

# **101 Tips on How to Buy Fundraising Software and Charity CRM Systems**



**Ivan Wainwright  
itforcharities**

## Strategy

### **1. You must do an internal needs analysis before looking at software or creating an ITT**

It is very easy to jump straight in and look at database software before considering what your requirements are – don't! You need to first do a needs analysis (a.k.a. requirements gathering phase) which will subsequently help you consider and research your options and appropriate software, and measure and compare the potential solutions against your needs.

To do a needs analysis, you should talk to as many different stakeholders of the possible new system as you can do. From database support staff and users who do the every-day operational work, through to managers who can specify what information and reports are important and directors who can advise on the current and future strategies which they are implementing. There may also be stakeholders who are 'slightly removed' from, say, a fundraising database, such as the finance team, so that even if they might not be using the new system themselves, they would still have an interaction with it.

If you are an organisation with regional offices/branches, then do get individuals involved from outside your central/head office; not only will you get different perspectives but you will engender an early user buy-in to the project which can sometimes be forgotten away from any central office.

And when you do the needs analysis, don't immediately ask 'What do you want a new database to do?' because many users may not know what such systems can do. Instead, discuss their work, ask about their processes, find out what their problems and 'pain-points' are.

At the end of this process, you will be able to draw up a 'specification of requirements' which you can then use to help you with your next steps in the whole procurement process. And if you intend to do a full Invitation to Tender process (ITT) then your needs analysis will also feed directly into that.

## **5. A new database is a chance to review all your processes. Are they really the best way to do things or have they been implemented because your old database made you do things that way?**

This is one of the most significant things you can do during a procurement and subsequent implementation. It can help provide some of the biggest benefits you can get out of procuring a new system.

All your business processes which you have set-up, are they set-up in a certain way because they really are the best approach and method to take, or were they originally set-up because your existing/old database forced you to do it that way? Or do they have no rhyme or reason anyway?! Whatever the case, buying a new system is the perfect time to review this and decide how you want to manage your processes in your new database.

In actual fact, changing processes to match your new database's approach isn't necessarily a dreadful thing to do; if, for example, it is a standard fundraising package and you do comparatively standard fundraising for some aspects of your work, then one would hope that, assuming you are following a good procurement process, that your new system may have been developed to help fundraisers with their processes, and if it used by many other charities then maybe your processes could be changed to work that way too.

But don't feel forced to do so if you really believe your processes are the best method you can use. If that is the case, then you need to have a database which is flexible enough to meet your requirements, whether out-of-the-box or with a degree of customisation.

## **6. Could your new database replace multiple existing databases in your organisation?**

If you are considering procuring a new database but you know that you currently have multiple databases within your organisation, then this may be a great opportunity to see if you can merge all or at least some of those multiple databases into one system. (And if you don't know whether other teams or departments in your organisation use other databases then ask if they do – they almost certainly will!). Unfortunately, there are too many issues and discussions on this subject to do it justice in a short chapter of this book, so at least consider the following and then investigate further if you think there could be some benefits for your charity:

Perhaps the first thing to ask is, why bother? Why does it matter if you have separate databases? The following benefits might be appropriate to consider

if you do have multiple databases and thus potentially have the same supporters on more than one system:

- ◆ Better Supporter Care: e.g. knowing that someone has run the marathon and donates £5 a month; ensuring a supporter's address is up-to-date; only sending them information on what they are interested in.
- ◆ Improved Efficiency: a simpler (technical) approach; improved data integrity, data accuracy, data consistency and data updates; and possibly saving costs (although that can depend on anything from software licenses to HR requirements).
- ◆ Maximising Marketing & Fundraising Opportunities: better marketing/targeting/data mining/predictive modelling; cross-marketing; the ability to raise higher income and higher average donations through improved knowledge; increased donor retention. And preventing multiple approaches - the last thing you want is that prospect you are after for £50,000 to be asked for £10 in another mailing.
- ◆ Support your Central Communication/Fundraising Strategy: no doubt a single system/view does this best.
- ◆ Help comply with data protection: no more issues with having to update multiple databases when people move address, opt-out of mailings, tell you that someone has died.

There are plenty of technical and business/organisational challenges if you do go down the route of a single database but the benefits can be huge.

## Costs

### **24. Fixed cost implementations are usually preferable but check carefully the deliverables & any constraints**

When you come to sign the contract, a fixed cost implementation is usually preferable, especially for charity budgeting; i.e. as opposed to a project done on time and materials. In actual fact, if you are doing a structured procurement process and are working closely with the supplier(s) during this time, then you will probably find that the costs you have asked them to provide for the different items of the project mean that they lead to a fixed cost approach anyway.

The key thing you need to do for such cost structures is check carefully the deliverables which the contract says the supplier will be providing and any constraints – i.e. any parts of the project where the supplier is only committing to x days as opposed to a specific deliverable regardless of time; or where they are specifying that a deliverable only has a specific scope, in which case ensure that such a scope is what you want and expect.

If you do find that a supplier will not commit to a fixed price for a specific part of the project then discuss how you could do some extra work initially to get a fixed price, or if you accept time and materials, then ensure you monitor such work closely when it comes to the implementation.

### **28. Never under-estimate the costs and complexity of synchronising data across multiple databases**

If you are considering procuring a new database but you have other, existing databases in your organisation which you are not intending to replace but with which you want to share the data from your new database, then do not underestimate the costs and complexity of synchronising data across multiple systems.

At a simple level, where all you intend to do is to transfer data one way from one system to another, then even that will still cost and take time and resources. But if you really want to ‘synchronise’ data between two or more databases (i.e. so you can update data in all the systems and then transfer updates between them) then “Two way synchronisation” like this takes a lot of thought and consideration and is far more complex.

For example: imagine if one of your users updates a supporter's address on one database on the same day as another user updates the same individual's address on another database, but they do it slightly (or completely) differently. When you come to the (say) nightly synchronisation of the systems, which update do you take as being correct? And then you have other fundamental considerations: do all systems have the same entity/data structures – probably not, so data can be difficult to map; if you are not careful you can get into an infinite loop of updating all databases with the same data; adding new supporter records needs careful attention; managing deletions on such systems can be very difficult; and how will you ensure there is 'guaranteed delivery' of all synchronised updates? Unless you have to implement this approach, and you have a good technical team, and plenty of money and time, then I would personally try to avoid it. It's not impossible, just very difficult!

### **39. If a supplier provides a discount on a previous quote, check you are still getting the same software & services**

If you do get an initial quote and then for whatever reason the supplier offers you a second quote at a discounted amount, then the first thing you should do (unless you know you are getting a different proposal) is check with a fine toothcomb if you are getting exactly the same proposal as you did first time around – exactly the same software with the same licenses and modules, and exactly the same services with the same number of days, level of support staff and so on. And if so, great – that's a proper and decent discount. But if not then go back and point that out to the supplier – a discount that has actually cut down on software or services isn't a discount at all, it's a revised quote for a different solution. Give them a chance to explain or make amends but if they won't then you need to consider whether the new quote is really what you want. Don't get hoodwinked!

## Managing Software Demos

### **73. Invite suppliers for pre-demo meetings: it gives your staff the chance to meet the suppliers & the suppliers can discover more too**

This is something I now do for all my database procurement projects. Before any presentations, I always invite the suppliers in to the charity's office to meet the key staff, say for a half day per supplier. Even if you send out an ITT or Statement of Requirements beforehand. And invite not just the salesperson but, more importantly, a staff representative from the supplier who would ultimately be working with the charity; e.g. an implementation consultant or support staff.

This is because it gives the supplier the chance to ask specific questions to the staff about their needs, to follow-up on any questions they might have about the ITT/SOR and to really understand the key requirements of the charity. This does mean I expect the supplier to be able to give a far more focused and specific demo which addresses the charity's needs, as opposed to a 'generic' demo which they might have done otherwise, so if they don't subsequently do that then they get marked down.

As importantly, it also gives your staff the chance to meet the suppliers' staff who they might actually be working with. It lets them see how they work, what sort of questions they ask, whether they think they could work with them, and can be a really useful task within any procurement process. Yes, it will mean a fair commitment of time on behalf of your staff involved as they will probably need to repeat the meeting with several suppliers, but this is the whole point of the exercise, and far better to spend at least some time now to find out how they all get on, than to find out later that you really don't have a good fit.

It's never let me down as a useful process.

### **76. At the demo, make sure you see how you could create reports, do queries, create mailings – i.e. get data out of the system**

It is easy to get carried away at a demo by beautiful looking screens and a 'smoke and mirrors' approach to creating a segmented mailing or sophisticated report, and many charities don't leave enough time at presentations to concentrate on the more critical aspects of actually creating the reports, queries, letters etc.

Because it is all very well putting data on your database but if you can't get the data out then it negates half the purpose of implementing the system in the first place: so many of the benefits of a database come from reporting, analysis, querying, mailings, data extractions, segmentation and so on, so if you can't get the data out of your database then what is the point of having it?

Remember this incorporates two key elements: obviously, the creation of the actual reports, mailing fields etc is one, but the ability to select and segment on the required records/data is another. Even if you can get the necessary data fields out of the system, you still have to be able to extract/report on just those records/supporters which you are interested in on a specific occasion.

Most systems these days can achieve most of this, but how they do it is key: some are definitely more user-friendly, some require far more technical skills, some have built-in functionality, some use third-party systems.

So find out for yourselves at the demo just how this is done.