR.SYSPRINT,DISP=(MOD,DELETE),UNIT=SYSDA,
//*           SPACE=(TRK,(1,1)) VOL=SER=222222
//MFSR     EXEC PGM=FABVAVRS
//STEPLIB  DD DSN=IMSBLD.I71RTS2A.CRESLIB,DISP=SHR
//         DD DSN=IMSTOOL.MFSR.FAB.SFABVLMD,DISP=SHR
//MFSSRCE  DD DSN=JENNYH.MFSR.SOURCE,DISP=SHR
//*           UNIT=SYSDA
//*           DISP=(NEW,CATLG),
//*           DCB=(RECFM=U,DSORG=PO,BLKSIZE=18432,LRECL=80),
//*           SPACE=(CYL,(1,1,10))
//SYSPRINT DD SYSOUT=*,DCB=BLKSIZE=133
//*YSPRINT DD DSN=JENNYH.MFSR.SYSPRINT,DISP=SHR
//*           UNIT=SYSDA
//*           DISP=(NEW,CATLG),
//*           DCB=(RECFM=F,BLKSIZE=133),
//*           SPACE=(TRK,(1,1))
//SYSOUT   DD SYSOUT=*,DCB=BLKSIZE=133
//FORMAT   DD DSN=JENNYH.MFS.FORMATA,DISP=SHR
//SYSIN  DD   *
 SELECT OE4CORI1,OE4CORO1
/*
```

Connected to remote server/host stlcmca.stl.ibm.com using port 23

© IBM Corporation 2003

IMS MFS Reversal Utility – generated source file

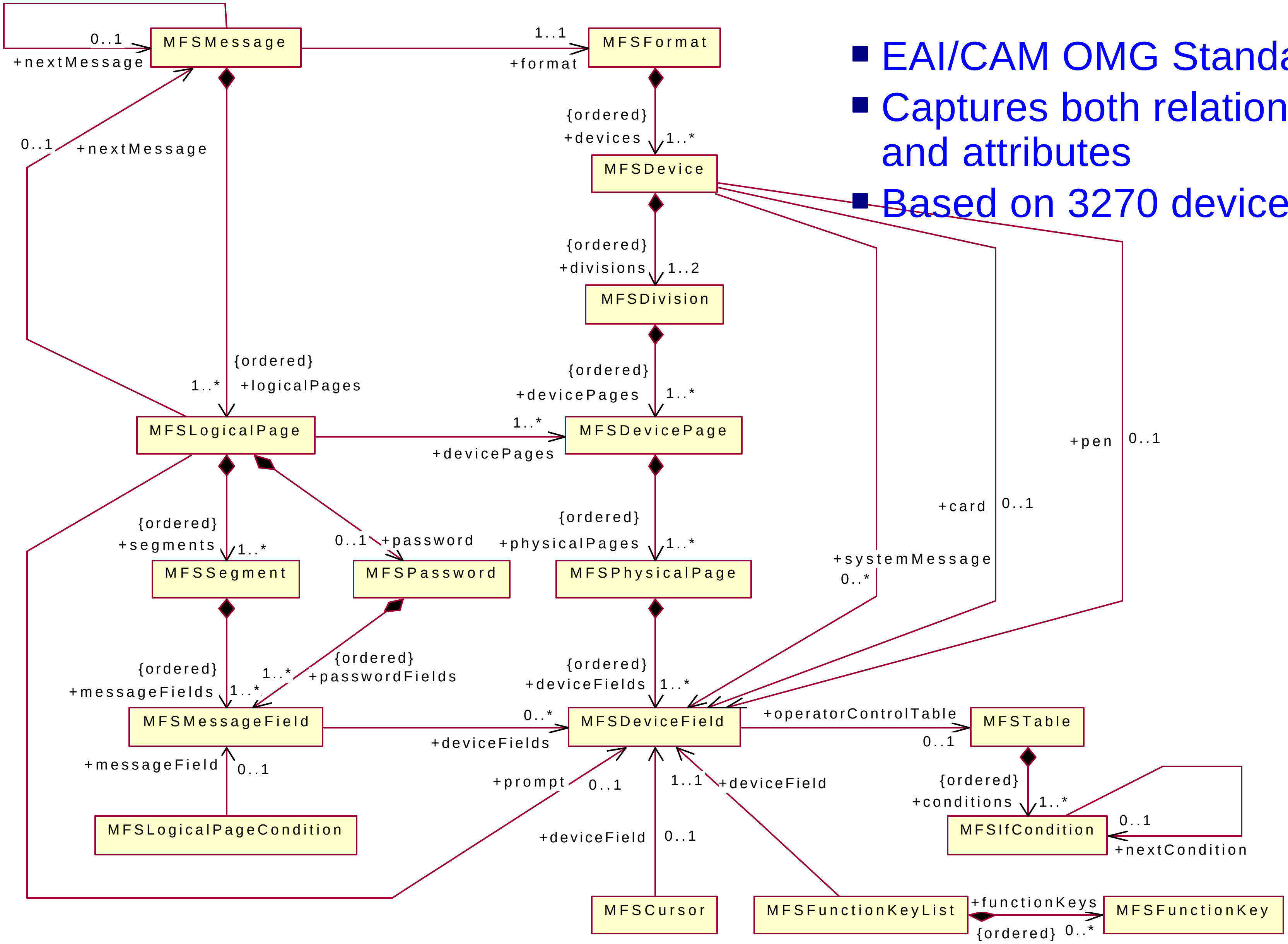

Session C - [32 x 80]
File Edit View Communication Actions Window Help
File Edit Edit_Settings Menu Utilities Compilers Test Help

```
VIEW       JENNYH.MFSR.SOURCE(OE4COR) - 01.00              Columns 00001 00072
Command ===> _                                                 Scroll ===> CSR
****** ***************************** Top of Data ******************************
       OE4COR   FMT
       DEV027F  DEV   TYPE=(3270,2),FEAT=IGNORE,DSCA=X'0220',SYSMSG=DL000100
       DIVINOUT DIV   TYPE=INOUT
       DPA00001 DPAGE FILL=PT,CURSOR=((23,12,CS000102))
                DFLD  '*** IMS/VS PRIMER:',POS=(1,20),ATTR=(PROT,NUM,HI)
       DL000080 DFLD  POS=(1,39),LTH=3,ATTR=(PROT,NUM,HI)
                DFLD  'CUSTOMER ORDER ***',POS=(1,43),ATTR=(PROT,NUM,HI)
                DFLD  'CUST. ID',POS=(3,2),ATTR=(PROT,NUM)
       DL000008 DFLD  POS=(3,12),LTH=27,ATTR=(PROT,NUM,HI)
                DFLD  ' ORDER STATUS:',POS=(3,40),ATTR=(PROT,NUM)
       DL000094 DFLD  POS=(3,55),LTH=17,ATTR=(PROT,NUM,HI)
                DFLD  'ORDER ID:',POS=(4,2),ATTR=(PROT,NUM)
       DL000000 DFLD  POS=(4,12),LTH=27,ATTR=(PROT,NUM,HI)
                DFLD  ' STATUS CODE :',POS=(4,40),ATTR=(PROT,NUM)
       DL000072 DFLD  POS=(4,55),LTH=2,ATTR=(PROT,NUM,HI)
                DFLD  'RECEIVED:',POS=(5,2),ATTR=(PROT,NUM)
       DL000004 DFLD  POS=(5,12),LTH=34,ATTR=(PROT,NUM,HI)
                DFLD  'PART ID    QTY  TAX LN UNIT PRICE    U/M      DESCRIPTION X
                           STATUS',POS=(7,2),ATTR=(PROT,NUM)
       DL000034 DFLD  POS=(8,2),LTH=8,ATTR=(PROT,NUM,HI)
       DL000046 DFLD  POS=(8,11),LTH=6,ATTR=(PROT,NUM,HI)
       DL000058 DFLD  POS=(8,18),LTH=1,ATTR=(PROT,NUM,HI)
       DL000022 DFLD  POS=(8,20),LTH=3,ATTR=(PROT,NUM,HI)
       DL000010 DFLD  POS=(8,24),LTH=55,ATTR=(PROT,NUM,HI)
       DL000036 DFLD  POS=(9,2),LTH=8,ATTR=(PROT,NUM,HI)
       DL000048 DFLD  POS=(9,11),LTH=6,ATTR=(PROT,NUM,HI)
       DL000060 DFLD  POS=(9,18),LTH=1,ATTR=(PROT,NUM,HI)
MA      c                                                                 04/015
```

Connected to remote server/host stlcmca.stl.ibm.com using port 23

© IBM Corporation 2003

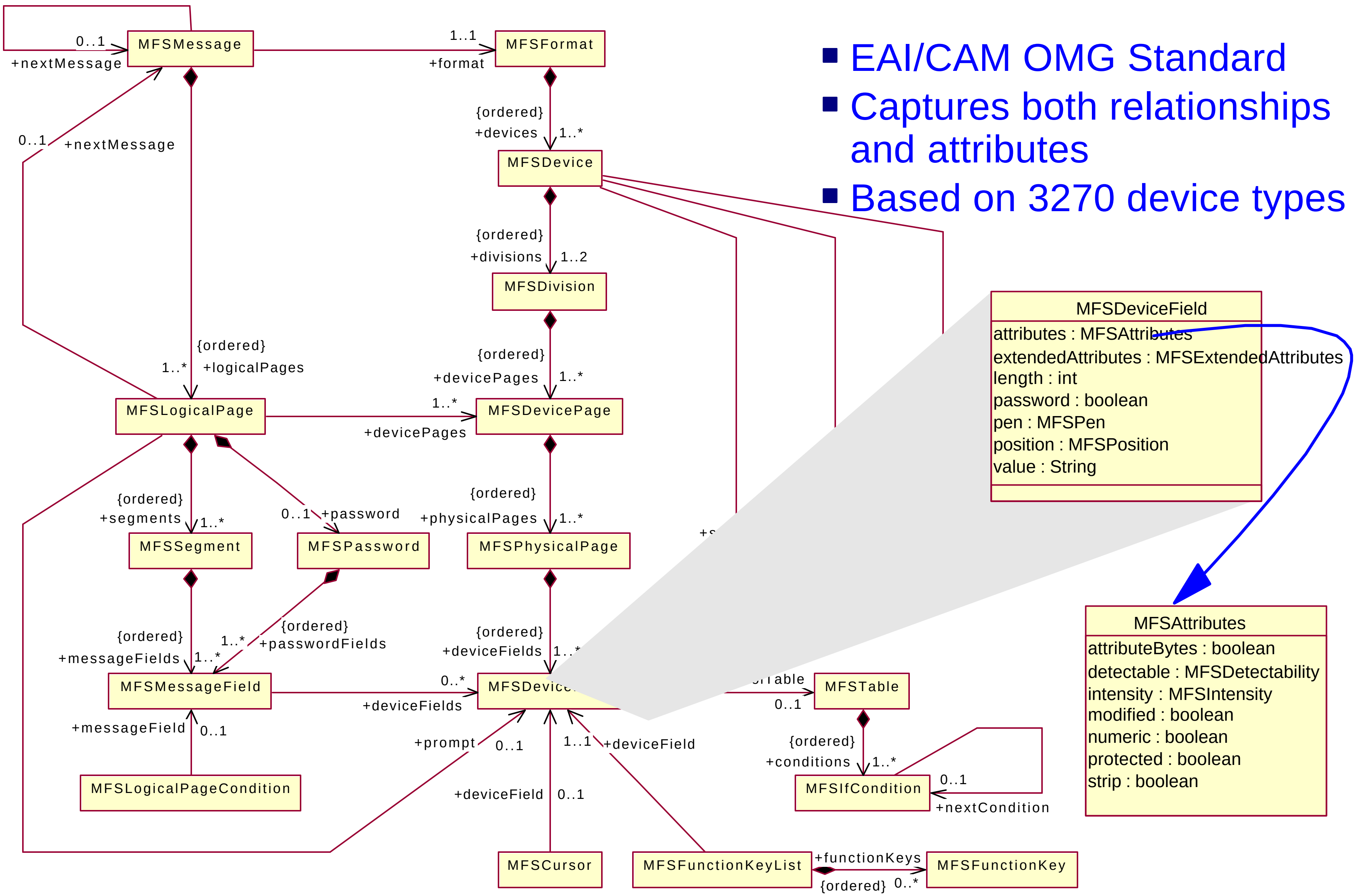
MFS Metamodel



- EAI/CAM OMG Standard
- Captures both relationships and attributes
- Based on 3270 device types

MFS Metamodel

- EAI/CAM OMG Standard
- Captures both relationships and attributes
- Based on 3270 device types



MFSDeviceField

attributes : MFSAttributes
extendedAttributes : MFSExtendedAttributes
length : int
password : boolean
pen : MFSPen
position : MFSPosition
value : String

MFSAttributes

attributeBytes : boolean
detectable : MFSDetectability
intensity : MFSIntensity
modified : boolean
numeric : boolean
protected : boolean
strip : boolean

MFS Importer

- An MFS Importer is built using the MFS metamodel
  - One of the many importers created by IBM to parse and generate XMI instance files based on the CAM metamodels
    - MFS importer reads and parses MFS source files for a particular IMS application to generate XMI files
  - XMI files represent all the application interface information encapsulated by the MFS source including the input and output messages, display information, MFS flow control, device characteristics and operation semantics
  - Two kinds of XMI files are generated
    - Group MID and DIF together; named after the MID name
    - Group MOD and DOF together; named after the MOD name
    - Or, a MFS table
  - XMI metadata can be stored in an XML repository

© IBM Corporation 2003

MFS XMI (XML Metadata Interchange)

```
<?xml version="1.0" encoding="UTF-8"?>
<xmi:XMI xmi:version="2.0" xmlns:xmi="http://www.omg.org/XMI" xmlns:MFS="MFS.xmi">
  <MFS:MFSMessage xmi:id="MFSMessage_1" label="IVTNOMI1" type="input" format="MFSFormat_1">
    <nextMessage href="IVTNO.xmi#MFSMessage_1"/>
    <logicalPages xmi:id="MFSLogicalPage_1" devicePages="MFSDevicePage_1">
      <segments xmi:id="MFSSegment_1">
        <messageFields xmi:id="MFSMessageField_1" length="10" literal="IVTNO     "/>
        <messageFields xmi:id="MFSMessageField_2" length="8" deviceFields="MFSDeviceField_1"/>
      </segments>
    </logicalPages>
  </MFS:MFSMessage>
  <MFS:MFSFormat xmi:id="MFSFormat_1" label="IVTNOF">
    <devices xmi:id="MFSDevice_1"  type="3270-A02">
      <divisions xmi:id="MFSDivision_1" type="inout">
        <devicePages xmi:id="MFSDevicePage_1">
          <physicalPages xmi:id="MFSPhysicalPage_1">
            <deviceFields xmi:id="MFSDeviceField_1" label="CMD" length="8">
              <attributes xmi:id="MFSAttributes_8" intensity="high" modified="true"/>
              <position xmi:id="MFSPosition_9" row="10" column="34"/>
            </deviceFields>
            <deviceFields xmi:id="MFSDeviceField_2" length="21" value="PROCESS  CODE  (*1) :">
              <attributes xmi:id="MFSAttributes_7" protected="true"/>
              <position xmi:id="MFSPosition_8" row="10" column="10"/>
            </deviceFields>
          </physicalPages>
        </devicePages>
      </divisions>
    </devices>
  </MFS:MFSFormat>
</xmi:XMI>
```

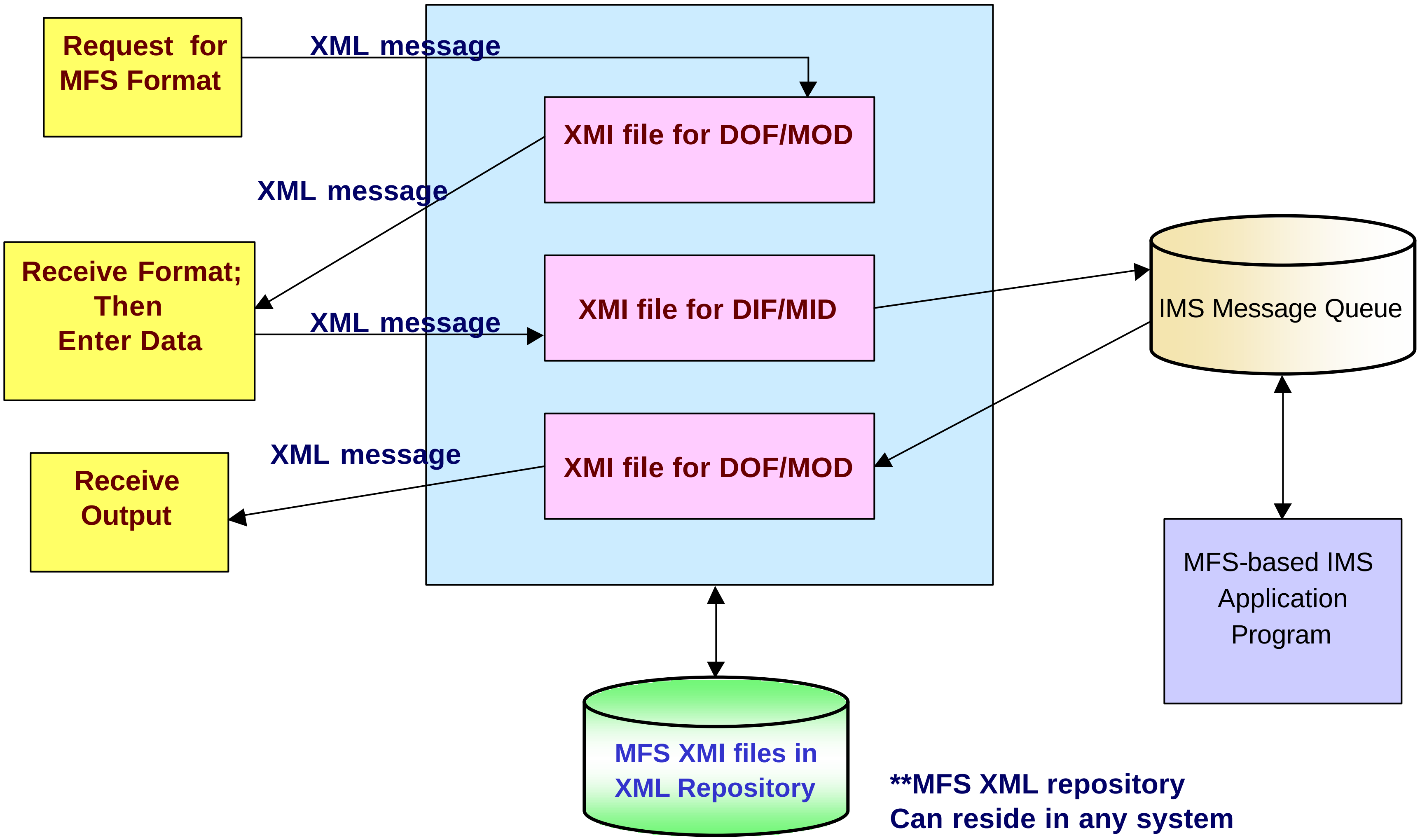
- XMI *is* XML
- XMI describes relationships

© IBM Corporation 2003

MFS Transformer

- An MFS Transformer is needed to translate XMI back into an equivalent MFS online processing; and vice versa
  - Replace MFS online processor
  - Can reside in any system in a network
  - No long need VTAM and 3270 emulator
- Retarget MFS-based IMS transactions to support B2B XML communication or Web services without changing existing IMS applications
  - Enable MFS based IMS transactions as Web Services
  - Facilitate operation of MFS based IMS transactions on any displayable devices, including browser

MFS XML Processing



****MFS XML repository Can reside in any system**

Agenda

- **What is MFS and traditional MFS online processing**
- **Business Challenge**
- **Modernize MFS Transactions**
- **MFS Web Services**
- **MFS Future Requirements**

IMS MFS and Web Services

- **Enable users to create a service definition from MFS sources, publish the service, and then deploy it to WAS, and make it available as an EJB or SOAP service**

- **Enable IMS customers to publish existing MFS-based IMS applications on the Internet as Web services**

- **MFS Importer and Web services wizards**
  - Iintegrated with WSAD-IE 5.0 Enterprise Services Toolkit
  - Parse MFS source files to generate **EAR** (Enterprise Application Resource)
    - XMI files
    - 3 service definition files (i.e. WSDLs)
    - input and output beans
    - Java, EJB and SOAP proxies

- **MFS Adapter & Generator**
  - generate format handlers to be included as part of **EAR** for data marshalling

© IBM Corporation 2003

MFS as a Web Services

```
<complexType name="IVTNO">
  <sequence>
    <element maxOccurs="unbounded" minOccurs="1" name="IVTNO_Pages">
      <complexType name="IVTNO_Pages">
        <choice>
          <element name="IVTNO_Page1">
            <complexType name="IVTNO_Page1">
              <sequence>
                <element name="SDATE">
                  <simpleType>
                    <restriction base="string">
                      <length value="8" />
                    </restriction>
                  </simpleType>
                </element>
                <element name="CMD">
                  <simpleType>
                    <restriction base="string">
                      <length value="8" />
                    </restriction>
                  </simpleType>
                </element>
              </sequence>
            </complexType>
          </element>
        </choice>
      </complexType>
    </element>
  </sequence>
</complexType>
```

```
<MFS:MFSMessage xmi:id="MFSMessage_1" label="IVTNO"
 format="MFSFormat_1">
<MFS:MFSFormat xmi:id="MFSFormat_1" label="IVTNOF">
```

```
<devicePages xmi:id="MFSDevicePage_1">
<logicalPages xmi:id="MFSLogicalPage_1"
 devicePages="MFSDevicePage_1">
```

```
<deviceFields xmi:id="MFSDeviceField_1" label="SDATE" length="8">
<messageFields xmi:id="MFSMessageField_1" systemLiteral="date2"
 deviceFields="MFSDeviceField_1"/>
```

```
<deviceFields xmi:id="MFSDeviceField_2" label="CMD" length="8">
<messageFields xmi:id="MFSMessageField_2" length="8"
 deviceFields="MFSDeviceField_2"/>
```

- Web Services use XML Schema types
- Need to map from MFS XMI to XML Schema
- Map from the Device side as opposed to the Message side since end user interacts with the Device side

MFS Functions supported by MFS Web Services

- Support for MFS statements

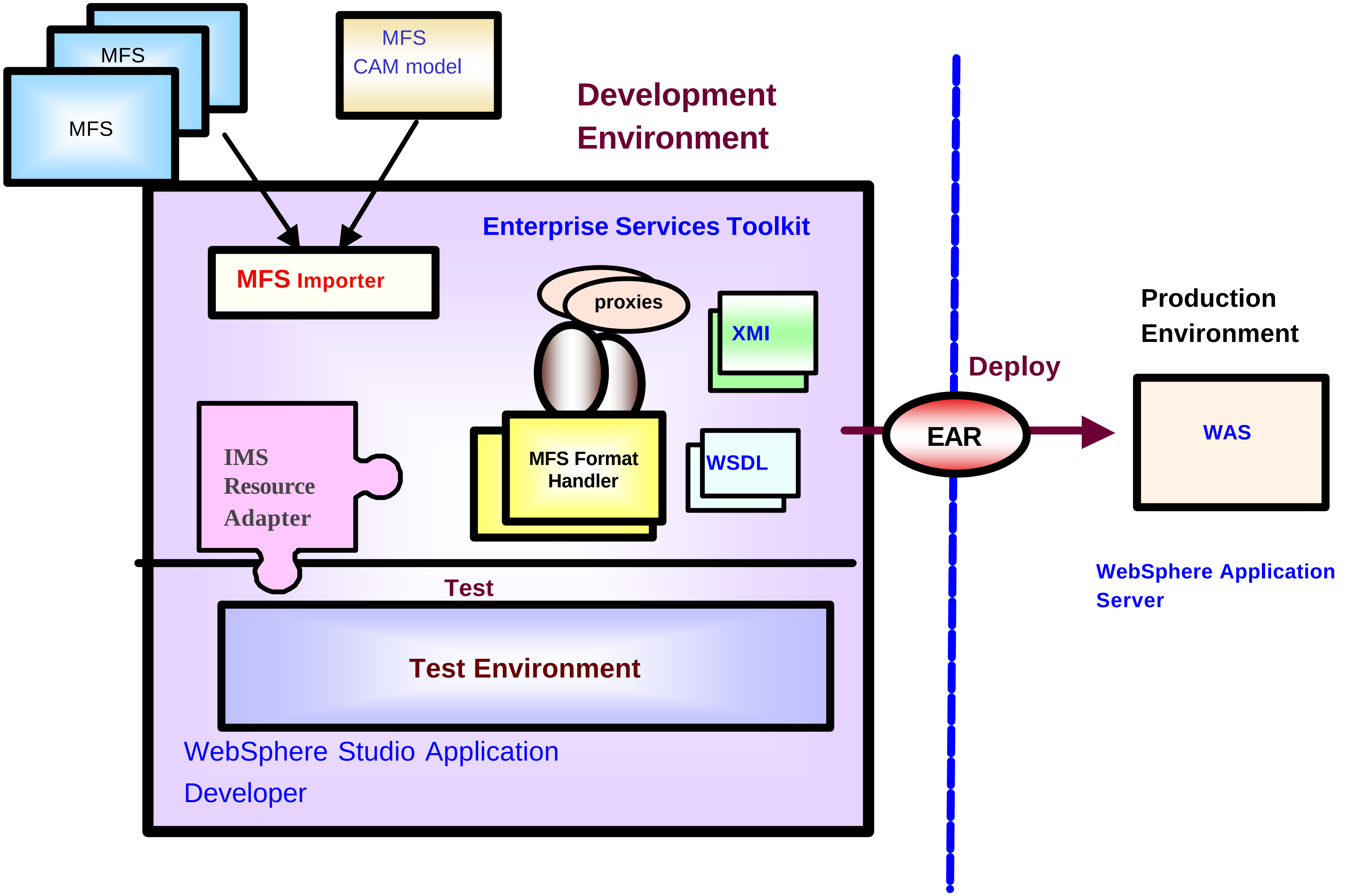
  - MSG
  - MSGEND
  - LPAGE
  - PASSWORD
  - SEG
  - MFLD
  - FMT
  - FMTEND
  - DEV
  - DIV
  - DPAGE
  - DFLD
  - TABLE
  - IF
  - ALPHA
  - COPY
  - DO
  - END
  - ENDDO
  - EQU
  - RESCAN
  - STACK
  - TABLEEND
  - UNSTACK

MFS Functions supported by MFS Web Services

- Support for device types
  - 3270
  - 3270,1
  - 3270,2
  - 3270P
  - 3270-AN
  - Other devices not supported but tolerated for syntax check

- Support for MFS features
  - Application Output with MODNAME
  - DBCS
  - Logical paging
  - Message option 1 and 2 for input
  - Message option 1 and 2 for output
  - Physical pages
  - System literals for date, time, and LPAGENO

MFS Web Services &WSAD-IE 5.0.1

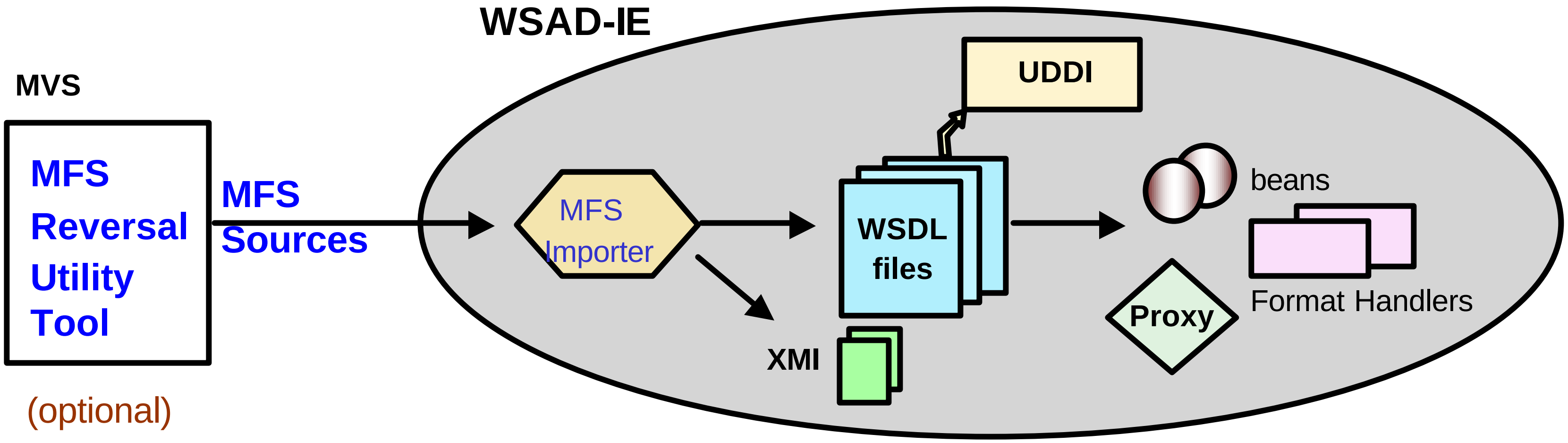


© IBM Corporation 2003

IMS MFS Web Services - Tooling Support

Importing Step 1 out of 4

- No source? No problem!
  - –MFS Reversal Utility Tool
    - ➢MFS Format Library → MFS source files



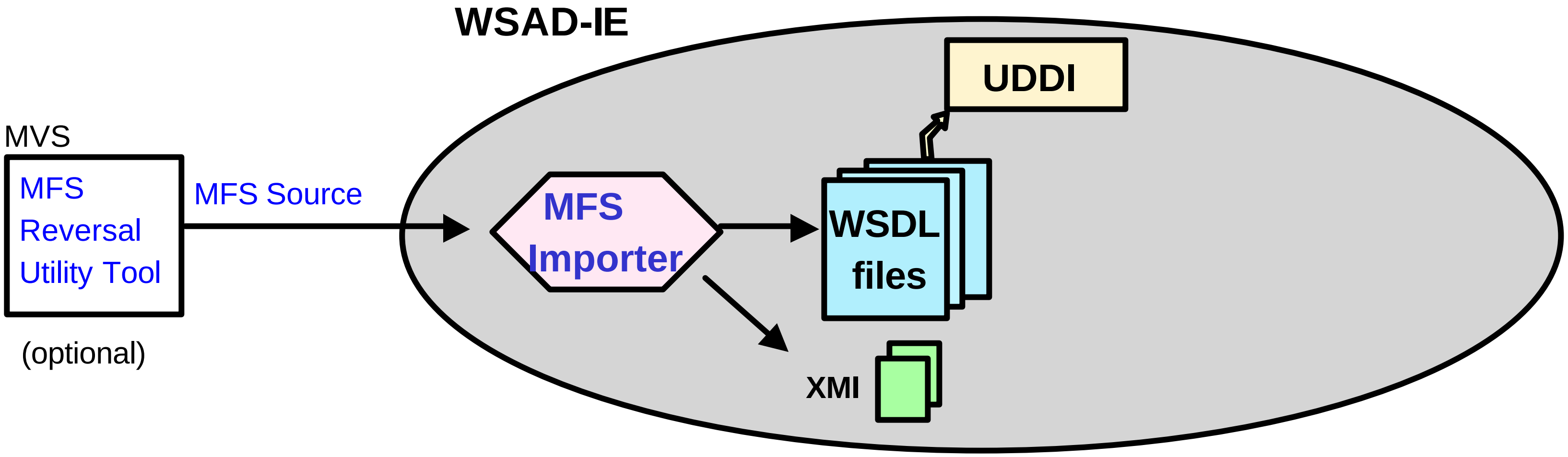
(optional)

© IBM Corporation 2003

IMS MFS Web Services - Tooling Support …

Importing Step 2 out of 4

- MFS Importer
  - –DBCS, Code page
  - –Parse results stored in log file
  - –Generate XMI files
    - •Filter on device type and feature
  - –Generate WSDL (Web Service Description Language) files
    - •Binding, Service, and Interface WSDL files
    - •Publish to UDDI



MFS WSDL Files

- **Interface WSDL file:**
  - **–Input and Output XML Schema types**
  - **–Multiple outputs support**
  - **–Input logical paging support (future item)**
- **Binding WSDL file:**
  - **–Operation properties**
  - **–Format Handler mapping**

```
<binding name="IMSBinding" type="tns:x">
        <ims:binding></ims:binding>
      +<format:typeMapping encoding="ibmmfs">
      +<operation name="runOperation">
</binding>
...
```

Binding. WSDL

```
 -  <complexType name="MID_Page1">
       <sequence>
         + <element name="Field1">
            ...
       </sequence>
    </complexType>
 -  <complexType name="ResponseType">
       <choice>
          <element name="MOD1" type="xsd1:MOD1"/>
          <element name="MOD2" type="xsd1:MOD2"/>
          <element name="MOD3" type="xsd1:MOD3"/>
       </choice>
    </complexType>
 + <complexType name="MOD1">
 +  <complexType name="MOD2">
 +  <complexType name="MOD3">
...
```

Interface.WSDL

- **Service WSDL file:**
  - **–IMS host connection properties**

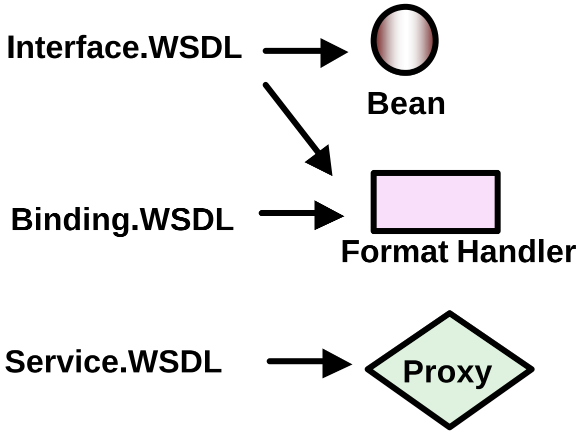
```
<service name="IMSService">

  <port binding="tns:IMSBinding" name="IMSPort">
     <ims:address dataStoreName="SOCKEYE"
                  hostName="ecxxx" portNumber="9999"/>
  </port>
</service>
...
```

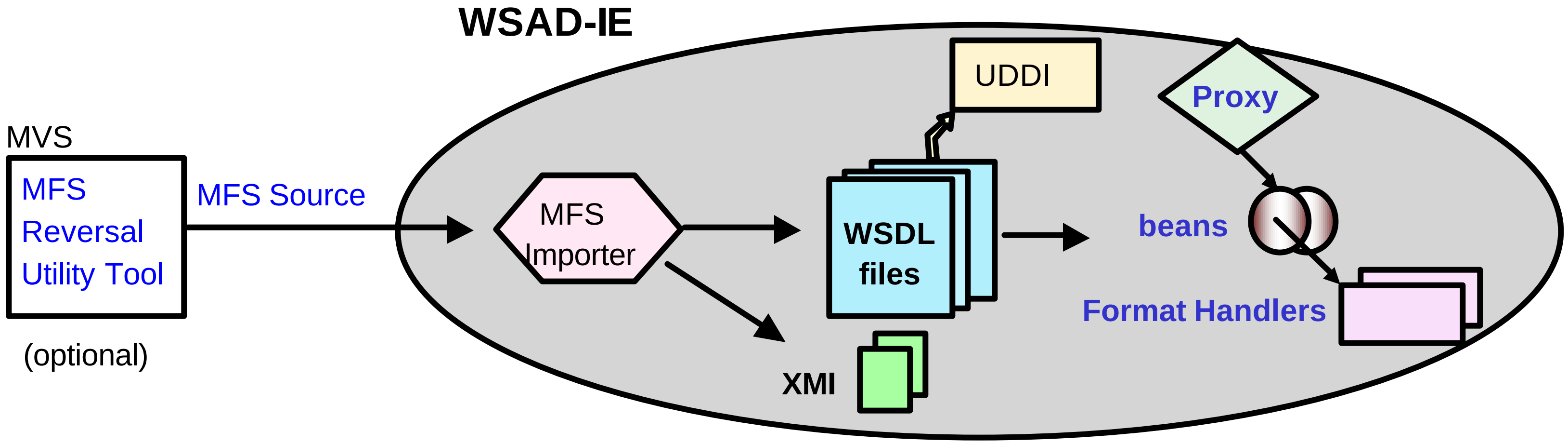
Service. WSDL

IMS MFS Web Services - Tooling Support …

Importing Step 3 out of 4

- Code generators
  - beans, format handlers, proxy

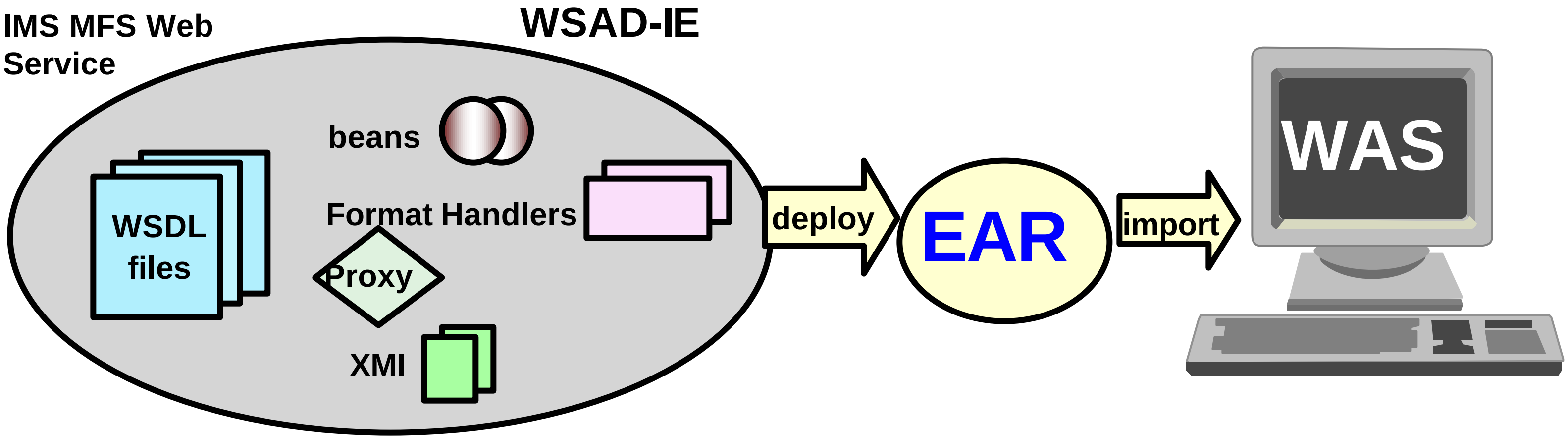




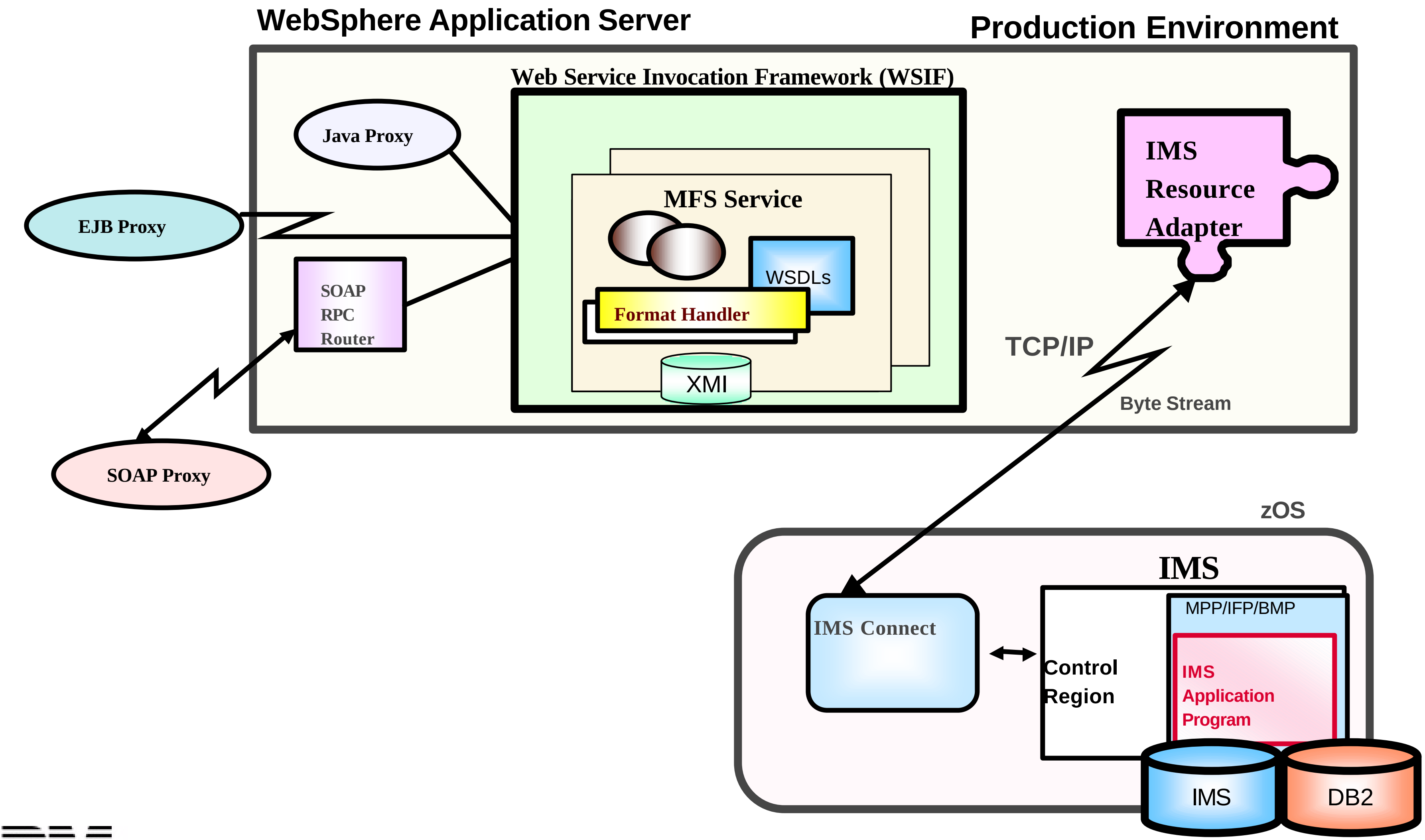
IMS MFS Web Services - Tooling Support …

Importing Step 4 out of 4

- Deployment to server
  - Package EAR file
  - XMI repository
  - Local WAS (Websphere Unit Test Environment) vs. WASEE or zOS WAS



MFS Web Services & WebSphere Application Server



Agenda

- What is MFS and traditional MFS online processing
- Business Challenge
- Modernize MFS Transactions
- MFS Web Services
- MFS Future Requirements

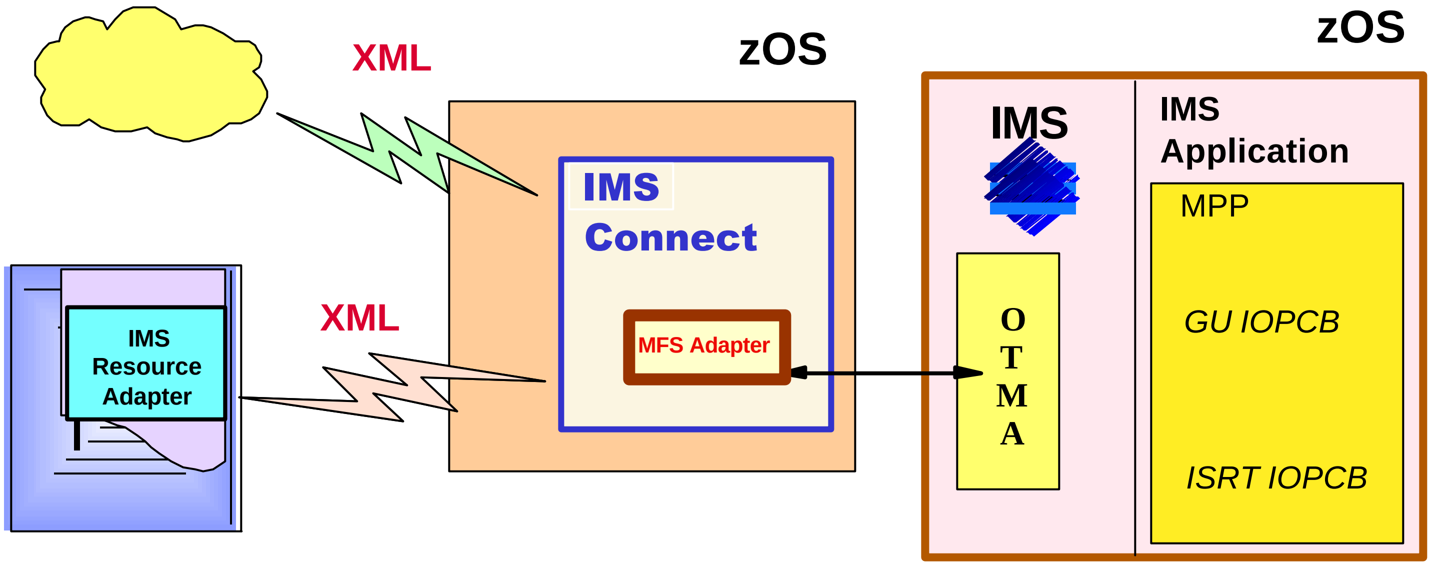
Future Requirements

MFS Web Enablement

- **Much of the MFS XMI metadata is unused in the Web Services paradigm**
  - display information (e.g. Position, Color)
  - control functions (e.g. Next Logical Page)
  - dynamic output based on input
- **MFS data transformation (MFS Adapter) can occur**
  - **Inside WebSphere Application Server, and/or**
  - **Inside IMS Connect**
- **This metadata could be used for rendering displays on new modern devices, extending the use of MFS without modifying existing applications**
  - Web browser
  - cell phones
  - PDAs
- **MFS command line utility**

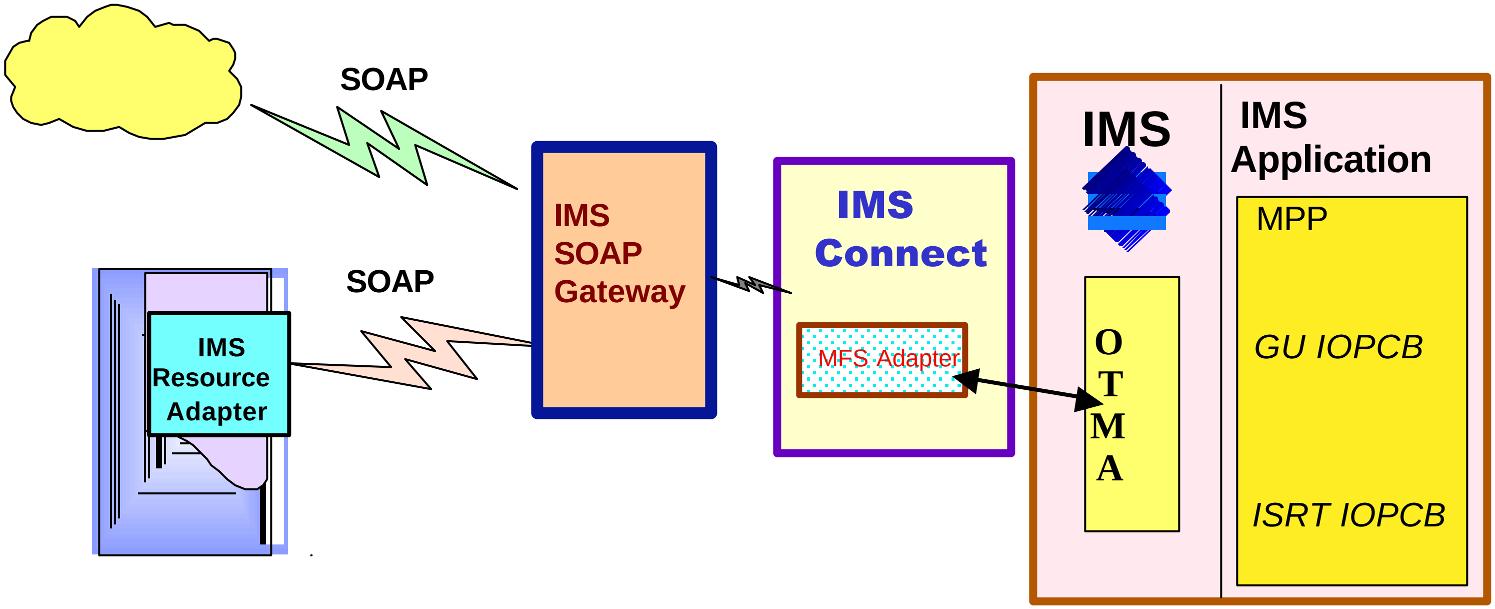
Future Requirements …

- **MFS Web Enablement**
  - **data transformation inside IMS Connect**



Future Requirements …

- **MFS SOAP & Web Services**
  - **IMS SOAP Gateway**



© IBM Corporation 2003

RETURN TO INDEX