

ITP

ITP/Server Performance

2006-08-09

This paper discusses the results of a number of ITP/Server performance tests



INTELLIGENT TEXT PROCESSING

Table of contents

Table of contents	1
1. Introduction	2
2. A Real-life test case.....	3
3. The hardware	4
4. The production process	5
5. The Software	6
6. Overall Performance.....	7
6.1 Dealing with Graphics	8
6.2 Output Data Rate	8
7. Scalability	9
8. Tuning document production	10
8.1 Optimize Graphical Elements.....	10
8.2 Evaluate Alternative Tools and Processes	10
8.3 Set up Suitable Hardware	10

1. Introduction

This paper discusses the results of a number of ITP/Server speed performance tests that have been performed in January 2006, during two days at the IBM Innovation Centre in Amsterdam, the Netherlands. The objectives of this paper are twofold:

- On the one hand, to serve as an aid in establishing whether ITP/Server will be able to meet the performance requirements of a particular case on particular hardware.
- On the other hand, as a guideline to tune of optimal performance. In particular, it identifies a number of pitfalls and advantages of choosing one technique over the other.

In order to obtain reliable results, and to avoid overlooking potential issues, we performed a large number of tests. We have avoided presenting all the performance figures here in great detail. Instead, we address a number of highlights and illustrate these with the corresponding figures.

2. A Real-life test case

In order to obtain useful results, we aimed at producing documents that closely resemble real-life cases. We tried to avoid over-simplification, but we also avoided making the test more complex than one would expect nowadays.

The test involved policy cancellation letters. These letters were designed to match typical real-life documents. Each letter was 2 or 3 pages long, depending on the number of policies cancelled, and each page contained a company logo.

The mail merge data was provided as a set of structured XML files containing policy records, each incorporating a customer record and multiple repeating records with policy information for each letter.

The final output consisted of PostScript files for printing and PDF files for archiving. A *single* PDF file was produced for *each* letter. The PostScript files contained batches of at most 1,000 letters each, and were enhanced with OMR codes.

3. The hardware

We had access to two machines at the IBM Innovation Centre.

- An [IBM eServer x365](#) system equipped with 4 Intel Xeon 2.8 GHz hyper threading processors.
- An [IBM eServer x460](#) system equipped with 4 Intel Xeon 3.0 GHz dual-core processors.

Apart from the different processors, the main difference between these machines was that the x365 was equipped with a RAID storage system (6x 36GB 15K RPM SCSI Ultra 320, Serveraid 6M SCSI controller), whereas the x460 "just" had 4 separate 10K RPM Ultra 320 SCSI disks.

4. The production process

The mail merge data consisted of XML files with data for up to 1,000 letters each. These were offered to ITP/Server via its Directory Watch interface.

For each XML file, ITP/Server performed the following steps:

- Execute an ITP model to produce a Word document containing the letters specified in the XML file.
- Print the Word document to a PostScript spool file.
- Convert the PostScript spool file to a PDF file.
- Enhance the PostScript file with OMR codes.
- Split the PDF file into separate PDF files each containing a single letter for archival.

5. The Software

Apart from ITP/Server 3.2, we have used a few additional tools:

- PostScript Printer: we used the Adobe Universal Driver Installer (1.0.6) to install an IBM InfoPrint 2085 PS driver.
- For the PostScript to PDF conversion we used GhostScript 8, and Adobe Acrobat Distiller 6.
- We added OMR codes with an OMR tool of our own.
- The PDF files were split with a commercially available command line tool (see <http://www.pdf-tools.com>).

6. Overall Performance

We will start with answering the main question: *how many documents can ITP/Server produce in a given period of time on this system?*

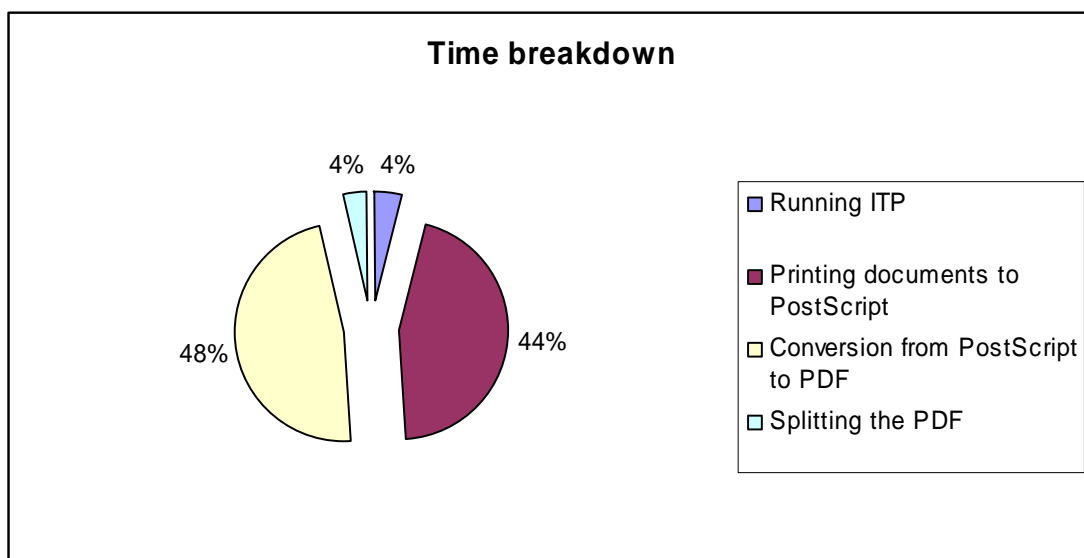
The first results below have been obtained on the x460. We configured the system to run one Document Processor per virtual CPU (totaling 8 Document Processors) and we submitted 250 XML files with 100 records each. We stored the input XML files, the temporary documents and the output on three separate disks in order to improve disk performance.

We included a WMF logo in the ITP model, and produced this logo in both the PostScript and the PDF output. GhostScript was used to convert PostScript to PDF. It took 8 minutes and 45 seconds to produce 25,000 letters, each having 2.5 pages on average. Extrapolating yields the following performance figures:

	Produced per hour	Produced per day
Number of letters	169,000	4.0 million
Number of pages	421,000	10.1 million

A breakdown of the steps in the document production process shows how time is spent:

Step	Time (for 8 DPs)
Running ITP	3:03
Printing documents to PostScript	31:42
Conversion from PostScript to PDF	33:59
Splitting the PDF	2:32



This breakdown shows that ITP spends only a small amount of time producing the documents compared to the time needed to generate the PostScript and PDF files.

On the sample system ITP alone would produce documents at the following rate:

ITP/Server Performance

	Produced per hour	Produced per day
Number of letters	3.9 million	94.4 million
Number of pages	9.8 million	236.0 million

6.1 Dealing with Graphics

It turned out that printing and PDF conversion spend a fair amount of time handling graphics, and in our case, handling logos. This offers room for improvement in case the letters are printed on preprinted paper. In that case, the logos only need to be added to the PDF output, and this can be done relatively efficiently *after* the printing/conversion process.

Testing on the x365, with 2 pages per letter showed the following results:

Number of Pages	Produced per hour	Produced per day
Logo in Word Document	424,000	10.2 million
PDF Logo Overlay	576,000	13.8 million
No logos	600,000	14.4 million

6.2 Output Data Rate

The table below shows the file sizes that got generated for each letter, as well as the total output data rate, i.e. the output size per document times the number of documents produced per hour. Note that we are not counting storage intermediate files here.

The PostScript file size is relatively little compared to the PDF size, because each PDF files contains only a single letter, and thus it contains much more overhead than each PostScript file, which stores several hundreds of letters.

	PostScript Size per Document	PDF Size per Document	Output Data Rate
Logo in Word Document	54 KB	91 KB	29 GB/h (8 MB/s)
PDF Logo Overlay	18.4 KB	87 KB	29 GB/h (8 MB/s)
No logos	18.4 KB	82 KB	29 GB/h (8 MB/s)

One thing that is remarkable is that this rate appears to be a constant. It is tempting to conclude that this can be used to predict performance based on measurements on the output size per document. Validating such a claim however, would require more testing.

7. Scalability

Best performance was obtained by configuring the same number of Document Processors as (virtual) CPU's were available. Adding more Document Processors did not help: it even turned out to be slightly counter-productive.

In our tests, we observed that each processor easily reached 95-100% CPU utilization. However, having 8 virtual CPU's, we did not reach a speed-up of 8. Instead, maximum speed-up was about a factor of 5. Given the high processor utilization, there may have been some low level resource contention that does not easily show up in the processor utilization graphs.

On the other hand, as I/O contention does affect the CPU utilization, the x460 does not seem to be hampered by the fact that it did not have a RAID configuration. It appears that an "ordinary" configuration with a number of high-speed disks is fast enough. An output rate of "only" 8 MB/s supports this view, despite the fact that total I/O requirements will be higher. Document production is rather a CPU-bound process than an I/O bound process.

Note also that the document production batches are completely independent. As a result, it will also perform well on a processor farm, at the cost of a slightly more complex infrastructure.

8. Tuning document production

There are several factors to take into account when optimizing the throughput of a document production system:

8.1 Optimize Graphical Elements

Graphics, such as logos, may impact the overall performance of the document production process considerably. Not only does it result in larger files, but it may also introduce expensive conversions and scaling during printing and PDF generation. Therefore:

- Introduce logos as late as possible. Use pre-printed letterhead paper and add logos to the PDF files as overlays. This avoids overheads during earlier stages.
- Include logos in a vector-based format. Use formats such as WMF or EPS. Consider converting Mono-color logos and signatures into a TrueType font. These formats get handled more efficiently during PostScript/PDF processing.
- If bitmapped logos cannot be avoided, use logos which are already saved in the correct resolution for the printer. This avoids expensive scaling operations.
- Choose graphics formats which can be handled efficiently by the other tools in the document production process.

8.2 Evaluate Alternative Tools and Processes

- Different tools have different performance characteristics, and no tool is best in all situations. This means that it is wise to evaluate the performance of alternative tools for the problem at hand. GhostScript for example has more trouble with GIF images than Adobe Acrobat Distiller. In one test, Adobe Acrobat Distiller only needed 1 minute for a job that took GhostScript 8 minutes to complete. In all other situations, we observed that Adobe Acrobat Distiller is somewhat slower than GhostScript.
- Consider different processing paths. If you do not require PostScript output, consider converting the Word document to PDF directly using the DocToPDF command of ITP/Server. Even if you do require PostScript output there may be cases where it is more efficient to convert a Word document to PDF directly than to use the PostScript output as the source for PDF conversion.

8.3 Set up Suitable Hardware

Make sure the server has abundant resources available to process large amounts of data. Any bottlenecks will affect performance.

CPU

- Use dedicated systems for document production.
- Configure one Document Processors per (virtual) CPU in the system and keep all Document Processors busy. Even though performance will not scale linearly with the number of Document Processors it will still improve the amount of time required to process a batch.
- Configuring more Document Processors than there are (virtual) CPUs in the system does not improve performance and can even be counter-productive. Each Document Processor should be capable to use run at 95-100% CPU utilization if there are no other bottlenecks.
Note: This is only true for ITP/Server batch processes and not for ITP/OnLine Server interactive document production. In the latter case, it is the amount of user interaction

ITP/Server Performance


with ITP/Server that plays an important role in the decision on how many Document Processors should be used.

Memory

- Make sure the server has enough memory available. The document production process generates large amounts of intermediate data as well as result documents. Free memory can be used by Windows for caching that data and prevents delays waiting for disks to store/retrieve data.
Statistics for memory usage depends on too many factors to provide exact figures. As a reference, our test configuration used at any time 1G for operating system and programs and 1GB for cache out of 8GB available memory.

Disks

- Split storage over multiple disks. Use RAID storage and/or use different disks for input, temporary storage and output.
- Plan for bandwidth. Disks and controllers must be able to keep up with the amount of data that is being produced.

ITP is developed by 

For more information please contact us.

Telephone: +31 24 371 02 30
Fax: +31 24 371 02 31
WWW: <http://www.aia-itp.com>
Email: itp@aia-itp.com
Postal Address: P.O. Box 38025
6503 AA Nijmegen
The Netherlands
Visiting Address: Kerkenbos 10-129
6546 BJ Nijmegen
The Netherlands