Introduction

Are your spreadsheets important? This book is for you! It describes good practices that all spreadsheet users should follow.

Use this book to learn how to avoid the most common errors and to make future development easier. If you are a software tester or a manager of end-user-computing, it gives you techniques for checking spreadsheets for accuracy and soundness. If you are an auditor looking for evidence of fraud, such as deliberately concealed data or functionality, you will also benefit from knowing the many ways in which data can be hidden or calculation methods subverted.

To derive the fullest benefit from this book, you should already have spreadsheet skills corresponding to the European Computer Driving Licence (ECDL) Module 4 (M4) or Advanced Module 4 (AM4). The focus of this book is on spreadsheet construction using formulas. All the examples are in Microsoft Excel. They have been tested in Excel versions 97, 2000, 2002(XP), and 2003. We mention a few differences between Excel 97 and Excel 2002/2003.

Why is this book necessary?

This book was created in response to expressions of concerns by managers about risk to the business from a pervasive dependence on spreadsheets. Barry Boehm, in his article Software Defect Reduction Top 10 List' [1] warned: 'The ranks of *sorcerer's apprentice* user-programmers will swell rapidly, giving many who have little training or expertise in how to avoid or detect high-risk defects tremendous power to create high-risk defects'. He proposed that to help reduce the likelihood of user errors, developers should provide the equivalent of seat belts and air bags in cars, along with safe-driving aids and rules of the road. This fits in nicely with the 'Driving Licence' analogy!

---

[1] www.cebase.org/www/News/top-10-defects.html Boehm, B. and Basili, V. R. (2001, January). "Software Defect Reduction Top 10," Computer, 135-137.

Regulators are examining spreadsheet controls

The UK HM Revenue and Customs department audit the tax returns for Value-Added Tax (VAT) submitted to them in spreadsheet form. The largest VAT error found in a spreadsheet in 2001 was around £1M sterling (about €1.5M, or $1.9M). Recovering that amount gave them a very good payback on their investment in developing spreadsheet auditing software and skills.

The US Food and Drug Administration (FDA) have rigorous specifications for the integrity of electronic records. These include spreadsheets used as laboratory log books.

The US Sarbanes-Oxley Act requires auditors to comment on internal controls over financial reporting. As consolidation and reporting is frequently done by spreadsheets, management are spending more time on review and testing.

The European Spreadsheet Risk Interest Group

The European Spreadsheet Risks Interest Group (Eusprig) publish research on spreadsheet errors on their website `www.eusprig.org`. Their annual conference attracts speakers from all over the world to present on tools and techniques for the detection, correction, and prevention of spreadsheet errors.

Researchers from ISACA (Northern UK Chapter), University of Wales Institute Cardiff and the University of Greenwich met in 1999 to discuss the ever-increasing problem of business risk associated with spreadsheet errors. As a result of this meeting they founded Eusprig. This group aims to increase awareness of spreadsheet risk among academia and industry. It also promotes research into the extent and nature of the problem, methods of preventing and detecting of errors and methods of limiting damage. This has brought together researchers and professionals in the areas of business, software engineering and auditing to seek effective solutions. It now includes interested parties from HM Revenue and Customs, PricewaterhouseCoopers, KPMG, the British Computer Society (BCS), the Dutch Computer Society (NGI) and the Netherlands Bank, as well as many other companies and individuals.

The ECDL Foundation

The European Computer Driving Licence (ECDL) foundation in Dublin, Ireland, develops the ECDL syllabus for the certification of users in computer competence. They found demand from companies for certification, beyond technical knowledge,

to include a level of self-awareness and responsibility for data integrity and calculation correctness. Accordingly, in 2003 they met with experts in the European Spreadsheet Risk Interest Group (Eusprig) to capture current best practices in design quality and the avoidance of errors. They presented a paper at the fifth Eusprig conference[2] in Klagenfurt, Austria, in 2004. In it, they outlined their goals for a future syllabus:

> [This] Module requires the candidate to appreciate their responsibility for good spreadsheet design and use given the scope and significance of spreadsheets use in the contemporary workplace. The candidate shall appreciate the need for good spreadsheets specification, be aware of some of the key security considerations, and appreciate the benefits of clearly organised, well presented and easy-to-use spreadsheets. The candidate shall be able to construct sound spreadsheets, find and correct common errors, and test for input and output accuracy. It is anticipated that the candidate will already have mastered the skills and achieved the knowledge detailed in ECDL Module 4, Spreadsheets, or ECDL Module AM4, Spreadsheets, Advanced-Level.

---

[2] http://www.isys.uni-klu.ac.at/ISYS/eusprig04/

The structure of this book

Each section is organised as follows:

> The key practice item appears in a heavy box.

The item is followed by some *definitions* of any special terms that are used.

This is followed by a description of the essential skills needed, and examples.

> Stories with web links in a shaded box give reported cases of expensive mistakes that have been made by not following good practice.

✍ Each section ends with a 'Check your knowledge' self-test marked with a *writing hand* symbol in the left margin. The answers are in the back of the book. Most exercises use sample files that you download from http://www.sysmod.com/sbp. You may be asked for a username and password taken from a page of this book.

Typographical conventions

Normal text appears in a normal typeface.

Worksheet names and special terms are shown *in italics*.

Spreadsheet file names are shown in **boldface**.

`Code samples and formulas appear in monospace text.`

Microsoft Excel Menu commands and dialog text appear in sans serif text.

The > symbol means you continue to a submenu or dialog tab, eg <u>F</u>ile > <u>O</u>pen. The underlined letters indicate where shortcut keys may be used.

Excel function names are in uppercase, eg ROUND, SUM.

We sometimes advise you to search the Help file contents for further detail. We provide a key phrase to type into the *Answer Wizard* or the *Ask a Question* box:
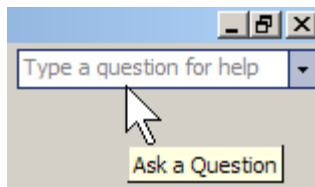
CATEGORY: 1 Design

This category comprises two skill sets:

> 1.1 Specification
>
> 1.2 Security

This section will give you the knowledge to answer these searching questions that your employer or auditor may ask:

- How do you know this spreadsheet is what is wanted?

- Has the design been reviewed and approved by the system owner?

- What change controls do you have in place?

- How often do you backup your key spreadsheets and data?

- How many previous versions can you go back to, on what dates?

- How often do you verify your backups, are they secure and off-site?

- What traceability is there on information exchanged between corporate data sources and spreadsheets for consolidation and reporting?

- Have you measured the sensitivity of your financial models to changes in input parameters?

Organisations in regulated environments may have to comply with standards relating to traceability, Standard Operating Procedures (SOP), validation testing and verification of calculation methods. Those responsible for enforcing SOPs need to explain to a spreadsheet creator how their work process and product is affected by these standards.

Skill set: 1.1 Specification

1.1.1   Define spreadsheet specifications.

A *Specification* is a statement of *how* the spreadsheet is to meet *the user requirements*. A specification can be as simple as explanatory text in a worksheet, or as complex as separate legal documentation. There is an example specification for a payroll spreadsheet on page 10.

The *purpose* of the spreadsheet is a statement of why it was created, the problem it solves, and the part it plays in the decisions and documentation of the business process.

The list of *user requirements* is the most important part of a specification. Clear requirements provide the basis for acceptance testing, and say what the user does and does not require, preventing disagreement as to when the work is complete.

Usability features are help text, navigation aids, protection, and automation of common tasks. Specify these so that the intended users can operate the spreadsheet correctly, avoid misunderstandings, and achieve their objectives easily.

The *author*, the initial creator of the spreadsheet, is often also the person responsible for maintaining it.

The *version history* describes the stages of evolution of the spreadsheet.

Such information may be recorded in:

1. the spreadsheet File Properties;
2. documentation worksheets;
3. separate document files.

If the spreadsheet is to be sent to people who should not see internal, personal, or private information, remove this data from the released version of the spreadsheet. Do not rely on hiding it to maintain confidentiality. Section 5.1.3 on page 139 describes how cells, formulas, rows, columns, and sheets can be unhidden.

Use the spreadsheet file Properties

Use File > Properties to fill in data that makes it easier to search for files, and to identify the author when questions arise. Typically these are: Title, Subject, Author, Category, Keywords, and Comments; and where appropriate the Manager and Company. The Custom properties allow you to add more detail. Keep contact details for those responsible for development, maintenance, and supplying data.

File properties initially contain the username (the registered user of the software) as the author.

Excel can prompt you to fill in the file properties each time you save. On the Tools menu, click Options, and on the General tab check Prompt for workbook properties:
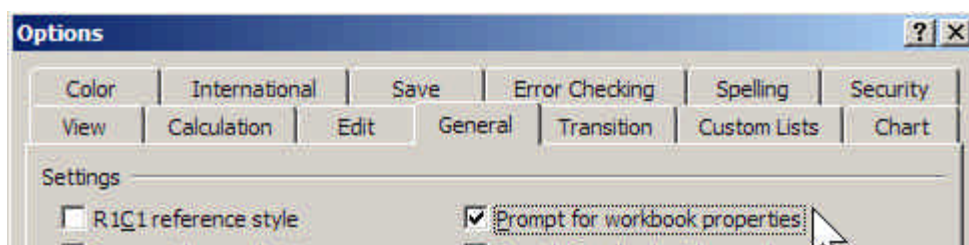
Place documentation in worksheets

Keep documentation close to where it used so that it can be updated along with what it refers to. Otherwise, neglected documentation or cell comments may become actually misleading. Section 1.1.2 on page 11 shows how to use cell comments to store information that may be needed by the user at the time of data entry.

Place in a separate *Instructions* worksheet the user instructions and explanatory notes on calculations and dependencies on external data sources. The example shown in the following screen shot is from the spreadsheet **TaxSimple.xls**, which is one of the sample workbooks accompanying this book.

22	Outputs			
23	*Periodic:*		After entering the gross pay, print off the sheet as a record for the employee.	
24				
25	**Formulas**	Column	Heading	Calculation
26		C	Cumulative Gross Pay to date	The first row is the same as the entry in col.B, subsequent rows accumulate the sum of the pay for the current period and the previous cumulative pay in col.C.

TaxCalc / TestCases / Notes / **Specification** /

When to use separate document files

Use separate documentation in cases where the documentation is extensive. This may be appropriate for user training manuals or operating procedures such as how to enter data, check for errors and correct them.

Keep a note in the workbook of where that separate document is so that it is not forgotten about.

Interviews and project notes are usually kept in word processed documents. Create a separate document of technical design notes for the maintainer. These include:

- Internal or hidden data;
- External links to other files or corporate databases;
- Data populated by program code such as live data feeds;
- Any special setup instructions for first time use;
- Passwords to access protected areas;
- Test plans (who applies what tests to the spreadsheet when and how)
- Test records (the results of executing the test plans)

How much specification is enough?

Describe assumptions and limitations, even apparently obvious ones. They may not be obvious to the user or to the next person who maintains the spreadsheet; or to you at some time in the future. You may already have had the experience of trying to make sense of someone else's large and sprawling spreadsheet. State what is intentionally omitted from the scope of a spreadsheet project, as that is the hardest thing to check later.

Exactly how detailed a specification needs to be depends on the relationship between the person who creates the spreadsheet structure and the user. If you are creating a spreadsheet for yourself, you need to record whatever you think is not obvious from looking at it. What is obvious to you, while you are immersed in work, can be easily forgotten later, outside that context.

Designing a spreadsheet for others to use

The extent of user-friendly features that you need to design in will depend on:

- The number of people who will be using the spreadsheet

- The wider the diversity of their skill level

- The lower their skill level (consider skills certification)

- The greater the turnover of staff who use it

Test the spreadsheet with different user groups in order to reduce the likelihood that a confusing or awkward layout will induce users to create errors.

- Can & do the users access the online help in text cells, cell comments, and instruction worksheets?

- Can & do the users access written documentation or co-worker assistance?

- How adequately does the help provided resolve the questions of different user groups about using the spreadsheet?

- In the usability evaluation notes, do you find user problem comments like 'slow, time-consuming, awkward, tedious, intensive, unclear, not sure, confused, bewildering, hard to navigate / follow / find, repetitious, duplication'?

✎   1.1.1 Check your knowledge (answers on page 165)

1. What are the risks in omitting a specification for creating a spreadsheet?
2. What are five key items to include in a specification and which is the most important?
3. What kind of information would you use *File Properties* for?

An example of a written specification

User requirement: calculate the tax due on employee salaries each month.

Examples of acceptance test cases are given on page 130.

The basis for calculation is as specified in the Irish Revenue Commissioners document 'PAYE Notice to Employers 2004' which states:

1) The standard rate of tax (20%) is applied to gross pay up to the standard rate cut-off point (SRCOP) for the pay period.
2) Any balance of gross pay over SRCOP is taxed at the higher rate (42%)
3) The gross tax is reduced by the tax credit to give the net tax payable.
4) The figures for tax credit and the SRCOP are defined in a Tax Deduction Card for each employee issued to the employer by the Revenue Commissioners. An example of this card is shown following:

Sample Tax Deduction Card for A.N. Employee

A	B	C	D	E	F	G	H	I
Date of payment	Gross Pay	Cumul Gross Pay to date	Cumul Std Rate Cut off Point	Cumul Tax due at Std rate	Cumul Tax due at Higher rate	Cumul Tax Credit	Cumul Tax (not less than zero)	Tax deducted this period
			1644.45			123.34		
			3288.89			246.68		
			4933.33			370.02		

The figures are calculated to full precision and displayed to 2 decimal places.

Calculation details (Source: http://www.revenue.ie/wnew/employ.pdf)

Col. A: Enter the date; Col. B Enter Gross Pay; Col. C is the cumulative of Col. B

Columns D and G are supplied by the Tax Office in the Tax Deduction Card.

Col. E: Enter the lower of column 3 or 4 multiplied by standard rate of tax

Col. F: Where Col. C minus Col. D is positive enter the result multiplied by the higher rate of tax. Where negative enter zero.

Col. H: is the sum of cols. E and F less Col. G

Col. I: Subtract the cumulative tax (H) for the last pay period from the cumulative tax (H) for this pay period and enter the result here.