



Getting what you want

Cailin takes on the task of writing a proposal

Throughout the last two years I have written a number of technical proposals for my employer. These usually concerned either the acquisition of new hardware or modifications to our current infrastructure.

Strangely enough my colleagues didn't always get the same amount of success as I did, which got me to thinking about the question "How does one write a proper proposal anyway?".

In this mini-tutorial I'll provide a rough outline of what a proposal should contain, along with a number of examples. Throughout the document you'll also find a number of Do's and Don'ts to point out common mistakes.

You will also notice that I predominantly focus on the acquisition of hardware in my examples. This is due to the nature of my line of work, but let me say that the stuff I'll be explaining applies to many other topics. You may just as well apply them to desired changes to your network, some software that you would like to use and even to some half-assed move which you want to prevent management from making.

Now I've never been a great fan of war, but Sun Tzu really knew his stuff! Even today his philosophies on war and battle tactics are still valid and are regularly applied. And not just in the military, since these days it's not uncommon to see corporate busy-bodies reading "The art of war" while commuting to work. In between my stuff you'll find quotes from Sun Tzu which I thought were applicable to the subject matter.

Zen and the art of 'getting what you want'

"Though we have heard of stupid haste in war, cleverness has never been seen associated with long delays."

Sun Tzu ~ "The art of war"

It happens occasionally that I overhear my colleagues talking about one of their proposals. Sometimes the discussion is about the question of why their idea got shot down and "what the heck was wrong with the proposal?". They had copied a proposal that had worked in the past and replaced some information with their own. When asked to show me said proposal, I'm presented with two sheets of paper, of which one is the quote from our vendor's sales department. The other consists of 30% header/footer, a short blurb on what we want to buy and a big box repeating all of the pricing info.

The problem with such a document of course is the fact that management gets its nose rubbed in the fact that we want to spend their money (and loads of it too). To them such a proposal consists of a lot of indecipherable technical mumbo-jumbo (being the quote and some technical stuff), with the rest of the document taking up dollars-Dollars-DOLLARS. While to you it may seem that the four or five lines of explanation provide enough reason to buy the new hardware, to management this will simply not do.

So... One of the things it takes to write a proper proposal is to write one which keeps your organization's upper echelons in mind. However! Don't forget about your colleagues either. It is more than logical that you run a proposition past your peer to check if they agree with all of the technicalities.

So in order to make sure that both your targets agree on your proposal you will have to:

- A) employ tech-speak to reach your peers and
- B) explain your reasoning in detail to your management.

In order to craft such a document there are a number of standard pieces to the puzzle which you can put in place. I'll go over them one by one. One thing I want you to realize though, is the fact that this will take time. Expect to spend at least half a day or even more on writing.

Pieces to the puzzle

My proposals tend to consist of a number of sections, some of which are optional as not every type of proposal requires the information contained therein. For instance, not every project will require resources which can be easily expressed in numbers and hence there is no need for a list of costs.

- | | |
|---------------------------|--|
| 1. A summary | Describes in short your current problem, your solution for this and the estimated costs. |
| 2. Introducing the scope | Gives your audience a clear picture of the troubled environment involved. |
| 3. Problems and solutions | Describes in detail what is wrong, what the repercussions are (and what they may become) and your proposed solution. |
| 4. What is required? | A list of things that you'll need to fix the problem. |
| 5. Other options | Of course management wants the ability to save money. Here's where you give them the option to. |
| 6. Making it work | Describes which departments need to put in resources and what their tasks will be. |
| 7. A break-down | The costs of the various options, set off against their merits and flaws. |
| 8. Final words | A least plea to your audience. |

1. A summary

“The art of war teaches us to rely not on the likelihood of the enemy's not coming, but on our own readiness to receive him.”

Sun Tzu ~ “The art of war”

Keep this part as short and simple as possible. Use one, maybe two short paragraphs to describe the current situation or problem and describe how you'll fix it. Use very general terms and make sure that it is clear which of the reader's needs you are addressing.

Be very careful not to put too much stuff in this section. Its main purpose is to provide the reader with a quick overview on what the the problem is that you're trying to solve and what your final goal is. Not only will this allow the reader to quickly grasp the subject of your proposal, but it will also make sure that it will be more easily found on a cluttered desk. A short summary means quick recognition.

For example:

In the past year UNIX Support have put a big effort into improving the stability and performance capacity of their BoKS and NIS+ infrastructural systems. However, the oldest parts of our infrastructure have always fallen outside the scope of these projects and have thus started showing signs of instability. This in turn may lead to bigger problems, ending in the complete inaccessibility of our UNIX environment.

I propose that we upgrade these aging servers, thus preventing any possible stability issues. The current estimated cost of the project is \$16,260.

2. Introducing the scope

Most managers only have a broad view on things that are going on in the levels beneath them when it comes down to the technical nitty-gritty. That's the main reason why you should include a short introduction on the the scope of your proposal.

Give a summary of the services that the infrastructure delivers to the “business”. This helps management to form a sense of importance. If a certain service is crucial to your company's day to day operation, make sure that your reader knows this. If it will help paint a clearer picture you can include a simple graphic on the infrastructure involved.

The whole point of this section is to imprint it on management that you are trying to do something about their needs, not yours. It's one thing to supply you with resources to tickle one of your fancies, but it's a wholly different thing to pour money into something that they themselves need.

For example:

BoKS provides our whole UNIX environment with mechanisms for user authentication and authorization. NIS+ provides all of the Sun Solaris systems from that same environment with directory services, containing information on user accounts, printers, home directories and automated file transfer interfaces.

Without either of these services it will be impossible for us to maintain proper user management. Also, users will be unable to log in to their servers should either of these services fail. This applies to all departments making use of UNIX servers, from Application and Infrastructure Support, all the way through to the Dealing Room floor.

3. Problems and proposed solutions

“Whoever is first in the field and awaits the coming of the enemy, will be fresh for the fight; whoever is second in the field and has to hasten to battle will arrive exhausted.”
Sun Tzu ~ “The art of war”

Meaning: when writing your proposal try to keep every possible angle on your ideas into mind. Try to anticipate any questions your reader might have and bring your ideas in such a way that they will appeal to your audience. If you simply describe your goal, instead of providing proper motivation you'll be the one who "is second in the field".

In the previous section of your document you provided your audience with a quick description of the environment involved. Now you'll have to describe what's wrong with the current situation and what kind of effects it may have in the future. If your proposal covers the acquisition or upgrade of multiple objects, cover them separately. For each object define its purpose in the scope you outlined in the previous section. Describe why you will need to change their current state and provide a lengthy description of what will happen if you do not.

However, don't be tempted to exaggerate or to fudge details so things seem worse. First off a proposal which is overly negative may be received badly by your audience. And secondly, you will have to be able to prove all of the points you make. Not only will you look like an ass if you can't, but you may also be putting your job on the line! So try to find the middle road. Zen is all about balance, and so the 'art of getting what you want' should also be.

For example:

Recently the master server has been under increased load, causing both deterioration of performance and stability. This in turn may lead to problems with BoKS and with NIS+, which most probably will lead to symptoms like:

- Users will need more time to log into their UNIX accounts.
- Users may become unable to log into their UNIX accounts.
- User accounts and passwords may lose synchronicity.

Close off each sub-section (one per object) with a clearly marked recommendation and a small table outlining the differences between the current and the desired situation. Keep your recommendation and the table rather generic. Do not specify any specific models or makes of hardware yet.

Of course the example below is focused on the upgrading of a specific server, but you can use such a table to outline your recommendations regarding just about anything. Versions of software for example, or specifics regarding your network architecture. It will work for all kinds of proposals.

For example:

!	UNIX Support recommends upgrading the master server's hardware to match or exceed current demands on performance
----------	--

	Current	Recommended
System type	Sun Netra T1 200	-
Processor	Ultrasparc IIe, 500 MHz	2x Ultrasparc IIIi, 1 GHz
Memory	512 MB	1 or 2 GB
Hard drives	2x 18 GB + 2x 18 GB ext.	2x 36 GB, int. mirror

The whole point of this section of your proposal is to convince your readers that they're the captain of Titanic and that you're the guy who can spot the iceberg in time. All is not lost... Yet...

4. What is required to make this work?

“The general who wins a battle makes many calculations in his temple ere the battle is fought. The general who loses a battle makes but few calculations beforehand.”
Sun Tzu ~ “The art of war”

Now that you have painted your scenario, and you've provided a vision on how to go about solving things you will need to provide an overview of what you will be needing.

Don't just cover the hardware you'll need to acquire, but also take your time to point out which software you'll need and more importantly: which departments will need to provide resources to implement your proposal. Of course, when it comes to guesstimates regarding time frames, you are allowed to add some slack. But try to keep your balance and provide your audience with an honest estimate.

One thing though: don't mention any figures on costs yet. You'll get to those later on in the proposal.

For example:

A suitable solution for both Replica servers would be the Sun Fire V210. These systems will come with two Ultrasparc II processors and 2 GB of RAM installed. This configuration provides more than enough processing power, but is actually cheaper than a lower spec-ed V210.

5. Other viable options

“Do not interfere with an army that is returning home. When you surround an army, leave an outlet free. Do not press a desperate foe too hard.”

Sun Tzu ~ “The art of war”

The above quote seems to be embodied in one of Dilbert's philosophies these days: "Always give management a choice between multiple options, even if there is only one".

Of course, in Dilbert's world management will always choose the least desirable option, for instance choosing to call a new product the 'Chlamydia', because “it sounds Roman”. It will be your task to make the option you want to implement to be the most desirable in the eyes of your readers.

In case your proposal involves spending money, this is where you tell management: "Alright, I know times are lean, so here's a number of other options. They're less suitable, but they'll get the job done". Any which way, be sure that even these alternatives will do the job you'd want them to. Never give management the possibility to choose an option that will not be usable in real life.

For example:

Technically speaking it is possible to cut costs back a little by ordering two new servers, instead of four, while re-using two older ones. This alternate scenario would cut the total costs back to about \$ 8360,-- (excluding VAT).

If the main subject of your proposal is already the cheapest viable option, say so. Explain at length that you have painstakingly eked out every penny to come up with this proposal. Also mention that there are other options, but that they will cost more money/resources/whatever. Feel free to give some ball park figures.

For example:

Unfortunately there are no cheaper alternatives for the Replica systems. The Sun Fire V120 might have been an option, were it not for the following facts:

- It is not in the support matrix as defined by UNIX Support.
- It is not natively capable of running IP Multi Pathing.
- It will reach its so-called End Of Life state this year.

Basically you need to make management feel good about their decision of giving you what want. You really don't want them to pick any other solution than the one you're proposing, but you are also obliged to tell them about any other viable possibilities.

6. Making it work

In the case of some projects you are going to need the help of other people. It doesn't matter if they are colleagues, people from other departments or external parties. In this section you will make a list of how many resources you are going to need from them.

You don't have to go into heavy details, so give a broad description of the tasks laid out for these other parties. Estimate how much time it will take to perform them in man-hours and also how many people you will need from each source. Not only will this give management a clear picture of all of your necessities, a list like this will also give your readers a sense of the scale of the whole project.

For example:

In order to implement the proposed changes to our overall security we will require the cooperation of a number of our peer departments:

- Information Risk Management (IRM) will need to provide AS and our customers with clear guidelines, describing the access protocols which will be allowed in the future. It is estimated that one person will require about 36 hours to handle all of the paper work.
- Security Operations (GSO) will need to slightly modify their procedures and some of the elements of their administrative tools, to accommodate for the stricter security guidelines. It is estimated that one person will need about 25 hours to make the required alterations.

7. Breaking things down

You'll need to try and keep this section as short as possible, since it covers the costs of all of the viable options that you provided in the past sections. Create a small table, setting off each option against the costs involved. Add a number of columns with simple flags which you can use to steer the reader to the option of your choice.

Reading back I realize that I'll need to clarify that a bit :) Try and recall some of those consumer magazines or sites on the web. Whenever they make a comparison between products they often include a number of columns marked with symbols like + (satisfactory), ++ (exceeds expectations), - (not too good) or -- (horrific). What you'll be doing is thinking of a number of qualities of your options which you can set off against each other.

It goes without saying that you should be honest when assigning these values. If another option starts to look more desirable by now you really have to re-evaluate your proposal.

For example:

			Expected growth	
			Current needs	
System	Hardware	Costs**		
Master	V240, 2x CPU, 2GB RAM, 2x 36GB HD, 1x DVD	\$6,638.10	+	+
Master-alt	V480, 2x CPU, 4GB RAM, 2x 36GB HD, 1x DVD	\$20,575.00	++	++
Replica	V210, 2x CPU, 2GB RAM, 2x 36GB HD, 1x NIC	\$4,811.10	++	++

** : Price per unit. Multiply by amount of systems to get full price. Excluding VAT.

8. Final words

“The clever combatant imposes his will on the enemy, but does not allow the enemy's will to be imposed on him.”

Sun Tzu ~ “The art of war”

Use two, maybe three, paragraphs to make one final impressing blow on the reader. Shortly summarize the change(s) that you're proposing and repeat your arguments. Be firm, yet understanding in your writing.

For example:

We have provided you with a number of possible scenarios for replacement, some options more desirable than others. In the end however we are adamant that replacement of these systems is necessary and that postponing these actions may lead to serious problems within our UNIX environment, and thus in our line of business.

Regarding tone and use of language

At all times keep in mind who your target audience is. It is quite easy to fall back into your daily speech patterns when writing an extensive document, while at some point that may actually lead to catastrophe.

Assume that it is alright to use daily speech patterns in a document which will not pass farther than one tier above your level (meaning your supervisor and your colleagues). However, once you start moving beyond that level you will really need to tone down.

Some points of advise:

- Avoid expletives at all times.
- Avoid using hack-speak in the canonical sense. Of course you're free to use standard technical terms, but leave out things like “boxen”, “a win” (or “winnitude” for that matter) and “kludge”.
- Avoid overly long sentences. This is a trap I fall into quite easily myself, as you may have noticed while reading through this article. It tends to make following your train of thought quite complicated.
- Don't be afraid of using some fancy wordplay. Words like “adamant”, “imperative” and such tend to have more impact than “I'm very sure”, or “it is important”.

Regarding versioning and revisioning

At my current employer we have made a habit of including a small table at the beginning of each document which outlines all of the versions this document has gone through. It shows when each version was written and by whom. It also gives a one-liner regarding the modifications and finally each version has a separate line showing who reviewed the document.

Of course it may be wise to you use different tables at times. One table for versions that you pass between yourself and your colleagues and one for the copies that you hand out to management. Be sure to include a line for the review performed by your supervisor in both tables. It's an important step in the life cycle of your proposal.

This may be taking things a bit far for you, but it's something we've grown accustomed to.

Final thoughts

“Begin by seizing something which your opponent holds dear; then he will be amenable to your will.”

Sun Tzu ~ “The art of war”

Or in other words: management is almost sure to give in, if you simply make sure they know things will go horribly wrong with their environment if you are not allowed to do what you just proposed.

Of course no method is the be-all-and-end-all way of writing proposals, so naturally neither is mine. Some may simply find it too elaborate, while in other cases management may not be very susceptible to this approach. Try and find your own middle road between effort and yield. Just be sure to take your time and to be prepared for any questions you may get about your proposal.

I would like to extend my warmest thanks to those people who helped me write this article.

A few of my colleagues and a number of members of the Ars Technica and Sysadmin Talk forums provided me with reviews of version 1.0 which were invaluable to me.