



# White Paper

Equivio v2.3.5  
Performance Benchmark

January 2008

## Table of Contents

Management Summary.....	2
Introduction.....	2
Test Overview.....	2
Summary of Findings.....	2
Test Scenarios.....	4
Equivio Application Overview.....	4
Benchmark Cases.....	5
Test Scenarios and Results.....	6
Scenario 1: EquiLevels.....	6
Scenario 2: Databases.....	8
Scenario 3: Number of Machines.....	9
Scenario 4: Processing Threads.....	10
Scenario 5: Local vs. Remote Data.....	11
Conclusion.....	12
Appendix A – Platform Description.....	13

# Management Summary

## Introduction

This document describes the results of performance benchmark tests performed on the Equivio Version 2.3.5 software in January 2008. The tests were carried out in Equivio's R&D laboratories. The tests were conducted using standard hardware and software equipment to ensure replicability in customer production environments.

## Test Overview

The benchmark tests were conducted to demonstrate the performance of Equivio v2.3.5 in detecting and grouping near-duplicate documents and email threads in a variety of common operational scenarios. Each test was run on one or more cases, each of which comprised a collection of documents from an actual enterprise e-discovery scenario. The scenarios were designed to examine the individual effect on performance of each of the following parameters:

- EquiLevel: threshold of similarity for determining a near-duplicate
- Databases: Microsoft SQL Server and MySQL
- Number of machines: single machine versus distributed configurations
- Threads: number of processing threads on single machine
- Data configuration: local versus remote storage of input data

## Summary of Findings

The benchmark tests clearly demonstrate the ability of Equivio v2.3.5 to process large sets of real-world data within a matter of hours. The test results indicate that the vast majority of cases, namely cases consisting of up to one million text documents, can be processed by Equivio on one machine in just a few hours.

Following are the main conclusions drawn from the test results:

- Recommended configuration is based on one machine for database and one machine for processing
- For cases over 5 million documents it is recommended to use more than one machine for processing
- Equivio can handle large cases with minimal degradation
- Beyond 2 million documents, Equivio processing rate stabilizes at 100K documents per hour
- The database type (SQL Server, MySQL) has a minimal effect on performance
- High EquiLevel thresholds (i.e., in the range of 95%) are generally not effective,

with the exception of very specific data scenarios

- At EquiLevel settings in the range of 60%, the application detects significantly more near-duplicates than at 75%. Due to the relatively low impact on performance, it is recommended to use EquiLevels in the 60% range, except in extraordinarily time-sensitive scenarios
- Speed of data transfer from the disk/network is a key prerequisite for optimal performance

# Test Scenarios

## Equivio Application Overview

Equivio offers patent-pending software to detect and group near-duplicate documents and emails. The Equivio product is used to expedite the management and review of unstructured document repositories. The grouping of near-duplicates and email threads allows similar documents to be handled and treated together. The result: Equivio users slash the time and cost of document review, while ensuring the consistent treatment of similar documents.

Equivio supports a broad range of business applications. Equivio is offered as a specialized component for integration within third-party applications for ediscovery, internal investigations, document retention, email archiving and intelligence.

Equivio>NearDuplicates detects and groups near-duplicate files. It exposes hidden near-duplicates and organizes them into sets, allowing each set to be assigned to a reviewer for efficient and coherent handling. For each set, Equivio>NearDuplicates determines the pivot document, which is the most representative document of the set. In the review process, the user reads the pivot document, then reviews its near-duplicates by invoking a compare tool which highlights the differences in each document vis-à-vis the pivot document. This dramatically reduces the time required to review documents.

Equivio>EmailThreads captures and structures email threads, significantly reducing the number of emails that need to be read in a review process. Beginning with an unstructured collection of emails, Equivio>EmailThreads groups emails belonging to a thread, and builds the thread hierarchy based on the original email and subsequent "events", such as reply and forward. Equivio>EmailThreads analyzes the email content and verifies that the last email in the thread contains all preceding emails, allowing users to focus review efforts on this "inclusive" email. This eliminates dependence on metadata, which is often corrupt and cannot guarantee that the last email in a thread is inclusive.

## **Benchmark Cases**

In order to provide a test environment that reflected a real world implementation of Equivio, the benchmark tests used data from five different cases, most of which are available in the public domain. These cases, which are described below, vary in terms of volume and file types, covering a wide spectrum of operational scenarios. Each of the tests was run on one or more case.

### **TREC (Full)**

The Text Retrieval Conference (TREC), co-sponsored by the National Institute of Standards and Technology (NIST) and U.S. Department of Defense, supports research within the information retrieval community by providing the infrastructure necessary for large-scale evaluation of text retrieval methodologies. More information about TREC can be found [here](#).

The case used for the Equivio benchmark consists of documents that were made public during various legal cases involving US tobacco companies and contain a wide variety of document genres typical of large enterprise environments.

This case comprised 6.3 million files, including both documents and emails.

### **TREC (Partial)**

This dataset is a subset of the TREC (Full) case, comprising 500,000 files.

The case includes both documents and emails.

### **Enron**

This case consists of information made public as part of the Enron trial in 2004. It contains data from about 150 users, mostly senior management of Enron, organized into folders. More information about the Enron case can be found [here](#).

The dataset contains a total of about 500,000 email messages.

### **OCR**

This dataset contains some 93,000 documents that were scanned using OCR technology. The contents of this dataset are proprietary and hence not available in the public domain.

### **NSF**

This case consists of a collection of abstracts describing awards for basic research granted by the National Science Foundation (NSF) between 1990-2003. This data was also made available as part of the Text REtrieval Conference (TREC). More information about the NSF dataset can be found [here](#).

The dataset includes about 134,000 documents (no emails).

## Test Scenarios and Results

In line with standard e-discovery practice, the test scenarios assume that the input to Equivo is extracted text files, provided by the client.

### Scenario 1: EquiLevels

#### Goal

*To measure the effect of EquiLevel settings on performance.*

The EquiLevel is the minimum percentage resemblance between two documents for them to be considered near-duplicates. Any two documents exceeding this threshold are considered near-duplicates.

#### Test Data and Setup

The setup for this test scenario is presented in the following table:

Configuration	Database	# of Threads	Application	Data Location
1 machine for processing and 1 machine for database	SQL Server	4 – NSF, Enron 8 – TREC (Full)	Near-duplicates	Local*

\*Data installed "locally" on the processing machine.

Details regarding the hardware configuration can be found in Appendix A.

This test scenario was run on the following cases: NSF, Enron, TREC (Full)

#### Test Results

The test results are presented for each case. The time units relate to hours, minutes and seconds. The speed is measured in thousands of files per hour (K f/h).

#### Case: NSF - 134,616 documents

Equilevel	Time	Speed (K f/h)	% of NDs	% of Unique
60	00:18:13	444.38	16.3	83.8
75	00:17:26	463.31	11.5	89.3
90	00:13:59	577.61	0	100

### Case: Enron - 517,431 documents

Equilevel	Time	Speed (K f/h)	% of NDs	% of Unique
60	01:44:18	297.66	84.1	15.9
75	01:17:48	399.05	79.6	20.4
90	00:53:15	583.02	32.7	67.3

### Case: TREC (Full) – 6,306,680 documents

Equilevel	Time	Speed (K f/h)	% of NDs	% of Unique
60	82:20:00	76.60	26.7	67.1
75	54:58:23	114.73	16.7	76.9

**Conclusion:** Reducing the EquiLevel from 75 to 60 yields a greater number of near-duplicates with a relatively minor impact on performance. While raising the threshold to 95 significantly improves performance, the resultant reduction in the number of near-duplicates minimizes the overall value of the Equivio application (i.e., many more documents to review).

## Scenario 2: Databases

### Goal

To measure the effect of different databases on performance.

The databases tested were:

- Microsoft SQL Server 2005 - 9.00.1399.06 (X64) on Windows NT 5.2 (Build 3790: Service Pack 2)
- MySQL 5.0.45-community-nt(InnoDB)

### Test Data and Setup

The setup for this test scenario is presented in the following table:

Configuration	# of Threads	Application	Equilevel	Data Location
1 machine for processing and 1 machine for database	4 – TREC (Partial) 8 – TREC (Full)	Near-Duplicates + Email Threads	75	Local

Details regarding the hardware configuration can be found in Appendix A.

This test scenario was run on the TREC (Partial) and TREC (Full) cases.

### Test Results

The test results are presented below.

#### Case: TREC (Partial) – 552,465 documents

Database	Time	Speed (K f/h)	% of NDs	% of Unique
SQL Server	01:25:55	385.81	11.6	88.3
MySQL	01:18:54	420.13	11.6	88.3

#### Case: TREC (Full) – 6,306,680 documents

Database	Time	Speed (K f/h)	% of NDs	% of Unique
SQL Server	54:58:23	114.73	16.7	76.9
MySQL	57:04:54	110.4	16.8	76.8

\* The slight differences in results for “% of NDs” and “% of unique” are due to the multithread processing.

**Conclusion:** The type of database has a minimal impact on the performance of the Equivio application.

## Scenario 3: Number of Machines

### Goal

To measure the effect of adding machines on performance.

The tests were performed on the following configurations:

- 1 machine for processing and 1 machine for database (1 + DB)
- 2 machines for processing and 1 machine for database with replicated data (2 + DB, replicated data)

Data replication means that both processing machines have a local copy of the entire dataset.

### Test Data and Setup

The setup for this test scenario is presented in the following table:

Database	# of Threads	Application	Equilevel	Data Location
SQL Server	8	Near-Duplicates	75	Local

Details regarding the hardware configuration can be found in Appendix A. This test scenario was run on the TREC (Full) cases.

### Test Results

The test results are presented below. The time units relate to hours, minutes and seconds.

#### Case: TREC (Full) – 6,306,680 documents

# of machines	Time	Speed (K f/h)	% of NDs	% of Unique
1 + DB	54:58:23	114.73	16.7	76.9
2+ DB (replicated data)	41:02:34	153.66	16.8	76.8

\* The slight differences in results for “% of NDs” and “% of unique” are due to the multithread processing.

**Conclusion:** Adding a second processing machine improves performance by about 25% and is recommended when handling very large cases exceeding 5 million documents. To achieve maximum performance benefits, the input data should be replicated to each processing machine.

## Scenario 4: Processing Threads

### Goal

*To measure the effect of the number of processing threads on performance.*

The tests were run using 1, 4, and 8 processing threads.

### Test Data and Setup

The setup for this test scenario is presented in the following table:

Configuration	Database	Application	Equilevel	Data Location
1 machine for processing and 1 machine for database	SQL Server	Near-Duplicates + Email Threads	75	Local

Details regarding the hardware configuration can be found in Appendix A.

This test scenario was run on the Enron case.

### Test Results

The test results are presented below. The time units relate to hours, minutes and seconds.

#### Case: Enron - 517,431 documents

# of machines	Time	Speed (K f/h)
1	04:35:02	112.88
4	02:39:47	194.3
8	01:46:59	290.19

**Conclusion:** Adding processing threads enhances performance, assuming that the machine has sufficient RAM memory. At the same time, the benchmark shows that using 8 rather 1 processing threads resulted in a very slight decrease, in the range of 0.5%, in the number of near-duplicates detected.

## Scenario 5: Local vs. Remote Data

### Goal

*To measure the effect of data access on performance.*

This test examines the impact of the proximity of the data collection to the processing module. In the first run the test data is stored locally, on the processing machine, while in the second run the data is stored remotely.

### Test Data and Setup

The setup for this test scenario is presented in the following table:

Configuration	Database	# of Threads	Application	Equilevel
1 machine for processing and 1 machine for database	SQL Server	8	Near-Duplicates + Email Threads	50

Details regarding the hardware configuration can be found in Appendix A. This test scenario was run on the OCR case.

### Test Results

The test results are presented below. The time units relate to hours, minutes and seconds.

#### Case: OCR – 93,000 documents

Local/Remote	Time	Speed (K f/h)	% of NDs	% of Unique
Local	00:17:10	325.05	35.2	65.1
Remote	00:24:58	223.50	35.3	65.2

\* The slight differences in results for “% of NDs” and “% of unique” are due to the multithread processing.

**Conclusion:** Storing the input data locally increases performance by slightly more than 30%. For best performance, it is important to ensure optimal access to the data. Similarly, when using a distributed configuration with multiple processing machines, it is recommended to replicate the data on each of the processing machines.

## Conclusion

The benchmark tests conducted on the Equivio v2.3.5 software examined how variances in several key parameters affect the performance of the application. The tests were run using five different sets of real-world data and were designed to reflect common operational scenarios.

The results of these tests clearly demonstrate the ability of the application to process cases containing up to one million documents in timeframes of up to a few hours. In terms of performance, the optimal configuration is to use separate machines (1+1) for database and for processing. For very large cases, i.e., containing over 5 million documents, it is recommended to use more than one machine for processing.

Standard PCs with 2 GB of memory can be used for efficient Equivio processing. In order to eliminate potential processing "bottlenecks," it is recommended to use a robust machine with 3-4 GB memory (RAM) to host the database. System performance can be enhanced by use of multiple threads, the availability of the input data on the processing machine, and a distributed configuration.

## Appendix A - Platform Description

This appendix provides details regarding the hardware configurations used in the performance benchmark tests.

The Equivio v2.3.5 application runs on standard hardware and software platforms commonly deployed in enterprise environments.

### Processing Machine 1

CPU	AMD Athlon 64 x2 Dual Core Processor 3800+ 2.0 GHz
Memory (RAM)	2 GB
Hard Disk	Western Digital WD5000KS Size : 465 GB
Operating System	Windows XP SP2
Network	Realtek RTL8169/8110 Family Gigabit Ethernet NIC
	Microsoft .Net Framework 2.0

### Processing Machine 2

CPU	AMD Athlon 64 X2 Dual Core Processor 6000+ 3.0 GHz
Memory (RAM)	2 GB
Hard Disk	Western Digital WD5000AAKS-22YGA0 Size : 465 GB
Operating System	Windows XP SP2
Network	Realtek RTL8169/8110 Family Gigabit Ethernet NIC
	Microsoft .Net Framework 2.0

### Database Machine

CPU	AMD Athlon 64 X2 Dual Core Processor 5600+ 2.81 GHz
Memory (RAM)	4 GB
Hard Disk	Western Digital WD1500AHFD—00RAR5 Size : 149 GB
Operating System	Microsoft XP Professional x64 Edition
Network	NVIDIA nForce Networking Controller
	Microsoft .Net Framework 2.0