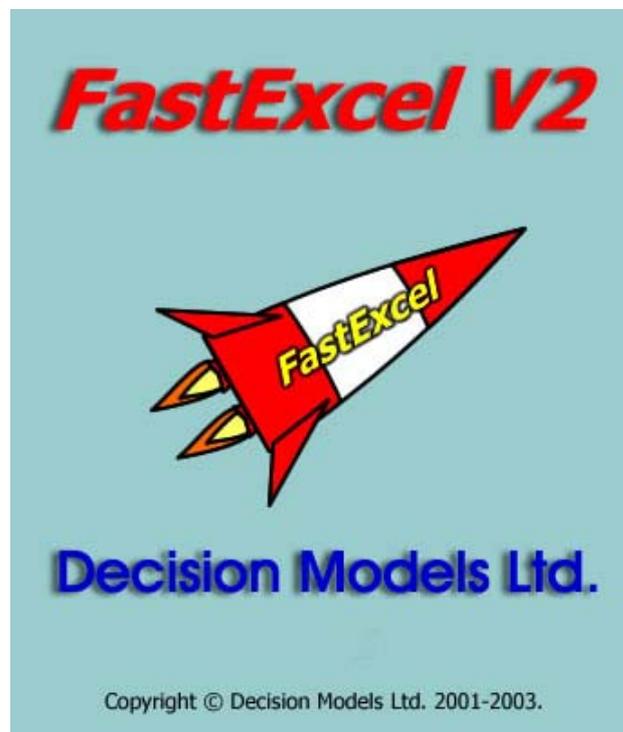

FastExcel Version 2

Sample Problem Guide

By Decision Models Ltd



Copyright

Copyright © 2004 Decision Models Limited. All rights reserved.

The information contained herein is supplied without representation or warranty of any kind, is subject to change without notice and does not represent a commitment on the part of Decision Models Limited.

Decision Models Limited, therefore, assumes no responsibility and shall have no liability, consequential or otherwise, of any kind arising from this material or any part thereof, or any supplementary materials subsequently issued by Decision Models Limited.

Decision Models Limited has made every effort to ensure the accuracy of this material. If you have any questions or comments, contact:

Support@DecisionModels.com

Document Version

Sample Problem Version Number 2.444 dated 12 September 2004.

Trademarks

Microsoft®, Windows®, Windows NT®, PivotTable®, Visual Basic® are registered trademarks of Microsoft Corporation. Other product and company names mentioned herein may be the trademarks of their respective owners.

Intoduction

This document is designed to guide you through the FastExcel sample problem workbook FXLSampV2.xls.

The workbook contains the results of a FastExcel analysis of three versions of the same spreadsheet: **Before, After and NoNamesAfter**.

If you like a challenge, look at the **Before** workbook and work out where the calculation bottlenecks are and what needs doing to reduce the calculation time from 22 seconds to 0.03 seconds. Or just look at the FastExcel Analysis results, and read this explanation.

The Before workbook takes 22 seconds to recalculate at 1200MHZ.

The After workbook takes 0.17 seconds to recalculate.

The NoNamesAfter workbook takes under 0.03 seconds to recalculate

All workbooks produce the same results, but the After and NoNamesAfter workbooks have been performance optimised using FastExcel.

The difference between After and NoNamesAfter is that After makes extensive use of dynamic range names, whilst NoNamesAfter has had the names removed. This also makes it much easier to see how some of the performance optimisation has been done.

Workbook Design Criteria

The sample workbooks are loosely based on the customer-driven market segmentation model built for BT Cellnet and RS Consulting.

The major design criteria for the Before and After workbooks were:

- Build the model to allow more data to be added without extensive formula modification: this required extensive use of dynamic range names.
- Main results to be shown in a single worksheet (Answers) showing the spread across levels and takeup as a percent of universe.
- Enable additional results to be shown in Pivot Tables: this requirement drives the design of the Products sheet as a single table containing the detail results.

The FastExcel analysis was done on a system with a 1200MHZ AMD Athlon and 768MB RAM using Excel 2000 SP3 and Windows XP.

Note: for a more detailed explanation of FastExcel output refer to the user manual, which is available online, and is also downloadable in PDF format, on the DecisionModels website <http://www.DecisionModels.com>

The FXLSampV2 Workbook

The Worksheets

The FXLSampV2 Workbook contains the following worksheets.

Explanation:

Text explaining the output worksheets in FXLSampV2.xls

B4BookProf

The output of FastExcel V2 Workbook Profile on the Before workbook.

B4OptSeq

The output of FastExcel V2 Optimise Worksheet Sequence on the Before workbook.

Step2Prof

The output of FastExcel V2 Workbook Profile on the Before workbook, after Clean Workbook and Optimise Worksheet Sequence.

Step3Prof

The output of FastExcel V2 Workbook Profile on the Before workbook, after eliminating the worksheet cross-reference loops.

Step3Answers

The output of FastExcel V2 Profile Worksheet Areas on the Answers worksheet in the Before workbook

Step4Products

The output of FastExcel V2 Profile Worksheet Areas on the Products worksheet in the Before workbook.

AfterBookProf

The output of FastExcel V2 Workbook Profile on the After workbook.

AfterProducts

The output of FastExcel V2 Profile Worksheet Areas on the Products worksheet in the After workbook.

AfterAnswers

The output of FastExcel V2 Profile Worksheet Areas on the Answers worksheet in the After workbook.

NoNamesAfter

The output of FastExcel V2 Workbook Profile on the NoNamesAfter workbook.

B4Names

B4Names contains a listing of the named ranges in the Before workbook, produced by FastExcel Name Manager. The names referring to the major worksheets are of the form Worksheetname.Rangename. When rangename refers to a column the name starts with "col".

AfterNames

AfterNames contains a listing of the named ranges in the After workbook, produced by FastExcel Name Manager.

Comparison of the B4Names and the AfterNames shows a number of the formula optimisations.

FastExcel Analysis of the Before workbook.

Analysis Step 1:

We start by analysing the Before Workbook using the FastExcel Profile Workbook Command:

Worksheet B4BookProf:

The output from Profile Workbook: major performance bottleneck areas highlighted in orange are:

- **%Waste** (used range wastage) on **Answers, Products, Respondents, Product Data and Respondent Data**: this wastes memory and can cause calculation time problems.
- **Over 5 seconds calculation time on Answers**, coupled with very high microsecond per formula time.
- **High workbook volatility (100%)**: this means that Excel's smart recalculation engine tells Excel that it has to calculate the vast majority of the formulae each time. This is probably caused by a large number of volatile functions in the workbook.
- **Book overhead** of 15 seconds.
- The high microseconds per formula on Counts can be ignored because there are very few of these formulae and hence the sheet calculation time is low.

Optimisation Step 1: Clean & Sequence

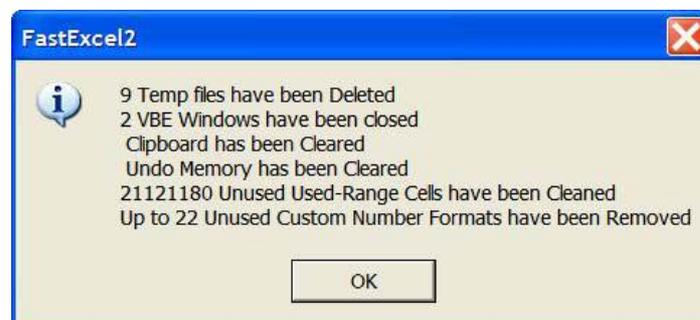
The largest bottlenecks are the 15 second workbook overhead and the high %Waste, so lets see what FastExcel **Clean Workbook** and **Optimise Worksheet Sequence** can do:

Worksheet B4OptSeq:

The output from the FastExcel Optimise Worksheet Sequence command, showing a potential reduction in Forward Worksheet crossreferences from 39,000 to 48. Using this optimised sequence will largely eliminate the 15 second workbook overhead. There are two worksheet cross-reference loops which will cause some residual workbook overhead, so we will run Workbook Profile again to see if they are worth eliminating.

Clean Workbook:

the output from the FastExcel Clean Workbook command gets rid of over 21 million wasted used-range cells.



Analysis Step 2: Residual book overhead

Worksheet Step2Prof

Shows the workbook profile after Clean Workbook and Optimise Worksheet Sequence.

The wasted usedrange has been eliminated and the recalculation time has been reduced by over 17 seconds, but there is a **residual book overhead of 1.8 seconds** caused by the circular worksheet crossreference paths.

Optimisation Step 2: Remove Circular CrossRefs

If we inspect the Universe sheet in the Before workbook we find formulae in I13:J25 with references to Respondents. Moving this block of formulae to the Respondents sheet breaks the loop and removes the workbook overhead.

Analysis Step 3: the Answers Sheet

Worksheet Step3Prof:

Shows that Recalculation time is now down to 2.2 seconds, and 96% of that is taken by the 2 sheets **Answers** and **Products**.

Answers has a high calculation time per formula, but Products has a low calculation time per formula.

Next we want to know why the Answers sheet takes 2.1 seconds for less than 2500 formulae: We will use the **Profile Worksheet Areas** command to analyse the worksheet.

Worksheet Step3Answers:

The output from Profile Worksheet Areas on the Answers sheet. This shows that a subset of the columns have very high calculation times.

Optimisation Step 3: the Answers Sheet

Looking at these columns on the Answers sheet in the Before workbook you can see that they contain array formulae calculating a sum with 2 conditions (product and sector):

```
{=SUM(( $A10=Products.colProductName) * ($B10=Products.colSectorName) * (Products.colL2Users) ) }
```

The FastExcel built-in Optimise Calculations help has sections on array formulae and multiple condition SUMIF with suggestions on how to improve performance:

The array formulae bottlenecks on **Answers** were eliminated by reducing the size of the range required to only the rows for a single product, using the steps below.

- Sorting the Products sheet by Product, which is the first condition in the multiple conditional sum.
- Calculating the start row and number of rows for each Product block of rows on Products, and storing these on the Counts sheet.
- Using these counts in an OFFSET function to return dynamically calculated ranges containing only the rows for a single product.

This significantly reduces the number of rows used in the conditional sum, and allows us to use SUMIF rather than array formulae.

The OFFSET formulae are held in dynamic named ranges, for instance Products.ProductNameRange contains

```
= OFFSET (A9_Products!$D$1,  
VLOOKUP (Answers.Product,ProductIndex,3, FALSE)-1,0,  
VLOOKUP (Answers.Product,ProductIndex,4, FALSE),1)
```

Also look at the After or NoNamesAfter.xls workbook to see how the array formula has been changed to a SUMIF on a much smaller range. (Its easier to see whats happening in the NoNamesAfter workbook).

Analysis Step 4: the Products worksheet

The next priority is the 0.2 seconds taken by **Products**, so let's see what **Profile Worksheet Areas** tells us.

Worksheet Step4Products:

The output from Profile Worksheet Areas on the Products Sheet. This shows that 6 (D:I) columns out of 39 take over 80% of the calculation time. Looking at the formulae in these columns you can see that they contain double exact Match Lookup formulae which trap missing data such as:

```
IF (ISNA (MATCH (ProductData.ProductSerial, RespondentData.colRespondentSerial, 0)), "Mismatch", INDEX (RespondentData, MATCH (ProductData.ProductSerial, RespondentData.colRespondentSerial, 0), 7))
```

Optimisation Step 4: the Products worksheet

The FastExcel built-in Optimise Calculations help contains a number of suggestions that can be used to reduce this time significantly:

These Exact match and Lookup bottlenecks on Products were eliminated by:

- Sorting both Products and ProductData in the same sequence, and using approximate match.
- Adding an additional column, RespondentIndex, containing the result of an approximate Match between Respondent serial and Product serial (or Mismatch if no exact match could be found), and using this as a rowindex in subsequent lookups.
- Optimising the dynamic range formulae by storing the result of COUNTA for the major counts of respondents, products, sectors and segments, and reusing these in the formulae.

Also look at the detailed implementation in the After or NoNamesAfter workbooks.

Analysis Step 5:

Worksheet AfterBookProf:

The output from Profile Workbook shows a performance improvement of 21.6 seconds or a factor of about 130.

- Book overhead has been reduced from 15 seconds to 0 seconds.
- Used range wastage has been removed by removing the excess formatting etc.: memory used has decreased by 1 MB and filesize has decreased by 1611KB.
- Answers sheet calculation time and microseconds per formula time have improved dramatically.
- Products sheet calculation time radically improved.

Workbook volatility is still high. This is mainly caused by the extensive use of dynamic named range formulae to allow additional data to be added to the productdata and companydata sheets without needing to change any formulae. These dynamic named range formulae contain OFFSET, which is a volatile function. Eliminating these dynamic ranges would remove most of the 0.17 seconds of volatility, and further reduce calculation time.

Worksheets AfterProducts and AfterAnswers:

The output from Profile Worksheet Areas on the Products and Answers sheets. The major calculation bottlenecks have been removed.

The After workbook overall calculation time is just under 0.17 of a second, which is a little slow for automatic mode. In practice this workbook will not require frequent data changes, so this is probably acceptable.

However it is interesting to see how much further optimisation can reduce calculation time... so The NoNamesAfter workbook has been further optimised by removing the dynamic named ranges (at the expense of requiring added work if new data is added).

Optimisation Step 5: NoNamesAfter

The dynamic ranges and defined names have been removed, and all the formulae now use normal Excel references rather than references to names.

To make it easier to understand the new formulae replacing the array formulae on the answers sheet, the start row and number of rows are referenced on each row of the sheet.

Analysis Step 6: NoNamesAfter

Worksheet NoNamesAfter:

The output from Profile Workbook on the NoNamesAfter workbook.

- Book overhead is now eliminated, in fact the workbook calculates faster than the sum of the sheets.
- Volatility is reduced by about 0.135 seconds: the recalculation time is virtually instantaneous at under 0.03 of a second.
- Memory useage (but not workbook size) has been reduced by over 2.5MB

This workbook can now easily be used in automatic calculation mode.

Summary of Changes made in the After workbook to optimise performance:

Note that because one of the key design criteria of these demonstration workbooks was to use dynamic named ranges the resulting formulae optimisations in this workbook are done using range names. Sheet AfterNames contains a listing of the revised range names which you can compare to the B4Names sheet.

- The used range wastage problems on ProductData, RespondentData and Answers were eliminated by using the FastExcel Clean Workbook command.
- The names of the worksheets were altered using FastExcel's optimised sequence from B4OptSeq.
- The forward worksheet crossreference loops were eliminated by moving a small block of formula from the Universe sheet to the respondents sheet.
- The array formulae bottlenecks on Answers, were eliminated by reducing the size of the range required to only the rows for a single product, using the steps below.
 1. Sorting the products sheet by product, which is the first condition in the multiple conditional sum.
 2. Calculating the start row and number of rows for each product block of rows on products, and storing these on the counts sheet.
 3. Using these counts in an OFFSET function to return dynamically calculated ranges containing only the rows for a single product.

This significantly reduces the number of rows used in the conditional sum, and allows us to use SUMIF rather than array formulae.

The OFFSET formulae are held in dynamic named ranges, for instance Products.ProductNameRange contains

```
=OFFSET(A9_Products!$D$1,  
VLOOKUP(Answers.Product,ProductIndex,3,FALSE)-1,0,  
VLOOKUP(Answers.Product,ProductIndex,4,FALSE),1)
```

The exact match and lookup bottlenecks on Products were eliminated by:

- Sorting both Products and ProductData in the same sequence, and using approximate match.
- Adding an additional column, RespondentIndex, containing the result of an approximate match between respondent serial and product serial (or "mismatch" if no exact match could be found), and using this as a rowindex in subsequent lookups.
- Optimising the dynamic range formulae by storing the result of COUNTA for the major counts of respondents, products, sectors and segments, and reusing these in the formulae.

Changes in the NoNamesAfter workbook.

The defined names have been removed, and all the formulae now use normal Excel references rather than references to names.

To make it easier to understand the new formulae replacing the array formulae on the answers sheet, the start row and number of rows are referenced on each row of the sheet.