



## E-BUSINESS SUITE PAYROLL 11i (11.5.10) USING ORACLE10g ON A UNISYS ES7000/one Enterprise Server

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

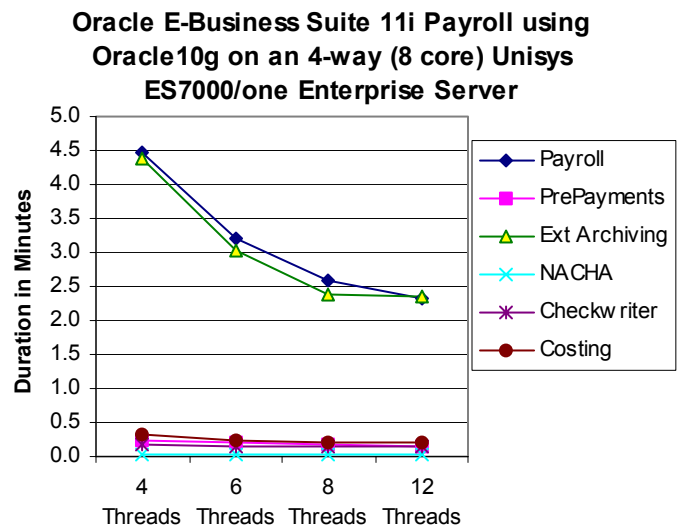
5000 Employees (12 Threads)	Time	Hourly Throughput
Payroll Process	2.32 min	129,310
PrePayments	0.13 min	2,307,692
Ext. Proc. Archive	2.37 min	126,582
NACHA	0.03 min	10,000,000
Check Writer	0.13 min	2,307,692
Costing	0.2 min	1,500,000
<b>Total</b>	<b>5.18 min</b>	<b>57,915</b>

Note that the hourly throughput numbers above are linear extrapolations. Additionally, the times are rounded to two decimal places, while the throughputs are computed on the extended (non-rounded) values in a spreadsheet. Many factors can influence performance and your results may differ.

### BENCHMARK PROFILE

In August 2006, Oracle and Unisys conducted a benchmark in Pleasanton, CA to measure the batch performance of the Payroll processes in Oracle's E-Business Suite Payroll 11i (11.5.10) with Oracle10g™ 10.2.0.2 for Linux on a 4-way Intel (8 cores) Unisys® ES7000/one® Enterprise Server database/application server, running Red Hat® Linux™ RHEL 4 Update 3.

The benchmark measured 'Payroll' application business process runtimes for a small database model. Testing was conducted in a controlled environment with no other applications running. The tuning changes, if any, were approved by Oracle Applications Development and will be generally available in a future update. **The goal of this Benchmark was to obtain reference performance results for Oracle's E-Business Suite 11i Payroll on a Unisys ES7000 Enterprise Server.**



**Figure 1: Oracle Payroll 11i Processing Times**

The benefits from running increasing numbers of threads are more pronounced on the two longer processes, Payroll and External Archiving. Note that the gains decline at higher numbers of parallel threads.

## METHODOLOGY

E-Business Suite 11i Payroll 11.5.10 batch processes can be initiated from a browser. For this benchmark, the payroll related jobs were executed by submitting concurrent manager requests.

The batch processes were run by submitting concurrent requests and these requests were then executed by the concurrent manager. The payroll program automatically divides the employee population across the number of sub-processes configured by the user (AKA threads). The Payroll multithreading engine is fully load balanced ensuring that each of the threads has a roughly equal amount of work to do. For this benchmark 4, 6, 8 or 12 threads were used.

Business Process	Threads	Process Type
Payroll Process	4, 6, 8 or 12	Pro-C
PrePayments	4, 6, 8 or 12	Pro-C
Ext. Process Archive	4, 6, 8 or 12	Pro-C & PL/SQL
NACHA	4, 6, 8 or 12	Pro-C
Check Writer	4, 6, 8 or 12	Pro-C & Oracle Report Writer
Costing	4, 6, 8 or 12	Pro-C

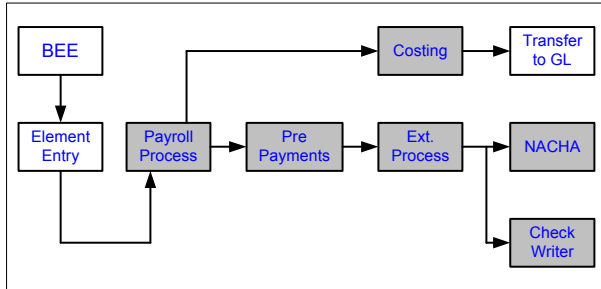


Figure 2: Process Flow

## BUSINESS PROCESSES

The Oracle E-Business Suite 11i Payroll processes tested are as follows:

**Payroll Process:** (Pro-C) Identifies all employees to be processed and performs calculations required to complete the gross to net calculation including earnings, deductions, and taxes. The specific groups of employees processed can be controlled by multiple parameters to the payroll process including the ability for a user to define a rules based set of employees.

**PrePayments:** (Pro-C) Distributes the net pay for each employee across the various payment methods (Direct Deposit, Check, or Cash). This a single payroll process or can be run across multiple payroll processes.

**External Process Archive:** (Pro-C, PL/SQL) Replicates the results of the Payroll run into a separate archive for audit purposes. This data is primarily used for Payslips (Both printed and on line), as a source for check and direct deposit printing, third party interfaces, and tax remittance reporting.

**NACHA:** (Pro-C) This is the US version of the Global Direct Deposit process which creates the bank interface file as per NACHA rules based on the rules in the Pre Payment process.

**Check Writer:** (Pro-C and Oracle Report Writer) This process allocates check numbers and creates/prints the payroll check and associated paper payslip.

**Costing:** (Pro-C) This process associates the payroll transaction data with the General Ledger (GL) accounts in preparation for transfer of the data to GL. This process uses a sophisticated hierarchical rules based engine to determine the mapping of the HRMS data and payroll results to the GL accounts.

## BATCH RESULTS

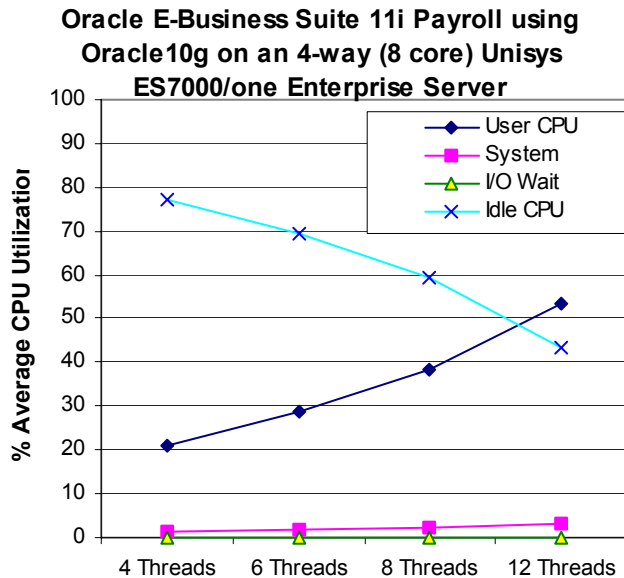
5000 employees were processed for the semi-monthly payroll in this test. Table 1 shows the processing time in minutes for increasing numbers of parallel threads.

5000 Employees	4 Threads	6 Threads	8 Threads	12 Threads
Payroll Process	4.48	3.22	2.6	2.32
PrePayments	0.25	0.2	0.18	0.13
Ext. Proc. Archive	4.38	3.02	2.38	2.37
NACHA	0.03	0.03	0.02	0.03
Check Writer	0.17	0.13	0.13	0.13
Costing	0.33	0.25	0.22	0.2
Net Totals:	9.65	6.85	5.53	5.18

Table 1: Batch Performance

## SERVER PERFORMANCE

Figure 3 shows the average CPU utilization for each process. The value shown is the average across all 4 processors (8 cores) in the combined database server/application server.



**Figure 3: Average CPU Utilization**

No processes were I/O bound in these tests.

	4 Threads	6 Threads	8 Threads	12 Threads
<b>User CPU</b>	21.12	28.59	38.18	53.22
<b>System</b>	1.36	1.84	2.36	3.03
<b>I/O Wait</b>	0.13	0.19	0.15	0.15
<b>Idle CPU</b>	77.39	69.38	59.27	43.55

**Table 2: Average CPU Utilization**

## DATA COMPOSITION DESCRIPTION

History data was created prior to the timed benchmark runs. This yielded data for 6 semi-monthly pay periods.

	Medium Model
<b># of employees</b>	Configurable (5000)
<b># of assignments</b>	Equal to number of employees (5000)
<b># of elements</b>	11
<b># of payroll</b>	1
<b># of consolidation sets</b>	1
<b># of Location</b>	2

**Table 3: Payroll Data**

	Criteria & Deductions
<b>Elements</b>	Regular Salary
	Federal Income Tax (FIT)
	FUTA
	SS_EE
	SS_ER
	Medicare_EE
	Medicare_ER
	Vertex
	Workers Compensation
	Union Dues
	After-Tax Life
<b>Payroll</b>	Semi-Monthly
<b>Locations</b>	Texas
	Orlando (FL)

**Table 4: Workload Factors**

## TUNING RECOMMENDATIONS

All tablespaces are locally managed. The Undo Tablespace and Redo log files were placed on raw devices.

Apply patch 4079903. This patch enables the "Single Latest Balance Table Upgrade" for US legislation.

Follow the recommendations in Metalink document 216205.1, "Database Initialization Parameters and Configuration for Oracle Applications 11i".

Note that excessive stats gathering may adversely affect database performance. Use caution when gathering stats across all schemas. Key to performance gains are appropriate stats on the following schemas: APPS, APPLSYS, and HR.

Set the number of target processes for Standard Manager greater than the threads defined in the database (PAY\_ACTION\_PARAMETERS). Also set CHUNK\_SIZE in the PAY\_ACTION\_PARAMETERS table to 20.

Run the "Generate Run Balance" concurrent program to validate the balance. This will improve the performance of External Process Archive. The parameters for the concurrent program are:

- Load Date: blank
- Mode: All Balances
- Balance Level: Assignment

Execute the following SQL statement before the Payroll run:  
 INSERT INTO pay\_legislation\_rules (legislation\_code, rule\_type, rule\_mode) values ('US', 'RR\_SPARSE', 'Y');

To minimize "SQL\*Net more data from clients" wait events, create the file protocol.ora in \$ORACLE\_HOME/network/admin directory of the concurrent manager server and add the following line in the file (protocol.ora):  
 tcp.nodelay=yes

## BENCHMARK ENVIRONMENT

### HARDWARE CONFIGURATION

A Unisys® ES7000/one® Enterprise Server was used as a batch/database and application server. It was equipped with the following:

- 4 × 3.0 GHz Dual-Core Intel® Xeon™ 7041 processors, each with 2 × 2 Megabytes of Level 2 Cache. HyperThreading was enabled.
- 32 Gigabytes of Memory
- 2 internal 72 GB SAS disks, mirrored
- 2 × Emulex LP10000DC-M2 dual-port Fibre Channel Host Bus Adapters
- 1,385 Gigabytes of total Disk Space (17 \* 73 GB + 2 \* 72 GB) (~80 GB used)
- 1 SAN-Connected EMC CLARiiON CX700

### SOFTWARE VERSIONS

Oracle's E-Business Suite (eBS Kit) 11.5.10

Oracle10g™ 10.2.0.2 (32-bit)

Red Hat® Linux™ RHEL 4 Update 3



#### Oracle

#### Applications Performance & Benchmarks

500 Oracle Lane

Redwood Shores, California 94065

Tel 650/506-0658

Fax 650/506-0658

Email [info@oracle.com](mailto:info@oracle.com)

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

E-Business Suite, AppsNet, Collaboration Suite, Oracle Direct and RAC are trademarks of Oracle, Inc. Oracle, JD Edwards and PeopleSoft are registered trademarks of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners. The information contained herein is subject to change without notice. Copyright © 2006 Oracle, Inc. All rights reserved. C/N 0613-1106

©2006 Unisys, Inc. All rights reserved. Unisys, the Unisys logo, and ES7000 are trademarks or registered trademarks of Unisys, Inc. in the United States and other countries.