

# E-BUSINESS SUITE APPLICATIONS R12 (R12.2.5) ORDER-TO-CASH (BATCH) BENCHMARK - USING ORACLE11g ON ORACLE’S DATABASE CLOUD SERVICE

As a global leader in e-business applications, Oracle is committed to delivering high performance solutions that meet our customers’ expectations. Business software must deliver rich functionality with robust performance. This performance must be maintained at volumes that are representative of customer environments.

Oracle benchmarks demonstrate our software’s performance characteristics for a range of processing volumes in a specific configuration. Customers and prospects can use this information to determine the software, hardware, and network configurations necessary to support their processing volumes.

The primary objective of our benchmarking effort is to provide as many data points as possible to support this important decision.

## SUMMARY OF RESULTS

This batch benchmark test was run on a 4-core server.

Batch Workload				
100,000 Order/Inv. Lines	Threads	Time (Min)	Hourly Order Line Throughput	
HVOP	16	3.68	1,630,208	
Pick Release	16	32.25	186,047	
Ship Confirm	1	0.22	27,272,727	
Interface Trip Stop	16	3.10	1,935,484	
Inventory	500	13.77	435,730	
Auto Invoice	16	7.58	791,557	
Revenue Recognition	8	4.68	1,282,051	
Accounting Submit	1	1.00	6,000,000	
Accounting Create	1	3.68	1,630,435	
<b>Sel. Process Total:</b>		69.96	85,763	
<b>Parent Proc Total:</b>		74.67	80,354	
<b>Wall Clock Duration*</b>		74.87	80,139	

More processes are included in this summary than were reported in either 12.0.4 or 11i. Consequently, these results cannot be compared to earlier releases. Note that the hourly throughput numbers mentioned above are linear extrapolations. Many factors can influence performance and your results may differ.

\* The “Wall Clock Duration” includes all of the job scheduling and management activity (parent process) as well as some idle intervals due to polling or waiting for all workers in a particular process to complete prior to kicking off the subsequent process. These intervals would not increase substantially, if at all, as the workload size is increased. Consequently, the throughput for larger workloads would converge towards the “Parent Process Total:” value.

## BENCHMARK PROFILE

In September 2016, Oracle conducted a benchmark in Pleasanton, CA to measure the batch performance of the Oracle E-Business Standard Benchmark processes in an environment running Oracle E-Business Suite R12 (12.2.5). The database server used Oracle11g™ (11.2.0.4.5) running on Oracle’s Public Cloud 16.2.2 Database Service with Oracle® Linux® 6.6 (64-bit) OS. Moreover, the instance of 4 OCPU, 8 threads, 30 GB used two of Oracle’s Public Cloud Storage Latency Volumes for data storage and redo log storage.

The benchmark measured the Order Management batch business process hourly throughputs for a large database model. Testing was conducted in a controlled environment with no other applications running. **The goal of this Benchmark was to obtain reference batch throughputs for Oracle E-Business Suite R12 Benchmark on an Oracle’s Database Cloud Service.**

## BENCHMARK METHODOLOGY

E-Business Suite R12 Benchmark batch processes are initiated from a benchmark-provided SQL script.

The batch workloads were run as standard concurrent processes via the concurrent manager.

Figure 1 shows the configuration used for this benchmark run.

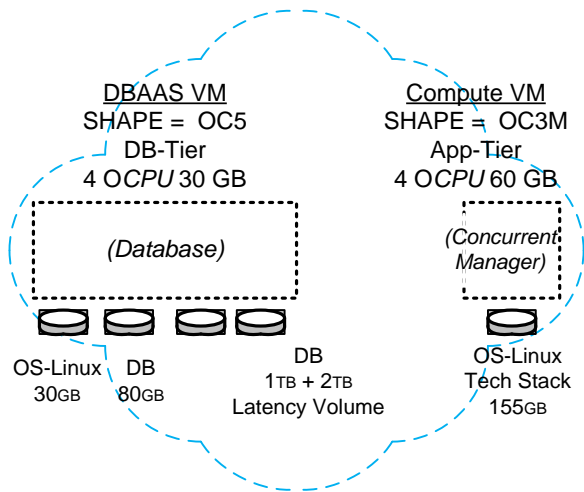


Figure 1: Virtualization Resource Apportionment

## BENCHMARK BUSINESS PROCESSES

This E-Business Suite benchmark consists of a batch flow with seven metered processes.

### Batch Order-to-Cash Processes

Business Process	Number of Threads Used
HVOP	16 (16)
Pick Release	16 (16)
Interface Trip Stop	16 (16)
Inventory	16 (500)
Auto Invoice	16 (16)
Revenue Recognition	16 (8)
Accounting Processes	1 (1)

Note that while 16 threads (workers) were entered at setup, the actual number of child processes spawned may have been larger (Inventory) or smaller (Revenue Recognition and Accounting).

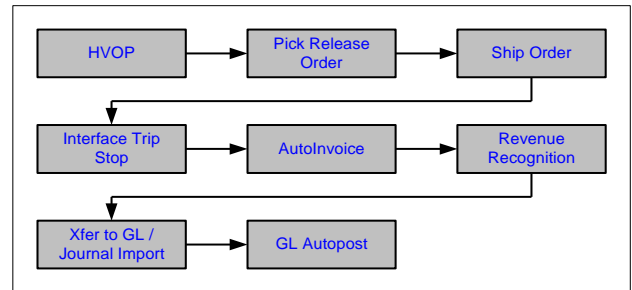


Figure 3: Order-to-Cash Process Flow

**High Volume Order Processing (HVOP):** The HVOP program processes orders by reading the rows from the Order Management Interface tables and converting the interface records into permanent order headers and their respective order lines. The orders are then booked and advanced to the shipping state.

**Pick Release (and Ship Confirm):** Pick Release finds and releases the eligible delivery lines that meet the release criteria, and creates move orders. The process of transacting move orders creates a reservation and determines the inventory source sub-inventory.

Ship Confirm is the process of confirming that items have shipped. When a delivery is ship-confirmed, Shipping Execution confirms that the delivery lines associated with the delivery have shipped.

## Batch Order-to-Cash Processes Continued

**Interface Trip Stop:** The deliveries created in the previous step are then assigned to trips, which may involve multiple stops depending upon the shipping addresses of the deliveries. SRS has been modified to accept Organization code as a parameter and process the trip stops for the specified organization. Interface Trip Stop - SRS has also been enhanced to spawn multiple child processes to process trip stops in parallel. The parameter Stops per Batch is used to specify the number of stops to be processed by each thread of the Interface Trip Stop - SRS. Interface Trip Stop - SRS has also been enhanced to defer the Inventory Interface processes. In the E-Business Suite kit, this profile is set to Yes so that the Inventory Interface transactions are processed in the background by the Inventory transaction manager.

**INV Material:** The material transaction manager is configured to execute material transaction by periodic concurrent request submissions and by direct submission of multiple transaction managers via the benchmark SQL script. The execution interval is set to 5 minutes.

**Auto-Invoice:** The Auto-Invoice process is used to import invoices, credit memos, debit memos, and on-account credits. 'Receivables' ensures that the data imported is accurate and valid.

**Revenue Recognition:** Revenue Recognition program generates the revenue distribution records for the invoices and credit memos that use Invoicing and Accounting Rules. Accounting rules were assigned to recognize revenue over a 12-month accounting period. The Revenue Recognition program will create distribution records for the invoices and credit memos that are created in Receivables and imported using Auto-Invoice.

**Transfer to General Ledger & Journal Import:** The General Ledger Interface program transfers Receivables transaction accounting distributions to the general ledger interface table (GL\_INTERFACE) and creates either detailed or summarized journal batches. "Receivables" creates un-posted journal entries in general ledger and executes Journal Import from Oracle General Ledger. It posts journal batches in Oracle General Ledger to update account balances.

**General Ledger Auto-post:** This posts journal batches to update the account balances of the detail and summary accounts. It can post actual budget or encumbrance journal batches.

## BENCHMARK RESULTS

Batch Business Metrics	Achieved Output
<b>Order to Cash</b>	
Number of Order Lines Created/Booked	100,000
Number of Order Lines Picked	100,000
Number of Order Lines Ship Confirmed	100,000
Number of Order lines Interface Trip Stopped	100,000
Number of Invoice Headers Created	100,000
Number of Invoice Lines Created	200,000

**Table 1: Batch Transactions Completed**

100,000 order lines were processed in this test. Table 2 shows the processing time in minutes.

Batch Workload				
100,000 Order/Inv. Lines	Threads	Time (Min)	Hourly Order Line Throughput	
HVOP	16	3.68	1,630,208	
Pick Release	16	32.25	186,047	
Ship Confirm	1	0.22	27,272,727	
Interface Trip Stop	16	3.10	1,935,484	
Inventory	500	13.77	435,730	
Auto Invoice	16	7.58	791,557	
Revenue Recognition	8	4.68	1,282,051	
Accounting Submit	1	1.00	6,000,000	
Accounting Create	1	3.68	1,630,435	
<b>Sel. Process Total:</b>		69.96	85,763	
<b>Parent Proc Total:</b>		74.67	80,354	
<b>Wall Clock Duration*</b>		74.87	80,139	

**Table 2: Order-to-Cash (16 Worker) Batch Performance**

R12.2.5 Application changes, data model additions and test methodology improvements render direct comparison to previous Oracle E-Business release 12.1.3, 12.0.4, 11.5.10 and 11.5.9 results invalid.

## SERVER PERFORMANCE

Figure 4 shows the average CPU utilization on the server. The value shown is the average across the processors (4 cores total, 8 vCPUs). Note that the sampling rate yielded few data points to average for the briefest processes.

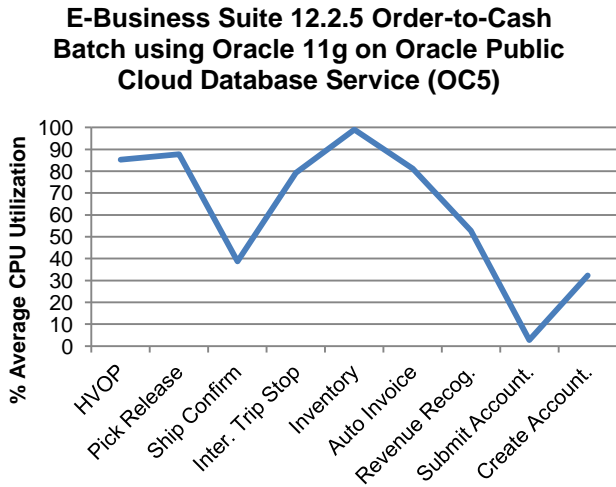


Figure 4: Average Database CPU Utilization

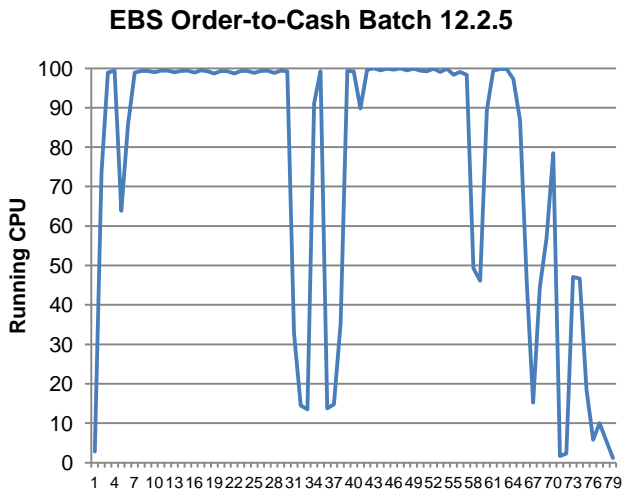


Figure 5: Running CPU Utilization

Workload	% User	% System	% Wait	% Idle
HVOP	76.83	7.13	1.28	14.75
Pick Release	72.97	14.36	0.38	12.29
Ship Confirm	30.1	5.2	3.6	61.2
Interface Trip Stop	66.47	11.47	1.27	20.80
Inventory	85.47	13.41	0.10	0.98
Auto Invoice	71.46	7.82	1.93	18.78
Rev. Recognition	42.72	5.46	4.58	47.20
Submit Accounting	1.8	0.5	0.5	97.2
Create Accounting	25.35	3.33	3.65	67.65
Wall Clock Avg.	64.85	10.75	1.08	23.31

Table 3: Average CPU Utilization Breakout (16 Workers)

Average GB Used	O-to-C
Database Server	30 GB

Table 4: Average Memory Utilization

## I/O PERFORMANCE

Two Oracle Public Cloud Storage Latency Volumes were used for storage of tables and indexes. The batch workload requires optimal I/O performance.

I/O Performance		
Transfers/Sec	Avg	3,728
	Peak	18,011
Write KB/Sec	Avg	55,813
	Peak	149,481
Read KB/Sec	Avg	5,035
	Peak	51,161

Table 5: Average I/O Utilization Breakout

## DATA COMPOSITION DESCRIPTION

Major data components for the model under test are summarized in the following table.

Application	Business Objects	Large/Extra-Large Model
TCA	Organizations	1,100,000
	Contacts	4,900,000
	Contact Points	3,700,000
	Accounts	1,100,000
	Account Sites	1,090,000
	Account Site Uses	2,180,000
Contracts	Contracts	222,000
Install Base	Instances	1,300,000
	Trackable Items	5
	Items	1,100,000
HR	Managers	800
	Employees	250,000
	Payroll Users	250,000
	Users	20,000
	Credit Card Entries	4,000,000
	Supplier(s)	10,000
Assets	Asset Categories	984
General Ledger	GL Code Combinations	93,417

**Table 7: Data Composition**

## PERFORMANCE INITIALIZATION

The init.ora file has to have this added.

```
filesystemio_options =setall
```

## APPLICATION TUNING

### Database:

```
create index oe_order_lines_all_N20
on ont.oe_order_lines_all(header_id, flow_status_code);
create index wsh.wsh_new_deliveries_tn8
on
wsh.wsh_new_deliveries(batch_id,hash_value,status_code);
create index inv.mtl_transactions_interface_tn7
on
inv.mtl_transactions_interface(inventory_item_id,organizati
on_id);
create index inv.mtl_transactions_interface_tn8
on
inv.mtl_transactions_interface(process_flag,transaction_mod
e,lock_flag);
```

## BENCHMARK ENVIRONMENT

### HARDWARE CONFIGURATION

A single DBaaS instance version 16.2.2 on Oracle Database Cloud Service with Shape OC5 (4 OCPU as 8 vCPU) was used. It was equipped with the following:

- 4 OCPU (8vcpu) running on 2.29 GHz Intel® Xeon™ E5-2699 v3
- 30 Gigabytes of Memory (~30 peak)
- Two Oracle Public Storage Volumes for a total of 110 GB were used to host Linux and Oracle 11g Database software.

### **Application Server:**

A single COMPUTE Instance of Oracle's Public Cloud 16.2.2 was used for this test. 1 × Oracle Linux COMPUTE Instance with Shape OC3M was used as an application server to host the Concurrent Manager.

- 4 OCPU (8vcpu) running on 3.00 GHz Intel® Xeon™ E5-2690 v2
- 60 Gigabytes of Memory (~14 GB used at peak load)
- One Oracle Public Storage Volume for a total of 155 GB was used to host Linux and the Application Tier software.

## SOFTWARE VERSIONS

Oracle E-Business Suite R12 (12.2.5)

Oracle 11g™ 11.2.0.4.5 (64-bit)

Oracle Linux 6.6 (64-bit) on the database server.

Xen 4.3.1 OVM

Java HotSpot™ 64-bit server VM (build 14.3-b01), mixed mode

The following Java™ Standard Edition (SE) versions have all been used in the Oracle Apps environment:

- Java 1.6.0\_17-b04

Glossary and Acronyms:

DBaaS Database as a Service

OASB Oracle Applications Standard Benchmark

OCPU Oracle CPU (1 physical core, for 2 execution threads with Hyper threading enabled)

RAC Real Applications Clusters



Oracle

Applications Performance & Benchmarks

500 Oracle Parkway

Redwood Shores, California 94065

Tel 650/506-7000

Fax 650/506-7000

Email [eBSkit\\_us@oracle.com](mailto:eBSkit_us@oracle.com)

World Wide Web <http://www.oracle.com>



E-Business R12 Order to Cash Batch  
September 2016

Oracle Corporation  
World Headquarters  
500 Oracle Parkway  
Redwood Shores, CA 94065  
U.S.A.

Worldwide Inquiries:  
Phone: +1.650.506.7000  
Fax: +1.650.506.7200

[oracle.com](http://oracle.com)



Oracle is committed to developing practices and products that help protect the environment

Copyright © 2016, Oracle and/or its affiliates. All rights reserved. This document is provided for information purposes only and the contents hereof are subject to change without notice. This document is not warranted to be error-free, nor subject to any other warranties or conditions, whether expressed orally or implied in law, including implied warranties and conditions of merchantability or fitness for a particular purpose. We specifically disclaim any liability with respect to this document and no contractual obligations are formed either directly or indirectly by this document. This document may not be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without our prior written permission.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. UNIX is a registered trademark licensed through X/Open Company, Ltd. 1010

**Hardware and Software, Engineered to Work Together**