Table of Contents

Introduction

Spreadsheets have become the pre-eminent application for staff dealing with numerical data in most organisations. Users with a high level of business knowledge now have access to a powerful tool that is flexible and easy to use. Business processes are sometimes adapted to fit in with spreadsheets, as a result there is no practical alternative open to a user other than to use a spreadsheet.

Simple applications with no risk are ideal for spreadsheet development, however as the level of complexity and dependency on the resulting data increases so does the risk. Spreadsheets have inherently weak controls, especially in the areas of data validation, testing, and change control. Users while knowing how to use spreadsheets, do not always know the best ways to design and test models.

The trade-off between extra time developing a spreadsheet against the assurance in quality of data has to be made in view of the risks associated with that spreadsheet. Being aware of the risks will allow an informed user to make a decision on the extent to which best practice guidelines should be incorporated into a spreadsheet.

This document aims to highlight the main risk areas, and provide best practice guidelines in order for the user to make the correct trade-off decision.

The scope of these guidelines exclude macros, as modern macros tend to be more like programming languages. This is a separate practise area, and as such contains totally different guidelines and areas of risk.

Why use spreadsheets ?

Upside		Downside
Ease of Use	v	. Easy to misuse
Flexibility	v	. Leads to lack of structure . Hard to validate data
Quick	v	. Can lead to a lack of thought and planning . Elapsed time should also include time to final solution (including testing)
Knowledge of use	v	. Have to make sure users are aware of design guidelines, as well as how to use the spreadsheet
Easy to Amend	v	. No inherent version control . No audit trail: cannot find out who did the changes, or what the changes were . Easy to make mistakes when changing (may forget assumptions/business rules initially built in to the spreadsheet)
Low cost	v	. Cost of software does not include cost of training, peer support, cost of amending spreadsheet, and cost of finding and fixing errors . Budget is effectively transferred from IS budget to 'virtual' budget i.e. the users time. This has an associated opportunity cost
Available & Timely - can be developed when wanted	v	. No downside, however it takes longer to develop a well designed spreadsheet
Confidence in application and result	v	. Even though the chance of making an error in an individual cell is low, due to the large number of cells in a spreadsheet the cumulative error rate is high. Over 90% of spreadsheets larger than 150 rows contain at least one significant formula mistake ('How to Make Spreadsheets Error Proof'- David Freeman)
More Focused - where an application is highly specific and contains a lot of pure business logic, there may be no other suitable application	v	. Limited downside, however these models tend to be complex; weigh up against the effect on business of failure
Industry standard	v	. General standard may not fit some specific requirements
Multi purpose tool	v	. Should be aware of the limitations
Easy to transfer data	v	. Reputation if data is incorrect

Upside		Downside
		. Repetition of errors . Chance of virus infection
Offers immense range of functionality that is not offered in other packages without huge effort	V	. As complexity increases, so does the chance of error as does the difficulty in changing the model . Complexity and poor design mean spreadsheet may be difficult to use
Ease in creating data and information	V	. Not necessarily easy to share with others as there are no standards e.g. not everyone uses the same format, difficulties in updating other peoples data
Data is available on a laptop, you can take it with you out of the office	V	. Raises issues regarding security, backup, and synchronisation

Areas of Risk

Spreadsheets inherently lack controls i.e. how do you control invalid data input, and audit trails i.e. who changed what, thus not only do you get problems with design, but problems are easily introduced and hard to detect (whether they are deliberate or accidental).

Spreadsheets are hard to test, as there is no enforced structure on spreadsheets, this means it is difficult to implement a structured testing program. Testing of spreadsheets is only point in time compliance, which means that the test is only valid when you tested the spreadsheet, unless you enforce rigid change control processes.

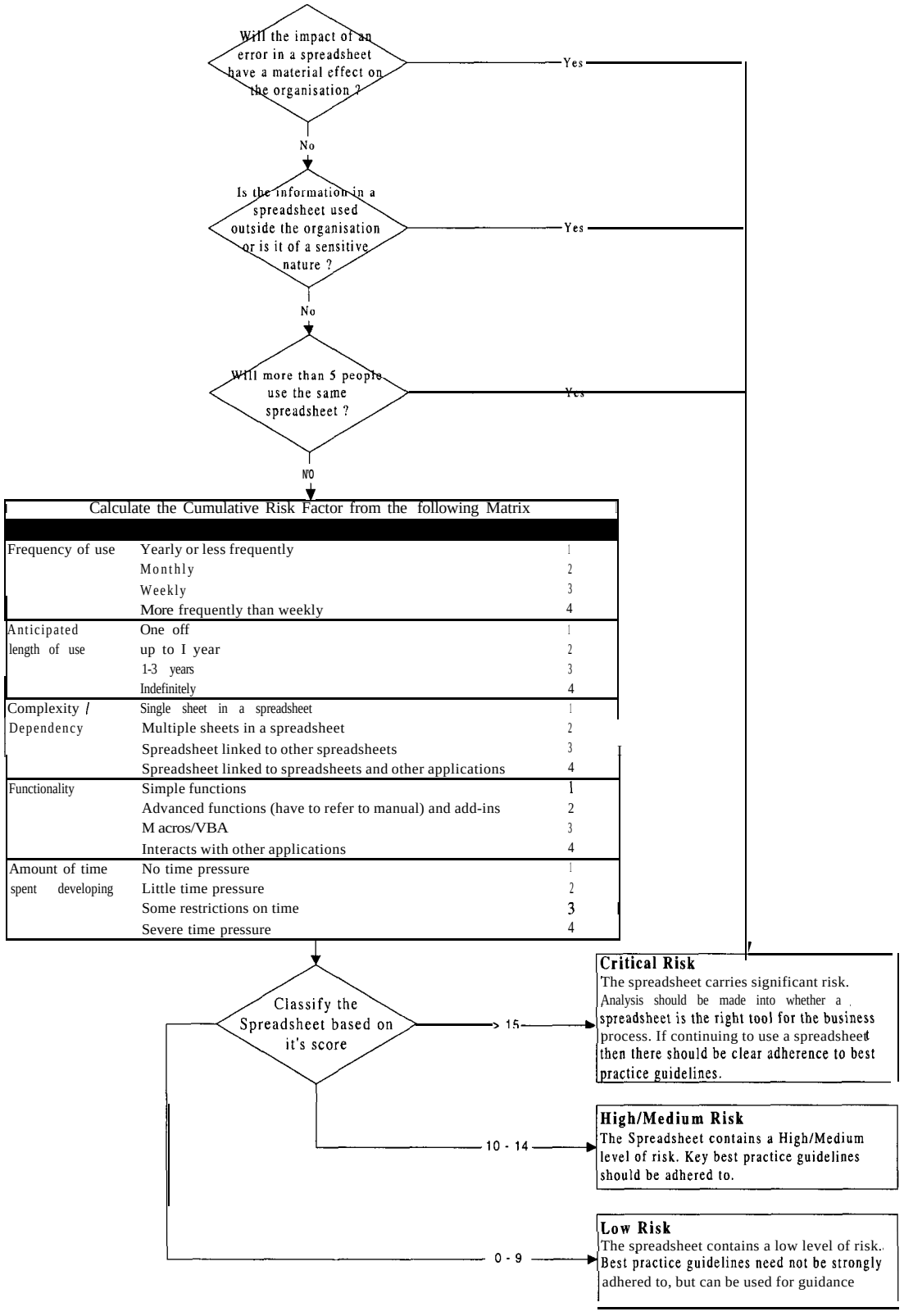
Documentation of spreadsheets is often ignored, and the assumptions and processes of the spreadsheet are usually kept with the person who built it. This means that spreadsheets are hard to maintain over time especially if the person who built the spreadsheet is no longer the user. Anyone that has had to make large changes to a spreadsheet that was created by someone else will be aware of these challenges.

Users become proficient and comfortable using spreadsheets and become reluctant to change to alternative systems, even if they are more suitable for that particular situation.

File management is also an area of risk for spreadsheets. Spreadsheets residing in a common directory have limited controls over accidental deleting, or file alteration by third parties.

Quantifying Risk

Spreadsheets can be assessed for risk by using the following decision tree/risk matrix:



Calculate the Cumulative Risk Factor from the following Matrix

Frequency of use	Yearly or less frequently	1
	Monthly	2
	Weekly	3
	More frequently than weekly	4
Anticipated length of use	One off	1
	up to I year	2
	1-3 years	3
	Indefinitely	4
Complexity / Dependency	Single sheet in a spreadsheet	1
	Multiple sheets in a spreadsheet	2
	Spreadsheet linked to other spreadsheets	3
	Spreadsheet linked to spreadsheets and other applications	4
Functionality	Simple functions	1
	Advanced functions (have to refer to manual) and add-ins	2
	M acros/VBA	3
	Interacts with other applications	4
Amount of time spent developing	No time pressure	1
	Little time pressure	2
	Some restrictions on time	3
	Severe time pressure	4

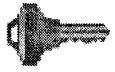
Critical Risk
The spreadsheet carries significant risk. Analysis should be made into whether a spreadsheet is the right tool for the business process. If continuing to use a spreadsheet then there should be clear adherence to best practice guidelines.

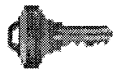
High/Medium Risk
The Spreadsheet contains a High/Medium level of risk. Key best practice guidelines should be adhered to.

Low Risk
The spreadsheet contains a low level of risk. Best practice guidelines need not be strongly adhered to, but can be used for guidance

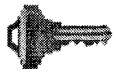
Best Practice Guidelines

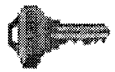
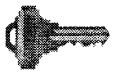
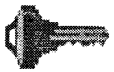
References to formulas in this document are from Excel, other spreadsheet packages should have equivalent functions, but with possibly different names.

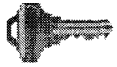
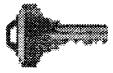
Key Business Practice	Best Practice	Benefit	Risks when deviating from this practice	Considerations
	Structure	*Guidelines on the layout and structure of a spreadsheet*		
	Spreadsheets should have a consistent layout. The functional areas of data input, workings/calculation, and output should be distinct and separate. As a general rule variables, assumptions and flexible inputs should be separate from the working area	• Gives the spreadsheet structure, easier to follow • The spreadsheet is self documenting, key variables and assumptions are clearly identified • Lends itself more clearly to 'what if analysis • Easier to check and validate data input • Can print all input/assumption data easily • It is easier to set-up protection of the working area thus reducing the risk of users making inadvertent changes • Can easily incorporate check totals for each section so you can reconcile what is input against what is output	• It will be harder for other people to use and understand your model • It will be harder to make major changes to the model • Will increase the likelihood of errors e.g. cannot check assumptions as easily if they are not grouped or displayable • Cannot check input data as easily as when it is separate • Cannot easily find or change assumptions	• Number of other spreadsheets dependent (or linked) to it • Whether or not the spreadsheet is 'Strategic' and likely to be added to, or changed at a later date
	Where possible the data input area of the spreadsheet should be in the same order as the source	• Easier to input, less risk of transposition errors	• Accuracy and reliability of data at risk	• May conflict with the point below • How much data you have to

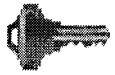
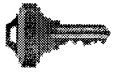
Key Business Practice	Best Practice	Benefit	Risks when deviating from this practice	Considerations
				input • How often you have to input • If it is unwieldy to print out, design a reformatted input area in the output area
	Separate the data input area into 2 sections: data you change regularly, and data you change irregularly	• Easier to input	• Accuracy and reliability of data at risk	• May conflict with the point above • How often you have to input
	Where a critical value is contained in a formula in one or more cells e.g. interest rate, then put it in a separate cell and refer to this cell in the formula	• Spreadsheet is easier to follow • Have consistency, ensures all formulas refer to same value • Easier to change one cell than a number of formula	• Accuracy and reliability of data at risk	• Good practice would mean referring to this the critical value as a named range e.g. Cl4 * Int_Rate rather than as a cell reference i.e. Cl4 * B3
	When designing large spreadsheets try not to put tables below or to the right of preceding ones. Alternatively if this is not practicable, put tables on different sheets	'Less likely to do operations to entire row or columns that inadvertently effect multiple tables e.g. changing a column width Reduces the chance of unintentional deleting	Accuracy and reliability of data at risk	• You have a good memory for how your spreadsheet is laid out • Ensure there is a backup policy in place, so your spreadsheet can be restored if necessary • Your spreadsheet has undo facilities • If you are putting each table on a different sheet this may increase the complexity of the formulas
	The data input area should not generally contain formulas	In line with the overall design, this area is for input of data only	Accuracy and reliability of data at risk	• However you should have a check total at the bottom of the

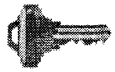
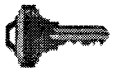
Key Business Practice	Best Practice	Benefit	Risks when deviating from this practice	Considerations
		• Minimises the chance of overwriting formula		data input to ensure integrity
	Try using colours or shading cells that contain data input	• Aids data input • Makes the spreadsheet easier to audit	• Accuracy and reliability of data at risk	• Be aware that some colours and colour combinations do not print clearly on black and white printers
	Uses	*What a spreadsheet should not be used for*		
	Spreadsheets should not be used as a database. A database is defined here as a spreadsheet with: more than 1 table, containing sensitive data, that has sophisticated reporting requirements etc.	• You do not get enforced data validation (although later versions of spreadsheets are starting to offer this) • You do not have control over access to the data i.e. create, amend, read, suppression • Hard to use more than one table on a spreadsheet • Cannot relate tables i.e. create relational databases • Hard to create complex queries • Cannot enforce business rules on data e.g. a customer must be a valid customer before they can place an order	• Accuracy and reliability of data at risk	• There is only 1 reasonably small table • Data input and validation is not critical • The information requirements are minimal
	Spreadsheets should not be used as an accounting system	• You cannot validate data properly • You cannot enforce business rules easily e.g. each debit must have a corresponding credit • It is difficult to implement security	Accuracy and reliability of data	

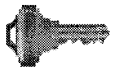
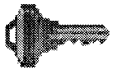
Key Business Practice	Best Practice	Benefit	Risks when deviating from this practice	Considerations
		effectively		
	Documentation	- *Guidelines on documentation. A more detailed description of documentation follows in the next section*		
	Each department should have a catalogue of spreadsheets in use. At a minimum it should all the 'critical'	. You can quantify how many spreadsheets are in use, and how much time is spent on end user development . How much time duplication there is. Whether spreadsheets could be integratet/rationalised . Identify the opportunity to create spreadsheet templates for generic development . Draw attention to the number of critical spreadsheets	. Increased development costs . Unaware of the number of spreadsheets in use . Unaware of the number of 'critical' spreadsheets in use	. This catalogue could be maintained by a call to a help desk to register for support
	use named ranges instead of cell references	. Spreadsheet becomes self documenting, e.g. SUM(salaries) is easier to understand than SUM (Bl:BlO) . Macros that refer to names rather than cell references are easier to debug . Aids in navigation as you can easily go to a named range . Makes the spreadsheet much easier to audit	. It will be harder for other people to use and understand your model	. Takes longer to set up
	Testing and Debugging	- *Most spreadsheet errors are caused by :* . *Failing to input all data* . *Data transposition errors* . *Mislabelling references (graph and spreadsheet)*		

Key Business Practice	Best Practice	Benefit	Risks when deviating from this practice	Considerations
		• *Incorrect range references* • *Inadvertent insertion/deletion of cells, rows and columns* • *Accidental overwriting offormula and data in unprotected cells* • *Leaving temporary changes or test data in the spreadsheet* • *Errors in logic* • *Errors in formula*		
	Document what tests were done and what the results were (vs the expected results)	• Provides an audit trail on the test plan • Gives assurance on the integrity of the model • Enables you to check how extensive the tests were	• Accuracy and reliability of logic	• If you take care in designing a set of test data, you may be able to use it on multiple spreadsheets
	Have the spreadsheet tested by someone other than the developer	• Ensures the spreadsheet addresses the business issue • The information in the spreadsheet is accurate and reliable • A fresh pair of eyes uncovers fresh problems	• Accuracy and reliability of logic	• The tester should be more of a business expert rather than a spreadsheet expert
	Check spreadsheet validity by printing it out and then doing the test on the printout. Print the data only and not the formulas (Spreadsheet Presentation and Error Detection: An Experimental Study Galleta et al, Journal of Management Systems)	• Reading from paper is 20-30% faster than from a computer • Reading from paper, and without formula you do not repeat the same assumptions/mistakes the spreadsheet makes You concentrate just on the data, and don't have the added mental load of interpreting the formula. It also means	• Accuracy and reliability of logic	• This is a time intensive task, will take longer than testing on the screen, but this is why it is seldom done • It is often difficult to do this due to the complexity of the formula, in this instance perhaps the formula could be simplified ?

Key Business Practice	Best Practice	Benefit	Risks when deviating from this practice	Considerations
		you concentrate on the business process rather than the algorithm		
	Formula---	*Guidelines when constructing formula*		
	Try to avoid complex formula. If its purpose isn't obvious break it into smaller components, or alternatively replace it with a user defined macro function	• Easier to understand • Easier to audit • Easier to change/edit	Accuracy and reliability of data	• Weigh this against the amount of space needed to break the formula down into its smaller parts. Perhaps do these on a separate sheet, and carry the final answer forward • May be able to use more complex functions like SUMIF, VLOOKUP, or CHOOSE • It is more difficult to audit user defined functions unless you have specialised knowledge
	When using the sum function try to ensure that the range to be summed includes a blank cell at either end (i.e. top & bottom, or left & right)	• When rows or columns are added the formula maintains it's integrity	Accuracy and reliability of data	• Likelihood of data being added to the spreadsheet • Impinges on aesthetic of compactness • You can hide blank rows/cols
	Use absolute i.e. C4 and partial absolute references i.e. C$4 in formulas where possible	This allow formulas to be more easily copied, eliminating the need to type in formulas, thus reducing the risk of transnosition error	Accuracy and reliability of data	• Implementation depends on size and structure of spreadsheet
	Round all formulas (except pure	Makes the spreadsheet more accurate	Accuracy and reliability of data	• Takes longer to implement, so

Key Business Practice	Best Practice	Benefit	Risks when deviating from this practice	Considerations
	addition and subtraction) to the same number of decimal places as displayed on the spreadsheet. This is because the number displayed on the spreadsheet is not always the same as the value in the cell. In Excel use the ROUND function.			weigh this against the degree of accuracy required from the spreadsheet • Although rounding differences are usually immaterial, they can affect the degree of confidence in the spreadsheet • Has implications with check totals
	Validating	▪ *Guidelines in validating a spreadsheet*		
	Incorporate validation checks on data input. Use functions like lF(OR(A1<5,Al>10),"out","in") to check a value is inside a range, or ISTEXT() to see if a cell contains numbers or text	• Reduces the number of errors on data input	•Accuracy and reliability of data	• Use discretion on what you test as not all data can, or should be validated e.g. you can't validate someone's name • Even though you incorporate controls, they can often be ignored by the user. However later versions of spreadsheets are incorporating stronger controls in this area
	Have a batch total to check the total of data input. Preferably input a control total, and have the spreadsheet calculate its batch total underneath	• Ensures all the data is in the spreadsheet before it is manipulated	• Accuracy and reliability of data	• If control totals are not used then to check data input you will have to check each piece of data individually
	Use formulas that foot and cross foot when summing data. This provides a	• More likely to catch errors when you have 2 checks on the data	. Accuracy and reliability of data	• Instead of having formulas in 2 cells, you can incorporate in 1

Key Business Practice	Best Practice	Benefit	Risks when deviating from this practice	Considerations
	cross check	• Will catch insertions of row or columns if you have errors in your formulas		cell using a formula similar to IF(SUM(A10:F10) <> SUM(Gl:GlO), "Cross Check Error", SUM(G1:G10))
	Large columns of data should not contain sub totals and totals. A significant number of errors are due to this practice	• Increases the reliability of the spreadsheet	• Lose their integrity when significant changes are made • Wrong range can be entered • Rows or columns added after formulas have been written leads to errors	• In Excel use functions like SUMIF instead (categorise the data in a separate column and SUMIF on this column) • You can also restructure the spreadsheet so the data is on one sheet, and the totals and subtotals are on a separate sheet
	Develop self balancing models. Having a profit and loss & a balance sheet in the same model (if you can't fit it all on one spreadsheet use links at a summary level) will mean your balance sheet will balance	• You have all relevant data in one source • Data will balance thus ensuring completeness of data input	• Accuracy and reliability of data	• Size and complexity of model • There is no self balancing model
	Don't use other peoples spreadsheets for critical applications, unless they have been independently tested by someone other than the developer	• You inherit risks from the spreadsheet	• Accuracy and reliability of data	• Spreadsheets that are similar and in common usage should be tested and set-up as spreadsheet templates.
	Change & Version Control	*- Ensure all changes to a spreadsheet are approved, and properly tested, and that documentation is updated. A change to the spreadsheet means a change to the logic of the spreadsheet rather than a change to the data..*		
	If you have set-up an input area, print it out whenever the data changes	• This acts as an audit trail of the data input you should be able to identify	• Accuracy and reliability of data	• Can't do this if you haven't set the spreadsheet up in this

Key Business Practice	Best Practice	Benefit	Risks when deviating from this practice	Considerations
	it out whenever the data changes. Ensure that either the header or footer has the time and date of printing as well as the name of the spreadsheet	input, you should be able to identify what data changed and when it changed		the spreadsheet up in this manner • If doing something such as modelling it may be impractical to print out the input data every time it changes, but instead save data for each scenario.
	Protect all formulas, and data that should not change	• Avoids accidental changes to data • Some spreadsheets take you easily from one unprotected area to another thus aiding data input	• Accuracy and reliability of data	• Harder to make changes to the model if it is protected Takes a lot longer to set-up, but will be easier to do if have set up spreadsheet in input-working-output format Often easier to build spreadsheet first then add protection
	Every time you change the logic (workings) of the spreadsheet, change the name of the spreadsheet to reflect the change e.g. BUDGETv9. Remember to keep a copy of the prior 2 versions as backup e.g. BUDGETv8 & BUDGETv7. So that you have 3 levels of backup i.e. grandfather, father, son	• You will know what version of the spreadsheet you are using, and from the documentation how it is different from other versions • Ensures you have a backup • Keeping backup in a separate directory will protect against accidental deletion	Won't be able to track changes in your spreadsheet easily	Obviously not necessary for spreadsheets that don't change much The backup process should be tested annually

Key Business Practice	Best Practice	Benefit	Risks when deviating from this practice	Considerations
	With each change ensure you retest the spreadsheet and make sure you change the documentation	. Gain assurance in the data . Helps with error detection ● Provides an audit trail	. Accuracy and reliability of data . You won't know what version of the spreadsheet you are using	. In areas of high risk document the test results . Just test the parts that have changed . It is easier to do if you have a set of test data established . Obviously just change the documentation that is relevant
	If you have spreadsheets that solve a common business problem, and are created quite frequently consider developing a spreadsheet template	. Spreadsheets will be consistent in layout and logic . Reduces the amount of time spent re-inventing the wheel	. Alot of time spent by many people solving the same problem . May not have consistent assumptions and logic in all the spreadsheets	. Make sure the template is thoroughly tested and documented

Documentation

Implementing best practice guidelines mitigates some of the need for extensive documentation, as the spreadsheet will become easier to use, understand and maintain.

Best practise guidelines state that documentation should, where possible, be included with the spreadsheet, rather than printed out and filed. Documentation is no good if it is not read, and if it is accessible it is more likely to be updated. A separate sheet within the spreadsheet should be used for general documentation e.g. description, assumptions, limitations, change log etc. Using notes within cells is a good to place to put documentation relevant to that sheet e.g. input instructions, description of formula etc.

Focus on documentation should be increased as the level of risk increases. Full documentation would include :

- Definition
  - Purpose of the spreadsheet and its structure/layout
  - Operating instructions
  - Assumptions and limitations both explicit e.g. interest rate is in cell B 1, as well as implicit e.g. cashflow is calculated on a cashflow rather than an accrual basis
  - Integrity checks and reconciliations that are built into the model
- Processes
  - A test log listing all testing done on the spreadsheet, e.g. printouts of input area, output area, what and how you tested etc.
  - A change log i.e. why it was changed, and what was changed. This should be able to be cross referenced to the version number of the spreadsheet
  - How to change the spreadsheet (document the areas that are most likely to be changed)
- What needs inputting (where you get the data from, under what assumptions, and how often) and where it needs to be input
- How data flows through the sheet (possibly use diagrammatic aids)
- All printouts should contain the spreadsheet name and date of printing
- Who developed it / uses it

Sources

Gartner Group: End User Development - When is it worth it? Questions and answers on End User Development Part 1

How to Make Spreadsheets Error Proof - David Freeman. Journal of Accountancy

Systems Auditability and Control (Module 7 End User Departmental Computing). Institute of Internal Auditors Research Foundation

Handbook of IT Auditing - Warren, Edelson & Parker

The CPA Journal Online - Standardising Spreadsheet Designs http:\www.luca.com\cpajournal

Spreadsheet Presentation and Error Detection - An Experimental Study - Galleta, Hartzel, Johnson, Joseph & Rustagi. Journal of Management Information Systems

The Spreadsheet Style Manual - David Harrison, John W Yu

Excellence in Spreadsheets. Institute of Chartered Accountants of New Zealand