Ascential™ Software

Enterprise Data Integration Benchmark
Part II Summary

Enterprise Data Integration Benchmark Part II Summary

This document describes the details of Ascential Software's current and ongoing efforts to provide factual and useful metrics for measuring the performance of data integration solutions.

Table of Contents

Ascential's Objectives for the Benchmark

The primary objective for this benchmark is to provide our customers with additional accurate metrics for measuring the performance of their data integration infrastructure. What performance can be expected from their current or soon-to-be purchased environment? Complex data integration requires resources from the hardware platform, the software, the database, and even the development team. Where are the bottlenecks? What levels of scalability can be expected if additional CPUs or nodes are added to the system? How difficult will it be to build a fully parallel implementation?

Answering these questions is simpler when variables concerning speed and throughput are more predictable.

This benchmark performed in September 2002 is the 2nd in a series as part of an ongoing effort to measure the "real world" performance characteristics of complex data integration solutions. It is a repeat of the first enterprise date integration benchmark performed in the same IBM Waltham lab early in February 2002 using an earlier version of the DataStage software. The specific aim of this benchmark was to compare the performance of the new DataStage 6.0 release which was made GA (Generally Available) in September 2002 to the results of the previous DataStage 5.1 release used in the previous benchmark.

It is our intent to provide you with the most up-to-date results on the performance of your infrastructure hence this 2nd benchmark in the series. Additional benchmarks to further help customers with accurate metrics are already in progress.

These benchmarks not only offer our customers an objective view of the performance of the Ascential toolkit, but also serve as an internal audit for Ascential Software engineers, who are always striving to maximize the throughput of our technology. It is our intent to continually expand the scope of this benchmark. New platforms, new product versions, more complex transformations, and new data sources and targets are just a few of the facets being added. We look forward to your feedback and additional challenges for this benchmark, and hope that it assists you in your resource planning.

Outline of Tests that were performed

The Enterprise Data Integration Benchmark consists of the following:

- Full, terabyte loads of sequential files into a relational database, with extensive transformations. These tests were conducted using 32-way parallelism,
- All tests were performed using generally available, production products. DataStage XE 6.0 with Parallel Extender manages the parallel execution of transforms and provides a graphical interface for the layout and specification of transformations and column mapping. This ensures that parallel mechanics are complemented by end-to-end meta data mapping and easy to maintain re-use of graphically based components.

Product as Tested

All benchmark tests described in this document use the new DataStage XE 6.0 Parallel Extender option, in conjunction with DataStage XE 6.0, a proven platform for data integration.

What is DataStage XE Parallel Extender?
DataStage XE Parallel Extender is an add-on option to DataStage XE that extends DataStage XE's scalability and enables companies to solve large-scale data integration problems. By leveraging the capabilities of multiprocessor hardware platforms, DataStage XE Parallel Extender satisfies the demands of ever growing data volumes and smaller processing windows. DataStage XE Parallel Extender achieves these results by providing an increase in performance equal to the number of processors. This is done by wrapping data integration components in a management framework and distributing the processing load to all available processors in the computing environment. For example, it would allow a given job to run twice as fast on two processors in the morning; 32 times faster on 32 processors in the afternoon; and 128 times faster on all 128 processors at night—without any change to the application.

DataStage XE Parallel Extender achieves the results in this benchmark because it shortens time processing requirements and linearly increases the rate of throughput when integrating massive amounts of data.

Applications that are running up against shrinking batch windows are particularly represented by this benchmark, as DataStage XE Parallel Extender redistributes existing processing so that loads are able to fit comfortably in allotted time periods.

In addition, DataStage XE Parallel Extender boosts developer productivity by eliminating the need to code new applications to run in parallel—a costly process that often requires the expertise of specialists.

Development is done using sequential data flow logic while the deployment configuration automatically adds the desired degree of parallelism.

DataStage XE Parallel Extender leverages the familiar DataStage XE user interface paradigm with a new Parallel canvas for developing data integration jobs. This capitalizes on DataStage’s connectivity and meta data functionality while enabling parallel data extraction, transformation, and loading.

The Benchmark Environment

Hardware Platform

Model: IBM P-690
OS: AIX 5.1.0.0_AIX_ML
CPU: 32 * RS-64 IV processors 1.1GHz (Silicon-on-Insulator), 8 CPUs per MCM (Multi-Chip Module), 5.6MB L2 cache per MCM, 128MB L3 cache per MCM
Memory: 96GB
Disk: 64 * 18GB SSA Drives = 1152GB / 1.125TB
(Used for Data Source Files – 1TB,OS, DB2 and DataStage XE with Parallel Extender)

Storage

Storage: IBM Enterprise Storage Server (Shark), type 2105 model F20
Connection: 2 fiber channel adapters
80 * 36.4GB SSA Drives = 2912GB or 2.84TB
(Used for DB2 Tablespace)

Source Data

Ascential Software selected mainframe data as the most appropriate "real world" model for a data integration benchmark.
Complex data types and file structures are not unique to the mainframe, but a significantly large percentage of the world's data still resides there.
Each record in the source file is made up of a set of fixed columns. The file contains representative samples of all the commonly occurring, business application data types.
File type: Fixed sequential (flat)
Columns: 97
Data types: Each of the following, in various byte lengths and significant digits
20 Packed (COBOL COMP-3)
30 Binary Integer (COBOL COMP)
4 Extended Packed (non-standard, but occurring on some platforms -17 thru 20 bytes long)
Record size: 534 bytes*
Number of records: 2059010688 rows (2 billion 59 million 10 thousand 688)
Volume: 1 Terabyte
Columns: See target table definition below for list of column names and data types.
*Actual record length, on disk. Fully expanded numeric values, after conversion are larger. Fully loaded relational table is nearly 1.2 terabytes.

Intelligent Partitioning

Critical for this benchmark was the loading of a full terabyte of complex data. The benchmark simulated a real world environment where such a quantity of data would typically be sourced from multiple locations, partitioned as to reduce
I/O and maintain the greatest efficiency when loading to the target (and also partitioned) relational database.
The flat file described above is partitioned as follows:
• 192 individual files generated for the benchmark
• "n" filesets, depending on the number of CPUs being utilized. Each file logically belongs to what is called a file set. This is a key feature of DataStage XE Parallel Extender that provides flexible management of the partitions.

Definition of the Target

The benchmark procedure loads the target using DataStage XE Parallel Extender's bulk load operator for DB2. This operator optimizes the load by using the partitioning method established within DB2 for the chosen table.
Database: DB2 EEE V7.2
Partitions: 32 (available and configured when table space initially created)
See Appendix A for an example of a syntax table.

Transformations

Data Type Conversions
Some of the most intensive work for a data integration tool involves data conversion. It's not always possible to pass this work off to the file transfer utility, the source system, or the target load facility. The data integration tool must be able to complete this activity itself.

DataStage XE Parallel Extender provides for automatic data conversions based on the meta data for each column definition. For this benchmark, binary integers and decimal (also called packed) columns are converted into their ASCII equivalents. This generally results in some degree of expansion, as the character, or ASCII representation of a numeric value often takes up more bytes than its binary counterpart.
For example, for the value +123456:

- As a binary integer (COBOL PIC S9(5) COMP for example) in most COBOL compilers, this value can be stored in four bytes, but will expand to a full six in its character representation.
- As a packed field (COBOL PIC S9(6) COMP-3 for example), this value is also stored in (minimally) four bytes, but will expand to a full six in its character repre sentation. The columns in the benchmark range in size (in bytes) and significance (number of digits), using both 2 and 4 byte integers, and packed fields in a range of 1 to 8 bytes in length.

Built-in Function Transformations

Specific columnar transformations were also performed, using the built-in functions and expressions available within the graphical environment of DataStage XE Parallel Extender. These transformations reflect transformations typical for real world data integration solutions. In total, 16 additional transformations are performed in the benchmark:

- 3 trims (removal of leading/trailing blanks)
- 1 date conversion
- 3 sub strings (extraction of a portion of a character string field)
- 3 concatenations (bringing together multiple strings into one column)
- 1 column replaced with another
- 2 new columns generated, and added to the output row
- 1 integer division
- 1 integer multiplication
- 1 addition

Lookup Transformations

Due to the fact that the lookups performed in the first benchmark added only a 2.5% increase in run time it was decided not to include them in this second benchmark. However the details of the lookups and the results from the first data integration benchmark are included.

The lookup transformations were designed to measure the impact of performing consecutive keyed lookups for each of the 2 billion+ rows. The intent was to establish a set of keys, with return values, so that approximately 80% of 2 billion+ lookups would produce a result. The other 20% would not find a value, and thus return nulls.
The lookup itself is based on a "benchmark" customer number. This customer number is compared to a detailed reference table containing address and other demographics. During the creation of the original source file, the benchmark customer number was generated randomly, providing values between 1 and 500,000. Loading of the reference table was limited to 400,000 rows, with customer number values between 1 and 400,000. This provides a hit ratio of approximately 80% when the lookups are actually performed at run time. As each of the 2 billion source rows are processed by the lookup transformation, the customer number in the current row is compared to the index in the reference table. If a matching value is found, the address details are returned and used to complete the target row. If no match is found, a null value is returned, tested, and replaced with blanks. As noted above, selected columns in the target row are filled with nulls if there is no match.

DataStage XE Parallel Extender has the ability to create and load dynamic hash structures. Hash structures dramatically increase the performance of lookups. Key validation, or return of values are the most common reasons for performing a lookup. Essentially, a lookup transformation is a type of join operation. Most importantly, as is done in this benchmark, these hash structures can be pre-loaded into memory. DataStage XE also supports in-

memory update of these structures, when necessary. Hash structures perform significantly faster than the equivalent SQL to repeatedly retrieve selected rows from a relational system.
80% Successful Lookup Rate
2 billion+ rows
500K unique key values
400K uniquely keyed rows
Using the in-memory hash structures as described, the lookup transformations increased overall run time by only 2.5%.

Benchmark Results

1) Straight loading of a full terabyte of data, with 32-way partitioning
These results are for the pure load of the files described above, including all of the data type and function transformations (without lookups).
Elapsed time: 2 hours, 52 minutes
Rates:
357gigabytes/hour:
11424 megabytes/CPU/hour:

Conclusion

Industry analysts are predicting that data volumes in the next few years will nearly double on an annual basis. Ascential Software recognizes this issue, and is heavily investing in technology to provide maximum throughput along with enhanced maintenance and seamless meta data integration for all tools used in the enterprise. Ascential Software has been working for over a year on parallel technology and strongly believes that proper use of the environment requires technology that takes advantage of hardware investments while ensuring well-managed development and deployment.

Tools must be aware of all the parallel and performance capabilities of other resources in the system (storage devices and configurations, high-end parallel rdbms', for example) but still provide automated parallelism so that developers can focus their attention on business problems and not the architecture.

The benchmark numbers in this report represent real world examples of transformation and loading of complex data.

While no two configurations are exactly identical, we believe that the variables of this test support effective comparisons that can be used to calculate and predict performance. When applied to a data integration environment, these metrics will assist in determining future thresholds, and support better allocation of resources. Future bottlenecks can be confronted, and then resolved on paper, before they force unexpected detours and constraints on production applications.

Ascential Software is committed to assisting our customers in resource planning, and enhancing this benchmark to cover all the facets of data integration. Platforms, source and target diversity, and types of transformations are just a few of the areas that will be exercised as Ascential moves forward with this standard. In addition, Ascential Software, as an active member of the Transaction Processing Council (TPC), is helping to define new benchmark specifications and standards for the global IT industry. We look forward to your comments and suggestions on this benchmark, as well as your requests for exploration of new data integration challenges.

Appendix A: Target Table Definition

```
CREATE TABLE TEST_BENCHMARK ( \
key CHAR(10) NOT NULL, \
rand INTEGER NOT NULL, \
LookUpKey INTEGER NOT NULL, \
CustName CHAR(15) NOT NULL, \
CustCode CHAR(15) NOT NULL, \
CustBalance INTEGER NOT NULL, \
CustLimit INTEGER NOT NULL, \
CustDOB CHAR(10) NOT NULL, \
CustAge INTEGER NOT NULL, \
CustAddress CHAR(10) NOT NULL, \
CustStreet CHAR(20) NOT NULL, \
CustState CHAR(15) NOT NULL, \
CustZip CHAR(10) NOT NULL, \
CustCountry CHAR(10) NOT NULL, \
SocialSecurityNo CHAR(9) NOT NULL, \
CustStartDate CHAR(10) NOT NULL, \
CustRenewDate CHAR(10) NOT NULL, \
Char01 CHAR(2) NOT NULL, \
Char02 CHAR(2) NOT NULL, \
Char03 CHAR(2) NOT NULL, \
Char04 CHAR(2) NOT NULL, \
Char05 CHAR(2) NOT NULL, \
Char06 CHAR(2) NOT NULL, \
Char07 CHAR(2) NOT NULL, \
Cha08 CHAR(2) NOT NULL, \
Char09 CHAR(2) NOT NULL, \
Char10 CHAR(2) NOT NULL, \
Char11 CHAR(4) NOT NULL, \
Char12 CHAR(4) NOT NULL, \
Char13 CHAR(4) NOT NULL, \
Char14 CHAR(4) NOT NULL, \
Char15 CHAR(4) NOT NULL, \
Char16 CHAR(4) NOT NULL, \
Char17 CHAR(4) NOT NULL, \
Char18 CHAR(4) NOT NULL, \
Char19 CHAR(4) NOT NULL, \
Char20 CHAR(4) NOT NULL, \
Char21 CHAR(6) NOT NULL, \
```

```
Char22 CHAR(6) NOT NULL, \
Char23 CHAR(6) NOT NULL, \
Char24 CHAR(6) NOT NULL, \
Char25 CHAR(6) NOT NULL, \
Char26 CHAR(6) NOT NULL, \
Char27 CHAR(6) NOT NULL, \
Char28 CHAR(6) NOT NULL, \
Char29New CHAR(6) NOT NULL, \
Char30New CHAR(3) NOT NULL, \
Char31New CHAR(3) NOT NULL, \
Integer01 INTEGER NOT NULL, \
Integer02 INTEGER NOT NULL, \
Integer03 INTEGER NOT NULL, \
Integer04 INTEGER NOT NULL, \
Integer05 INTEGER NOT NULL, \
Integer06 INTEGER NOT NULL, \
Integer07 INTEGER NOT NULL, \
Integer08 INTEGER NOT NULL, \
Integer09 INTEGER NOT NULL, \
Integer10 INTEGER NOT NULL, \
Integer11 INTEGER NOT NULL, \
Integer12 INTEGER NOT NULL, \
Integer13 INTEGER NOT NULL, \
Integer14 INTEGER NOT NULL, \
Integer15 INTEGER NOT NULL, \
Integer16 INTEGER NOT NULL, \
Integer17 INTEGER NOT NULL, \
Integer18 INTEGER NOT NULL, \
Integer19 INTEGER NOT NULL, \
Integer20 INTEGER NOT NULL, \
Integer21 INTEGER NOT NULL, \
Integer22 INTEGER NOT NULL, \
Integer23 INTEGER NOT NULL, \
Integer24 INTEGER NOT NULL, \
Integer25 INTEGER NOT NULL, \
Integer26 INTEGER NOT NULL, \
Integer27 INTEGER NOT NULL, \
Integer28 INTEGER NOT NULL, \
Integer29 INTEGER NOT NULL, \
```

```
Integer30 INTEGER NOT NULL, \
Packed01 DECIMAL(1) NOT NULL, \
Packed02 DECIMAL(3) NOT NULL, \
Packed03 DECIMAL(5) NOT NULL, \
Packed04 DECIMAL(7) NOT NULL, \
Packed05 DECIMAL(9) NOT NULL, \
Packed06 DECIMAL(11) NOT NULL, \
Packed07 DECIMAL(13) NOT NULL, \
Packed08 DECIMAL(15) NOT NULL, \
Packed09 DECIMAL(1) NOT NULL, \
Packed10 DECIMAL(3) NOT NULL, \
Packed11 DECIMAL(5) NOT NULL, \
Packed12 DECIMAL(7) NOT NULL, \
Packed13 DECIMAL(9) NOT NULL, \
Packed14 DECIMAL(11) NOT NULL, \
Packed15 DECIMAL(13) NOT NULL, \
Packed16 DECIMAL(15) NOT NULL, \
UserDefined17C CHAR(35) NOT NULL, \
UserDefined18C CHAR(37) NOT NULL, \
UserDefined19C CHAR(39) NOT NULL, \
UserDefined20C CHAR(41) NOT NULL \
)
```

About Ascential Software

Ascential Software Corporation offers the industry's only scalable enterprise integration solution that spans the entire data life cycle: data profiling, data quality management and cleansing, end-to-end meta data management and data extraction, transformation and loading. Ascential's Enterprise Integration Suite enables businesses to achieve the fastest time-to-value for their strategic enterprise applications, to have the highest confidence in the data they use to drive revenues and profits, and to continuously realize more value and ROI from their data assets and IT systems. Headquartered in Westboro, Mass., Ascential has offices worldwide and supports more than 2,100 customers in such industries as telecommunications, insurance, financial services, healthcare/life sciences, media/entertainment, manufacturing and retail. For more information on Ascential Software or a list of our international offices, visit our website at: www.ascentialsoftware.com.

Profit from Intelligent Information™

50 Washington Street
Westboro, MA 01581
Toll Free: 800.966.9875, Option 2
Tel. 508.366.3888
www.ascentialsoftware.com

WP-3020-1002

Printed in USA 10/02. All information is as of October 2002 and is subject to change.