



---

## Highlights

- Test IMS applications without an IMS Online Control Region
  - Generates detailed reports on IMS application calls and transaction statistics
  - Fully integrated with IBM Eclipse for IMS Development, Test, and Debugging using IMS Explorer and IBM Debug Tool for z/OS
  - Supports calls to IBM DB2 and IBM MQ
  - Allows IMS transaction testing from IBM WebSphere Application Server (WAS) environments
  - Program Number: 5655-BT4
- 

# IMS Batch Terminal Simulator (BTS) Tool

*Test IMS Applications and Increase Your Productivity*

IMS™ is IBM's premier transaction and hierarchical database management system. IMS was designed for high availability, superior performance, growth and capacity, and full database integrity. The ability to operate and manage this highly complex IMS system and database environment determines the Total Cost of Operation (TCO). The IBM® IMS Tools lowers TCO by equipping IMS system programmers and IMS database administrators (DBAs) with the facilities they need to effectively monitor and manage this mission-critical environment. The IBM IMS Tools provide automation, validation, and auditing of all database and transaction management activities.

The *IMS Batch Terminal Simulator (BTS)* product is an IMS application testing tool. It allows developers to verify application logic and interfaces, input and output 3270 formats, and access to databases. With *IMS BTS*, IMS applications can be executed without starting the IMS Control Region and without having to define input and output terminals to the IMS system. The *IMS BTS* tool reduces the overhead in preparing test environments and reduces the costs and MIPs incurred in creating IMS test environments. The *IMS BTS* tool can be executed in an IMS Batch environment, from a TSO session, and it is fully integrated into an IBM Eclipse Development environment on the workstation and is supported using a special Resource Adapter in an IBM WebSphere Application Server (WAS) environment. The tool generates reports that are useful in improving IMS application design.



## IMS BTS Tool

The *IMS BTS* tool simplifies the process of testing IMS applications. The applications can be tested using IMS Batch environments, a full screen TSO session, using the WebSphere Application Server (WAS), and from a fully integrated IBM Eclipse development environment on a workstation. The IMS BTS tool generates reports about each transaction which include statistics about specific application calls.

## IMS BTS Execution Environment

The *IMS BTS* Tool can run in two distinct ways. The first is in an IMS Batch environment where terminal input and output are simulated. In this way, it supports the IMS Batch environments (DLI and DBB) and the IMS Online Batch environments (BMP and JBP) it is unnecessary to configure actual terminals for testing. The second way is in an Eclipse Development environment where the IMS BTS test apparatus is seamlessly integrated with the IMS Explorer and IBM Debug Tool products.

## IMS Batch Simulation Test Environment

In the IMS Batch Test environment, the *IMS BTS* Tool receives control from TSO using the Full Screen Image Support (FSS) or from the BTSIN input stream. It receives the 3270 input message and either handles the message directly or it calls the IMS Message Format Service (MFS) to simulate the formatting of the message. This avoids the need to configure the actual terminal. The IMS BTS tool then schedules the IMS application program and intercepts the actual application program calls to gather information about the application as it is executing. The output message is either formatted in the MFS format or it is sent to the BTSOUT output data file. This flow is shown in Figure 1.

The IMS BTS tool supports the /SET and /RESET commands to set terminal or transaction destinations. It supports the /FORMAT command to select specific MFS Formats for testing. The tool also supports the /EXIT command to terminate a Conversational Transaction. It is also possible to tell the IMS BTS tool to ABEND or to create a SNAP ABEND during DEBUG mode. The ABEND

produces a U4091 while the SNAP ABEND sends the debugging data to the BTSDEBUG data set without terminating.

In this IMS Batch Test environment, the IMS BTS tool can also be integrated with the IBM Debug Tool for z/OS product. The IBM Debug Tool also runs in batch mode and can examine, monitor, and control the execution of an IMS application program for testing purposes. The IBM Debug Tool can run in Batch Mode or interactively in Full Screen Mode and it supports programs written in Assembler, C, C++, COBOL, and PL/I.

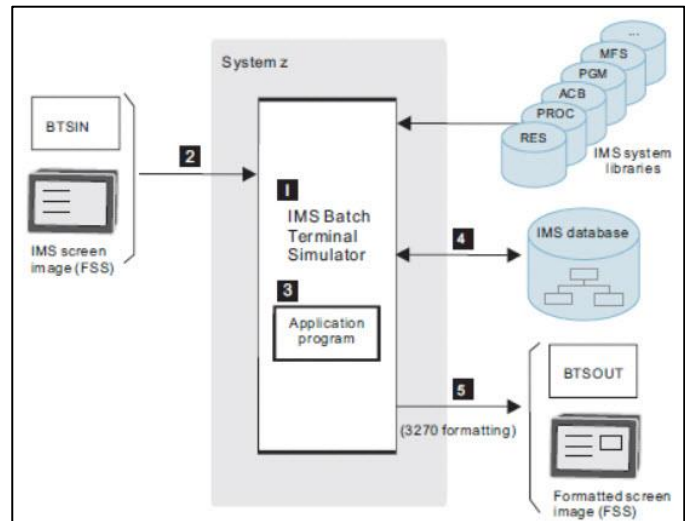


Figure 1: IMS BTS Batch Simulation Test Environment

## Integrated Eclipse Development Environment

The *IMS BTS* Tool also runs in a fully integrated Eclipse Development environment with the IMS Explorer and IBM Debug Tool products as shown in Figure 2. This Graphical User Interface (GUI) provides an excellent application development environment for programming, debugging, and testing IMS applications. The IMS BTS product comes complete with Eclipse Wizards to set up runtime parms and library and data definitions easily.

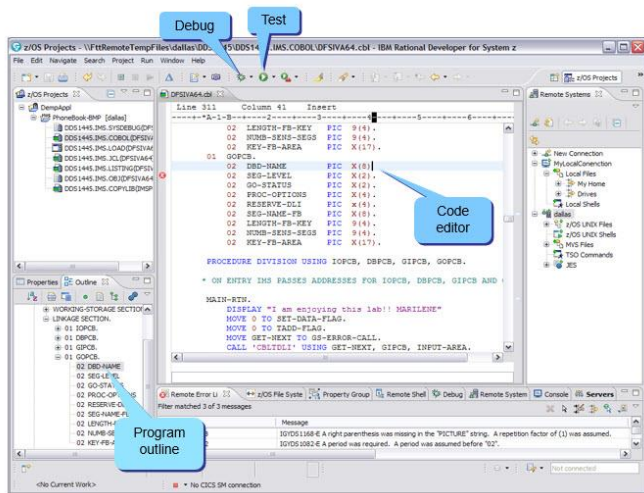


Figure 2: IMS BTS Eclipse Integrated Development Environment

### IMS BTS Distributed Resource Adapter

In a distributed environment, IMS transactions can be executed using the IBM WebSphere Application Server (WAS) using the IMS Transaction Manager Resource Adapter (TMRA). These TMRA transactions connect to IMS over TCP/IP using the IMS Connect address space as shown in Figure 3.

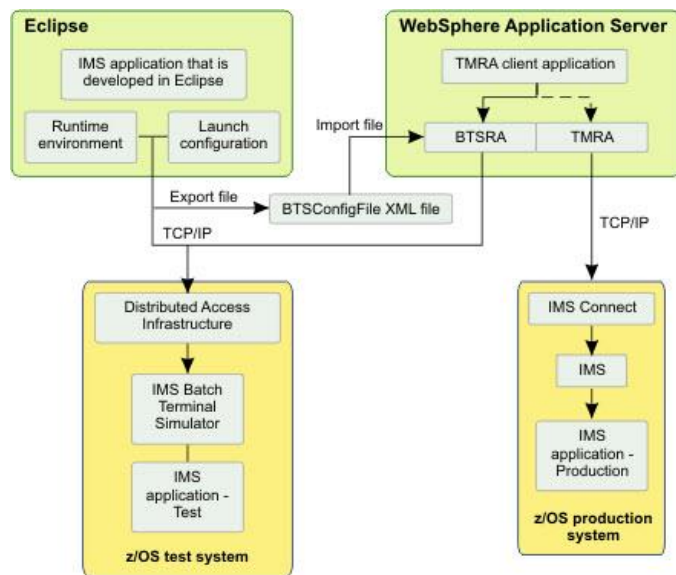


Figure 3: IMS BTS Distributed Eclipse Architecture

The IMS BTS tool provides its own resource adapter called the IMS BTS Resource Adapter (BTSRA) to intercept and direct WAS transactions to the IMS Batch Terminal Simulator using the Distributed Access Infrastructure (DAI) address space. In the Eclipse Development environment, the IMS System Programmer sets up the

Runtime environment which contains the libraries that the IMS BTS tool requires. They also set up BTS Launch Configurations which contain parameters specific to individual IMS applications. Together, the Runtime Environment and Launch Configurations are Exported as XML files from the Eclipse environment to other test environments. Since there are no changes required to the IBM WAS application when using the BTSRA in place of the TMRA, it is very easy to switch between testing and production environments when testing these IMS applications.

When the IMS application is executed, it is run in the IMS BTS address space and the output can be viewed right next to the application program logic as shown in Figure 4. If corrections are needed, the IMS application can be changed, recompiled, and re-submitted very easily in the Eclipse development environment.

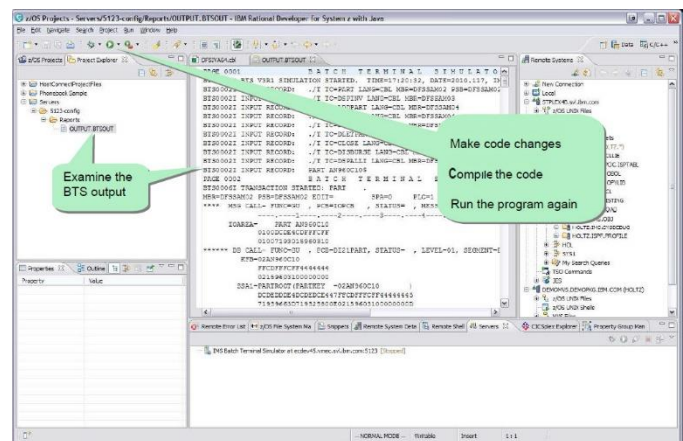


Figure 4: IMS BTS Examining Program Output in Eclipse Environment

### IMS BTS Report Generation

The IMS BTS Tool provides statistical reports on application program calls that help IMS application programmers analyze the results of their programs easily to improve testing productivity. For each application program, a summary of message and database activities are shown including the number of calls, elapsed time, function codes and status codes. These reports further breakdown these calls within the function codes by segment names which is very useful for program analysis. There are reports showing the status codes received for message and database DL/I calls which helps IMS application programmers determine which calls are working correctly. These reports provide details of

each application call including the Segment Search Argument (SSA), the Scratchpad Area (SPA), the Key Feedback Area, and the I/O Area. The reports are available under TSO or as output data sets and the data set can become a regression test file for subsequent testing.

There are also special reports detailing IMS ICAL call activities where an IMS application, running in an IMS TM environment, sends a synchronous request for data or services to a non-IMS application program or service running in a z/OS or distributed environment or to initiate a synchronous program switch to another IMS transaction. There are additional reports showing calls to DB2 as shown in Figure 5 and to WebSphere MQ application programs including the number of calls and the elapsed time by call type as shown in Figure 6.

```

PROGRAM ANALYSIS STATISTICS BY SQL STATEMENT FOR TRANSACTION: REPOSQ
MBR=SSASQL PSB=DFSSAM02 FROM 2011/06/27 15:44:07 FOR 1.742729SEC
SQL-STATEMENT      CALLS  WARNS  ERRORS  MAX-ELAPS-SEC  AVG-ELAPS-SEC
OPEN                1      0      0      0.000998      0.000998
CLOSE               1      0      0      0.000014      0.000014
SELECT              3      0      0      0.001495      0.001312
FETCH               1      0      0      0.000025      0.000025
INSERT              2      0      0      0.006540      0.003832
DELETE              0      0      0      0.000000      0.000000
UPDATE              1      0      0      0.002153      0.002153
PREPARE             0      0      0      0.000000      0.000000
DESCRIBE            0      0      0      0.000000      0.000000
EXECUTE             0      0      0      0.000000      0.000000
EXECUTE IMMEDIATE  0      0      0      0.000000      0.000000
OTHER               7      0      0      0.782744      0.153567
-----
** TOTAL           16      0      0      0.782744      0.068110
  
```

Figure 5: IMS BTS DB2 Report by SQL Statement

```

PROGRAM ANALYSIS STATISTICS BY MQI FUNCTION FOR TRANSACTION: MQREPORT
MBR=SSAMQI PSB=DFSSAM02 FROM 2011/06/29 15:08:14 FOR 0.275878SEC
MQI-FUNCTION      CALLS  WARNS  ERRORS  MAX-ELAPS-SEC  AVG-ELAPS-SEC
MQCONN            1      0      0      0.050573      0.050573
MQDISC            1      0      0      0.014761      0.014761
MQOPEN            1      0      0      0.008358      0.008358
MQCLOSE           1      0      0      0.007690      0.007690
MQGET             5      0      1      0.018390      0.012261
MQPUT             2      0      0      0.022744      0.022310
MQPUT1            2      0      0      0.007415      0.007391
MQINQ            1      0      0      0.007277      0.007277
MQSET             1      0      0      0.008225      0.008225
MQCMIT           1      0      0      0.009272      0.009272
MQBACK           1      0      0      0.007621      0.007621
MQCONNX          0      0      0      0.000000      0.000000
MQSUB            0      0      0      0.000000      0.000000
MQSUBRQ          0      0      0      0.000000      0.000000
MQSTAT           0      0      0      0.000000      0.000000
OTHER            0      0      0      0.000000      0.000000
-----
** TOTAL           17      0      1      0.050573      0.013793
  
```

Figure 6: IMS BTS MQ Report by MQI Function

### For more information

To learn more about the IBM IMS Tools product line, please contact your IBM representative or IBM Business Partner, or visit: [ibm.com/software/data/db2imstools/products/ims-tools.html](http://ibm.com/software/data/db2imstools/products/ims-tools.html)

© Copyright IBM Corporation 2016

IBM Corporation  
Route 100  
Somers, NY 10589

Produced in the United States of America  
December, 2016

IBM, the IBM logo, ibm.com, and IMS are trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the web at "Copyright and trademark information" at: [ibm.com/legal/copytrade.shtml](http://ibm.com/legal/copytrade.shtml)

This document is current as of the initial date of publication and may be changed by IBM at any time. Not all offerings are available in every country in which IBM operates. THE INFORMATION IN THIS DOCUMENT IS PROVIDED "AS IS" WITHOUT ANY WARRANTY, EXPRESS OR IMPLIED, INCLUDING WITHOUT ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND ANY WARRANTY OR CONDITION OF NON-INFRINGEMENT. IBM products are warranted according to the terms and conditions of the agreements under which they are provided.



Please Recycle