

Conflict Resolution Center International

Consultant, Matthias Wenk
Community Partner, Paul Wahrhaftig

1. Situation Description

A. Organization

The Conflict Resolution Center International (CRCI) is situated in Lower Lawrenceville. It is a non-profit center promoting non-violent conflict resolution. It aims to help those in conflict to get help, to support dispute resolution initiatives, to network conflict resolvers throughout the world with each other and to provide latest information and resources. Potential 'clients' are individuals, groups, institutions, organizations or whole nations.

Paul Wahrhaftig, a conflict resolver with an international reputation, is president and the only staff member working at the office at the moment. The organization consists also of a board of Directors (approximately 30) who serve in different committees, specialized on religion, education, media, business disputes, politics and international issues. There is also the National Advisory Council and the International Council with about 15 members each, providing contacts and resources.

Since a long time, computers have been used to facilitate the work at the office, replacing successively paper files. The most recent example are the invoices that are now generated and managed electronically. However, even if E-mail has an important place as a communication medium, a lot of the communication has to be done by traditional mail, for example the newsletter.

B. Facilities

The organization is hosted in the 'Our Lady of the Angels Ministry Center', a former school building that has been transformed into offices and now hosts different community organizations like for example also the 'Christian Associates of Southwest Pennsylvania' or the Boy Scouts.

The CRCI office itself is one big room. Some desks are arranged to form workplaces for two persons, equipped with computers (a tower and a laptop with docking station) and a printer. Farther in the back is a round table for discussions and the library. Other equipment is a copying machine and a fax. The office is connected with three telephone lines; one of them is shared by the modem and the fax. The CRCI has moved in the office two and a half years ago.

To enter the building during daytime, the door has to be opened with a code. At night, an alarm is turned on and to leave or enter, a key is required. Paul mentioned that this was sometimes a problem in the past, because interns couldn't work there alone in the evening. In revenge, there weren't any problems with thefts so far.

C. Program

The office hours of the CRCI are from 9am to 5pm. Sometimes meetings with board members are held in the office. Paul also uses the facilities for conflict resolution sessions, but he does this independently from the CRCI.

As one goal of the organization is to provide addresses and other resources for conflict resolution, an important point of the staff program is the maintaining of a resource directory: The CRCI maintains a database of about 1000 conflict resolvers all around the world. It's actually the largest existing database on this subject. For a fee, the CRCI provides members links to people for specific problems. Therefore, this database is an important research tool.

The library contains important articles, books and studies on issues related to conflict resolutions. A database contains an annotated list of these articles and books. A copy of this database resides also on the web server,

and members have unlimited access to it to do queries on this database via the CRCI's home page. It is also possible to ask for copies of articles.

Also on the web, information about current events such as conferences is published. Four times a year, a newsletter, 'CR Notes' is published, providing latest ideas on CR. This newsletter is sent to subscribers (about 500) by traditional mail and a sample edition can be read on the home page by everyone.

CRCI has also been creating conferences, both in the U.S. and international. It tries to network people with similar challenges – current work is in Northern Ireland and Serbia – and promotes training, in cooperation with Duquense University – recently for example for people from Cyprus.

D. Staff

I think that Paul is quite computer literate; he has worked with computer since a long time, starting with CP/M, later MS-DOS and now Windows. He has good user knowledge of the MS Office applications, he is able for example to create and modify queries, export data and print serial letters. He also has a good basic understanding of the operating system to perform maintenance tasks on the computer such as backups or installing new programs. I noticed that he often uses the keyboard to access menus – what is faster than using the mouse. For some of the more complicated tasks, he has instruction sheets that provide him the information to perform them.

As mentioned before, Paul is currently the only staff member in the office, but he usually has interns. The computer skills of these interns are very different from one to the other. There have been interns with pretty well knowledge of Access who set up databases and enhanced them. Others had just basic computer skills. They usually don't stay for a very long time, nine months in the best cases.

E. Technical Environment

CRCI has two computers: One Pentium tower, 64MB RAM / 8GB Hard disk, including a modem. The other is a Pentium laptop, 32MB RAM / 2GB Hard disk and also with a modem. The two computers can be connected over the serial port. There is also an external ZIP drive connected to the tower, which would probably also work on the laptop, but the drivers aren't installed. As a printer, he has a Deskjet 692C. Recently, he has also acquired a scanner, and he'd like to use it to scan in articles of the library.

When an intern is employed, he usually works on the laptop and Paul uses the desktop PC. He explained that in these cases, he had to pay attention to avoid both of them modifying the same file at the same time on the two computers. Paul also uses the laptop to work at home. To synchronize documents between the two computers, he uses Lap Link. He usually keeps the files up to date on both systems every day. At the same time, this serves as a backup mechanism. Hence he rarely uses the ZIP drive for backup.

He has recently had installed Windows 98 on the tower, and he says this has noticeably slowed down the system. However, the performance seems still to be adequate for most of the applications he runs on it.

Other relevant software installed is MS Office 97, Quicken, Adobe Pagemill 3.0 and Netscape Navigator. To upload data to the web server (updates on the homepage as well as the library database and member list), he uses WS_FTP.

All databases are implemented in MS Access 97. Some of them are split up in two parts, the back end database and a database containing queries, reports and the user interface. On one hand, this makes handling a bit more complicated for Paul, but it surely makes maintenance easier and allows uploading only relevant data to the web server. The three main databases are the resource directory, the library database and the member database.

F. Technology Management

Paul does most of the maintenance by himself, for more difficult tasks he uses professional help, his hard disk vendor has done the installation of Windows 98, for example.

The database and the web page have been set up by different interns and a consultant. Paul attested them in general good knowledge, but the problem is that not all functions have been fully implemented or don't satisfy current requirements. In particular, there are sometimes extensions or modifications to be made, for example a new field has to be appended to a table, or recently the required format of the member list for the web server has changed. Paul could implement some of these modifications by himself in a provisory manner, but he doesn't understand the database enough to integrate it completely in the whole system. He also lacks of knowledge for example to create or change reports in an effective way.

H. Problems and Opportunities

One of the most important problems is probably the database itself. It isn't possible for a new intern without advanced computer skills to learn how to use the databases in a reasonable amount of time. Part of the databases is clear and intuitive to use with menus, but some tasks can only be accomplished in complicated ways and require modifications in queries and specific knowledge of the database. One point is also that there are a huge number of queries, whose purpose is not always documented. Some of them are orphaned. That makes it difficult to find a particular query. The same thing applies to the directory he exports data to; once a file has been exported, it is sometimes difficult to find it among other files with similar sounding names. Paul said that often he just worked on the database by himself, because it would have been too complicated to explain everything. Also, the database doesn't have all the desired functionalities, and some of them are broken. One example is the list of subjects: The different topics covered in a book or article are in one field, separated by semicolons. The old CP/M database allowed generating an alphabetically ordered list of the single subjects, but with Access, the expressions are not separated and integrated individually in the list.

Also, even if he knows quite well how to use Access, there are things that are difficult for him, and features that could perhaps simplify his life but he don't know about. It would be good if he had a solid understanding how to create and change forms and reports. He is willing to learn more about these things, and it would certainly be a big benefit if he were able to change the database for future requirements by himself.

Currently, he manages the member records in MS Access and the invoice handling in the billing software separately. It would be a possible to combine these two databases to economize work and to avoid inconsistencies when there are two duplicates of an address in the system.

Another opportunity he mentioned was scanning of documents, what he has already begun doing. It could be useful to integrate these scanned documents in the library database, so that requests of copies could be satisfied electronically.

To update the web pages, he currently uses Adobe Pagemill 3.0, but he only knows to use the basic features. He usually just replaces the text for updates. He'd like to learn for example how to highlight items with special effects.

He sometimes has to print large volumes, but the Deskjet printer is quite slow, so that it takes a lot of time. He mentioned that he had been advised to buy a used Laser Printer to increase the speed of printing. This is certainly an opportunity to be considered.

2. Analysis of Problems and Opportunities and Work Plan

2.1 Analysis of Problems and Opportunities

2.1.1 Microsoft Access Database

The first problems are the Microsoft Access databases that have not all necessary and desired functions implemented and are difficult to understand for new interns, especially for those with few computer skills. It would be desirable to have a database that can intuitively be understood and used even by people without prior computer knowledge.

The databases are used to manage addresses of the members, conflict resolvers and the annotated library. This is an essential tool to enable networking between conflict resolvers, the very goal of the organization. Since often modifications have to be made and research requests have to be answered, it is important that not only Paul can use these databases, but also the interns.

One solution is for me to improve the database so that it is easier to use and to modify. This can be done to different degrees, from reorganizing the forms and adding new buttons, creating pull-down menus, up to write code that allows doing all modifications automatically, for example adding a new field. The solution will be a tradeoff between automating certain tasks and instructing Paul how to do it, perhaps with the support of how-to-sheets. A very important point is that the database will be maintainable and changeable by Paul, rather than setting up a fancy, but complicated and inflexible application. Basically, everything that could change in future should be possible to implement without deep knowledge of Access.

An example for this is what we did in a past meeting: There was a report where the addresses had to be selected in a query; that would be too complicated for an inexperienced intern. We created a macro that asks the user for some criteria of these addresses and opens the report. At this occasion, I showed him how to create a new button and macro necessary to do this, so that he will be able to add such buttons later on his own.

An alternative approach would have been to use another database than Access that is more easy to use. However, such a database would be probably less flexible, and considering all the time necessary to install it, convert the old addresses and learning the new database would never be justified. I also considered updating to Access 2000. However, my experiences with this version didn't really convince me, and it would probably run slower on the current hardware. I think it's better to stick with Access '97, since this program already offers all the possibilities needed.

Impact:

Having an easier to use database would save a lot of time for interns to get into the database, and also for Paul to use it. Furthermore, having it documented would make maintenance easier, in particular for future interns that have the knowledge to do this. It would also be a good base to implement new features, for example maintaining scanned documents in the database. With some tutoring, Paul would be able to adapt the database for future enhancements.

If we keep Access 97, there is no need for new hard- or software, and there is no time lost in learning to use a new program.

Feasibility:

It's possible to achieve some goals during the rest of this semester. Of course, the process of learning to use such a complex program as Access is never fully terminated, and there will always be new modifications to be made to the database, but with the learned material, there will be results in short and long term, for every feature learned. Paul is motivated to learn more about Access and he sees the need to make the database easier to use. There won't be any resources necessary, and I think I know Access well enough to accomplish this task. One risk is that as he begins to create more macros, forms, reports and query that the database becomes even more complicated. To avoid this, I think it's important to maintain a thorough documentation of every object in the database and how they are related.

2.1.2 Web Page

The web page is an important resource for conflict resolvers. It was set up by earlier consultants and interns. However, Paul has difficulties updating and enhancing the page by himself, and there are certainly a lot of opportunities to make the page even more useful. The web page is directly tied to the mission of the CRCI, as it enables conflict resolvers all over the world to get in contact with CRCI and through it get access to resources and contact to other conflict resolvers.

Paul asked me to show him how to do use different features, how to protect some section with passwords and how to organize the files in subdirectories to cope with the growing number of files of pages delivered by board members. I also think it would be useful to bring the pages to an easier to maintain format, for example using frames, so that it will be easier to modify contents in future, and to create templates that will allow him to make new pages rapidly. Also, the structure of the web pages should be somehow documented to make modifications easier.

Teaching is certainly a much better approach than just me modifying the page, since things change very rapidly and web pages need to be updated often to be useful and attractive.

Other possibilities that we didn't talk about yet would be for example setting up a bulletin board or searching and including links to other web pages on conflict resolution.

Impact:

With its presence on the web, the CRCI allows a lot of people to get information more easily. I think the Internet is an important means for the CRCI to accomplish its mission. Since the page represents CRCI to a lot of people, its content is of major importance.

With improved skills, Paul will be able to update the page more easily and to include home pages of board members, without this consuming too much of his time. A good documentation and organization would minimize the work of future experts to implement special features.

There is no accurate need for new hardware or software at the moment. Most things can be modified and uploaded on our own, only implementing certain features, like enhancing the credit card form or adding password protection would probably require contacting the provider of these services.

Feasibility:

I surely don't have time to show him all possible things he could do on the web. Also, I don't have very profound knowledge in this domain. I think therefore it is useful to focus on basic issues that will allow him to update the page regularly and to include pages delivered by board members, keeping all links working and using frames correctly, also after I have left.

2.1.3 Computer skills in general

Computers are used not only for the databases, but also for other tasks, for example word processing, scanning of documents or invoice management. However, he doesn't always know for example how to link documents of different applications, for example how to use addresses from an Access database in Quicken, that is used for billing for example people that order services on the web page.

Computer skills aren't tied directly to the mission of CRCI, but as I pointed out in the other two paragraphs, computers are an absolutely essential mean of the organization, and knowledge to use them more efficient would have positive influence on the goals of the organization.

To help him using his computer, I first need to know the programs Paul uses, for example by watching him using them. Then, I can show him how to do certain things, or make suggestions what other software to use.

Another approach is also be to point him to resources on the web, for example FAQ's that allow him to find responses to his questions on his own.

Impacts:

Since computers are already an important pillar of the organization, optimizing their use will be beneficial to the organization as a whole. Better knowledge of the computer will allow Paul to work more efficiently and use less

time with computer problems. The computer should only be a tool to make work easier, so that he can focus on more important things.

Feasibility:

As for the other problems, it can't be said that the work is completed at a certain point; there is always more to learn and to discover, but already learning a few new things can have a positive effect on the workflow. Paul is motivated to learn how to use the computer more efficiently, for example to use the scanner to import documents into the computer and publish them on the web page.

Even if I don't know the particular programs he uses, there are certainly things I can help him to do, notably where to search for help if a problem arises. These things will also be valuable when I leave and when he installs other programs.

2.2 Work Plan

The following table shows a plan when we accomplished the different tasks. It is only an approximate overview, because a lot of things are related to each other, and we had to come back to certain things at a later point in time.

	Resp.	Date	March	April	May
MS Access Database					
Tutoring					
Creating Buttons	cp / co	3/2/00			
Creating and changing Macros	cp / co	3/2/00			
Use of queries / filters	cp / co	3/7/00			
Queries with parameters	cp / co	3/7/00			
Aligning elements on reports and forms	cp / co	3/7/00			
Important documentation for future maintenance	cp / co	5/4/00			
Documentation for use by interns	cp / co	5/4/00			
Creating forms with parameters	cp / co	4/27/00			
Counting, computing sums, averages, etc.	cp / co	4/13/00			
Adding new fields to a table and all derived objects	cp / co	5/4/00			
Creating context menus (popup menus)	cp / co	5/2/00			
Sending emails to a person or group	cp / co	5/2/00			
Exporting / linking to other documents (Word)	cp / co	4/27/00			
Work on the Database					
Naming of objects, grouping macros	co	5/4/00			
Document structure of database	co	5/4/00			
Subject list in library database	co	4/27/00			
Simplification of divers forms	co	4/27/00			
How-to-sheets and/or help screens	co	5/4/00			

cp: Community partner

co: Consultant

 : spring break

Web page
Tutoring

3.1.1.2 Sustainability

The goals achieved will lead to sustainable increase of capacity, because the forms and queries created can be used also in future and enhanced if necessary. The only danger is that the database becomes more complex and less easy to use as more and more buttons and functions are added. In order to avoid this, every new button that is created should be well commented so that it doesn't happen that a similar button is created again later because it wasn't clear that the already existing button did exactly what needed to be done.

3.1.1.3 Recommendations

I think the most important thing to remember when working with the database is to continue documenting it thoroughly and using names in a consistent manner. It may seem a loss of time at the moment, but it can speed up later changes and troubleshooting a lot, in particular if some other intern or consultant has to work on it. There are many possibilities in Access to document elements on a form; for almost everything, a text can be defined that is displayed when the mouse is over it.

To further improve ease of use and functionality of the database, it is certainly useful to read articles in books or on the Internet on this subject. Also FAQ's can be very helpful.

3.1.1.4 Resources

I'm about to create a documentation of the current database, and wrote some guidelines how to document new functions (see appendix). This should allow keeping the documentation up to date, allow other people using and/or modifying the database within less time, save time and help to get rid of orphaned objects. Another resource is books on Microsoft Access to learn about features of this program. However, it is sometimes difficult to find help there on a specific problem.

There are also many resources on the web concerning Access. A good starting point is <http://www.mvps.org/access>. This page contains many useful tips, and, most importantly, links to other interesting pages and other resources, as for example news-groups or books. A lot of the stuff there may be of a too advanced level, but there are also useful tips and workarounds for common problems.

Especially the search function on this page may be useful. For example, we were once looking for an easy way to filter records in query according to a criterion in a form, but only if the criterion given is not NULL, in this case we wanted all records. Later, I found that by searching the key words '*query all if null*' – the answer would have been at the first position.

3.1.2 Web Site

The goal of the work with the web site was to allow Paul to update and enhance the pages frequently and easily.

3.1.2.1 Outcomes

To create and modify his web pages, he currently uses Adobe Pagemill. However, as I worked with the program, I realized that its use is not always very intuitive, and on the desktop computer, it crashed when we tried to use frames.

First, my intention was to use frames for his pages. However, along with the difficulties with Pagemill, and because he cannot update all parts of the site by himself (the credit card form in particular), I decided it would be better for the moment to stick to a version without frames.

The main problem without frames was that it was very tedious to change buttons in the menu list, because this required changing every single page that contains these buttons. Therefore, I wrote a small C program that does this work for him, and at the same time allows having a bit fancier buttons that change when the mouse is over them.

To create new buttons, it is necessary to create graphic objects with the text on it. Again, the software currently available on the computer doesn't allow creating very fancy buttons, and I didn't find free software that does this in an easy to use way.

The best method I found was using Microsoft Word, because it allows to center text on the button and to implement some 3D-effects. The images can be exported via Paint to get GIF files.

3.1.2.2 Sustainability

With the software I wrote, Paul is able to change buttons on his own or add new ones and update them to all web pages, also if new web pages with the same button list are added. For the moment, I consider this an acceptable solution that may be kept for an indefinite time.

When sometime he is going to us another web authoring program and changes the design of the pages, these programs may become unusable, but the new program should be able to replace this functionality, so that he doesn't need these utilitarian programs any more.

3.1.2.3 Recommendations

As I mentioned, I consider the current solution as acceptable.

However, if sometime an intern is hired to redesign the web page, I recommend purchasing a new web authoring program. One reason is that, in my opinion, Pagemill doesn't allow to make much fancier pages that are at the same time easy to maintain, update and enhance, without profound knowledge of the program. There are also many new features, for example style sheets or JavaScript, that aren't supported by Pagemill. Also, most programs use themes and a lot of templates to make it easy to generate pages with a consistent look. The decision what program to purchase should possibly be taken with the intern, according to what program he has experience in. Recommendable programs are for example VisualPage or Dreamweaver from Macromedia. It is very important to make him understand that the pages should be easy to update and change later without his help.

Another point is that the possibilities to create buttons with Microsoft Word are quite limited. However, I didn't find a program that allows easy design of buttons. If there is at some time an intern with strong computer skills, I strongly recommend purchasing Paint Shop Pro, a quasi-standard to generate graphics for the web. In particular, it has the capability to record macros that would allow Paul later to generate new buttons by himself.

Another recommendation would be to investigate texts about design of web pages; there are a lot of useful tips that can be applied directly and make can help to make the page more and more attractive, and they are available on the web for free, to download and print out. I've included some references in the following section.

3.1.2.4 Resources

An interesting web site with general tips how to design web pages can be found at

<http://www.gooddocuments.com>.

I found an interesting program to create customized buttons using Java applets at <http://www.coffeecup.com> (Applet Button Factory). Although using Java applets for this purpose may be a bit 'heavy', the program is certainly worth checking it out.

Other useful shareware and freeware concerning web design can be found for example at <http://www.tucows.netreach.net>.

3.1.3 Computer skills in general

In order to use the computer more efficiently, it is useful to acquire general computer skills, be it to create and use shortcuts for certain tasks, to make linking documents more easier or use the scanner more efficiently. Like this, time may be saved, and therefore allow more time to be invested in other things than using the computer.

3.1.3.1 Outcomes

We didn't go deeply into other issues than the access database and the web page. However, there are things that are useful also for other programs. For example, I showed him how to make a link to his homepage folder on the desktop for faster access to these files.

When trying to use the scanner with OmniPage Pro 10, we encountered the problem that the program isn't able to use the scanner as it is supposed to, but it is necessary to use the scanner software to scan the page and then pass the result to OmniPage. This makes scanning of multiple page documents tedious.

A problem we sometimes encountered were crashes or the computer slowing down significantly, maybe due to the many accessory programs running in background.

Something that may become a bigger problem is the research of documents that are stored in rather unstructured directories. At the moment, the 'find' function allows finding documents, but this may become more and more difficult in future, as the number of files grows further.

3.1.3.2 Sustainability

Since he most likely won't change the operating system in near future, these general things learned will remain of value.

3.1.3.3 Recommendations

OmniPage seems to be a pretty good program for OCR, and the user interface makes the use easy – at least in theory. The problem with the link between the program and the scanner may be fixed soon. The omniPage homepage is a good resource to get answers to questions and information.

To face the problem of crashes, it may be helpful to reinstall windows at some time. This will be possible without slowing down other work once the second computer is set up and working. It should then be carefully evaluated what utilities to install. I strongly recommend setting up a well-organized directory structure for the documents to make retrieving them easier.

3.1.3.4 Resources

To get help for OmniPage, the Caere support web site at <http://support.caere.com> may be useful.

3.2 Technology Literacy Benchmark areas

3.2.1 Strategic Technology Planning

The organization doesn't really have a long-term technology plan. Upgrades are made when they are available for good conditions, or when they turn out to be necessary. However, a general important point to be considered when purchasing new hard- and software is that new product doesn't just work by themselves optimally most of the time. In fact, there is a rule used by many large companies called 70/30 rule that says that for every \$1 invested in technology, 30 cents would be used for hard- and software, the remaining 70 cents for training and support. In our case, this doesn't necessarily mean money to be spent, but also time invested. This certainly shouldn't discourage from investing into new technology. Even if it may seem expensive at the beginning, the investment can turn out to be valuable – if a good choice is made.

I think my community partner is already aware of these points. As I mentioned, he has already used computers for years and gone through several upgrades.

In order to make decisions what software or hardware to purchase, how to maintain the computer and a lot of other issues, there are many resources available on the Internet, even particularly focused on non profit organizations. A good FAQ can be found at <http://www.nonprofits.org>.

3.2.2 Technology Use – Staff Level

It is certainly useful to exploit computer skills of interns. In order to take more advantage of this resource, it may be important to consider what software interns are already familiar with. For example if some time an intern has to redesign the web site, he should be asked what kind of program he has experience with before purchasing it.

A problem that has been encountered in the past is that interns were able to set up web sites and databases, but Paul couldn't implement changes himself afterwards. It is therefore important not only to consider computer skills when choosing an intern, but also if he has tutoring skills and is able to make a solution that can be maintained without his help afterwards. It should be explained clearly at the beginning that this has the first priority.

3.2.3 Technology Use – Organization Business Systems

Paul uses Quicken to manage billing. However, the address base isn't tied to the member database, what would be useful. Gradually, more and more of the accounting may be automated in future.

To send serial letters, he uses in general the exporting function of Access to fill in the fields in serial letters in Word. I implemented a function that allows to export a set of addresses to a Word document in the same way as mailing labels are printed. Another function I've implemented is to export such a set of addresses to Outlook Express, so that emails can be written to a group of people.

3.2.4 Technology Use – Organization Networks & Internal Communications

Regarding the size of the office and the number of computers, a network isn't necessary and internal communication can be done without using computers, or by simply exchanging data files between the computers, if necessary. Laplink is a software that allows to do this in an easy way. Only, there is a problem to it that newer versions of Access databases can be overwritten when the transfer is done in the wrong direction, because Access changes the date of the last modification even if it was only open for read. Special care has therefore to be taken when such databases are transferred.

What concerns security, the computers have already be attacked by viruses, but there are virus protection programs installed that could clean the hard disks from these viruses.

3.2.5 Technology Use – External Communications

The organization maintains a web site on the Internet, one of the issues I was mainly concerned with during this semester.

E-mail addresses and URL's of the member's web sites are kept in the member database, and e-mail is used as an important means of communication – to people who have it available.

One possibility that would be interesting to look at would be setting up sort of a bulletin board, allowing more interactive communication for the visitors of the web page.

At the moment, mass e-mail is sent without d-lists. This would also be something that may be interesting for future when more and more mail is sent electronically.

3.2.6 Technology Sustainability

Paul manages the technology mainly by himself, with occasional support from other people. Neither an inventory nor a budget is used. Since upgrades occur irregularly and not always predictable, based on donations or good opportunities, or accurate need, I think it wouldn't be easy to set up a fixed budget for the computers.

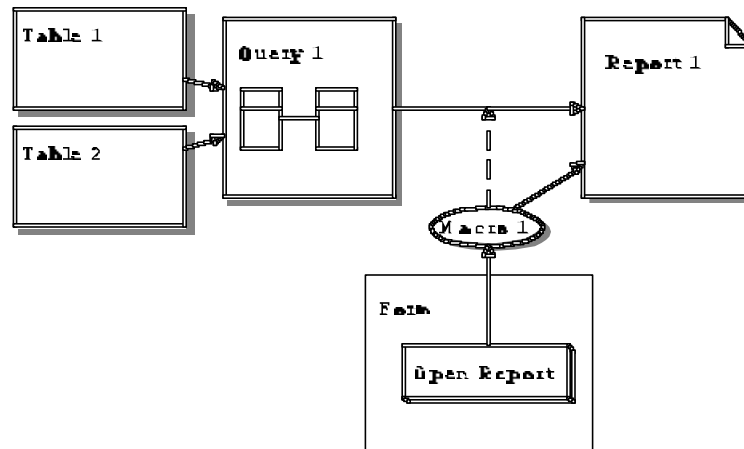
Appendix: “How-to”-sheets for the work with Access

As I mentioned before, one of my tasks consisted in writing manuals that explain how to do certain tasks. Some of them are specific for the database at the CRCI, and therefore not of general interest. In this appendix, I’ve included two that are of rather general nature, one of them explaining how to use queries, filters, reports and buttons with macros, the other one giving some suggestions how to document and maintain the database.

A. Using Queries, Filters, Macros to generate Reports

Opening a Report with a Macro

This diagram represents the objects involved when a report is opened with a button in a form.



Creating a new button to open a report requires the following steps:

Find the Names of important Objects

- Find out the name of the report you want to open and of the underlying query (In the design view of the report, go to Properties; the query name is on page ‘Data’ in line ‘Record Source’)

Create the Macro

- If there exist already macros with a similar purpose (for example macros that open the same report), open this macro and add the new submacro on a new line, giving it an own name
- Otherwise, create a new macro and save it under a meaningful name (for example openReportname)
- Choose Action ‘OpenTable’ and set the fields at the bottom of the window:

Report Name	-> Name of the Report <-
View	Print Preview
Filter Name	
Where Condition	-> Condition <-

Fill in the name of the report and, if you want, a condition for the records to be selected

- As indicated in the diagram, a macro can take some influence on the data in the report. This is done by setting a condition. It can be an expression like `Member=Yes`, where Member is contained the underlying query

xxx
xxx
xxx

This button shows the names of the submacros

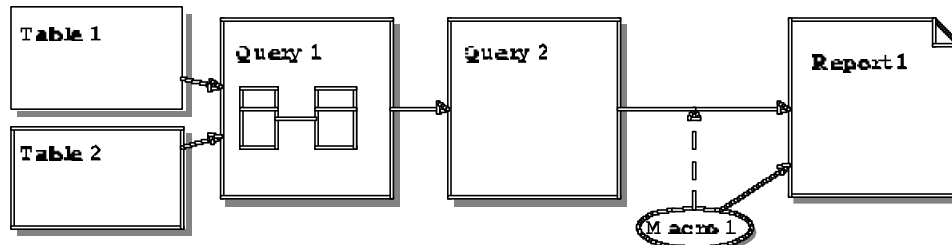
Create the Button

- Go to the design view of the form you want the button to insert in and create a new Command Button
- Go to the properties of the button, to the page 'Event' and enter the name of the macro in the line 'On Click'.



This symbol creates a new Button

Asking the user for parameters upon opening a report



To add this functionality to a report, you will probably need to add a new query ('Query 2' in the diagram) between the query the report was based on so far ('Query 1') and the report.

Create a new query

- **Create a new query, choose 'design view'**
- In the window that 'Show Tables' that comes up now, go to page 'Queries' and double-click the old query ('Query 1'). Then hit 'Close'
- Make sure that all fields are selected. The easiest way to do this is to drag '*' in the field list.

Instead of dragging '*' to the field list, you may also go to the properties and set 'Output All Fields' to 'Yes'

Asking conditions

- Drag the column you want to select on to the field list and specify the condition. Instead of using a specific value, insert the question in square brackets
- Useful examples are:
Select a specific day in a date field: = [What day?]
Select a range of names: Between [Name to start with]
And [Name to stop]
Allow use of wildcards (*, ?): like [Last name?]
Show only members when the user enters 'yes', otherwise all records:
[Only show members?] or yes

Asking for constants

- Add a new column to the query: in the line 'Field:' enter an expression of the form: Expression: [Question]
- Examples are:
FirstPage: [Enter the first page number]
Title: [Enter the title of the report]

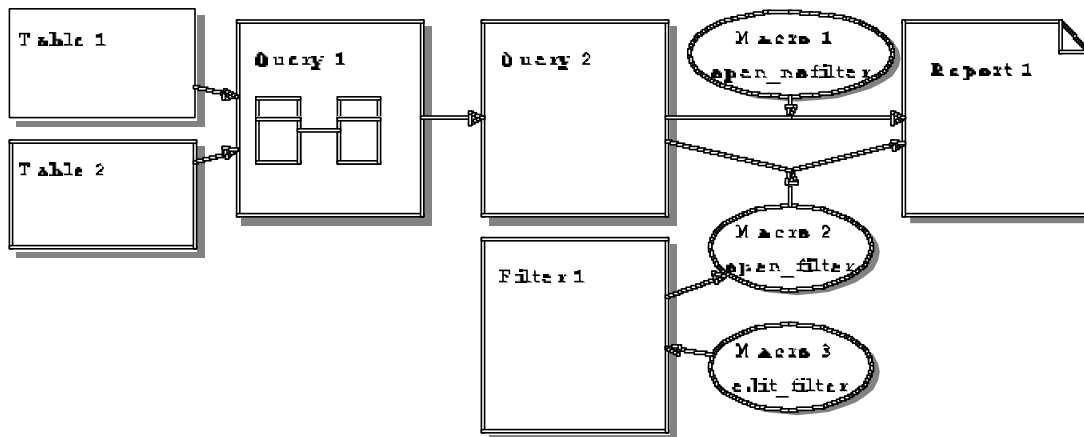
! If you use two times exactly the same question, the question is only asked once, and the same value is used for both places the question is asked.

Changing the report

- Change 'Record Source' of the report to the name of the new query
- Constants that have been asked from the user can be used anywhere.
You can for example make a Text Box containing the field 'Title'.
The page number can be used in the page number field: =FirstPage+page-1

Using filters

- ➔ Filters have the advantage over parameters asked from the user that they are much more flexible, records can be selected like in a query.
- ➔ The advantage over changing the query is that the report can still be opened without the filter, for example by an other macro that selects a fixed set of records.



Create a filter

- Create a new query, choose 'Design View'
- Add the query you want to select records from in ('Query 2' in the diagram)
- Select all fields by dragging '*' to the field list
- Drag all fields down you want to select the records on
- Save the query under a meaningful name (for example fltQueryname)

Opening the report with the filter

- Create a new submacro to open a report as seen in 'Opening a Report with a Macro', using the action 'OpenReport'
- Enter the name of the filter in the line 'Filter Name'

Opening the filter with a macro

- It is useful to have a submacro to open the filter in design view.
To do that, use the action 'OpenQuery' and enter the parameters in the lines below.

Query Name	-> Name of the Filter <-
View	Design
Data Mode	Edit

It is useful to have all submacros concerning the same report in the same macro. Such a macro could look like this:

	Macro Name	Action	Comment
	open_nofilter	OpenReport	Open the report without filter
	open_filter	OpenReport	Open the report with applying the filter
	edit_filter	OpenQuery	Open the filter in design view for editing

Using Parameters in Forms

- Add a text box, a combo box, a list or a check box to the form
- Go to its properties, on the page 'All' change 'Name' to some meaningful name
No-No's: Caption, Count, Cycle, Dirty, Event, Filter, Module, Name, Page, Pages, Parent, Picture, Section, Set, Tag, Visible
- If the field has a specific data type, go to 'All' and change 'Format'.
- You may now use this field in any query or report as a variable, using the expression
`[Forms].[name of the form].[name of the field]`

Example: Title of a report

- Add a new field to the form
- Give the field a name
- Open the report and add a new text box for the title
- Enter an expression like this in the text box:
`= [Forms].[formname].[fieldname]`
or, if you want to have constants in the title text:
`= "Text1 " & [Forms].[formname].[fieldname] & " Text2"`

Example: Select on yes/no fields

- Add new fields to the form
- Give the fields names and set the format to yes/no
- Open the query underlying the report you want to open, and add criteria for the yes/no fields you want to select on.

If you want to allow fields to be left empty so that these aren't considered as criterias, you have to use the following expression:

```
[Forms].[formname].[fieldname] Or  
(IsNull([Forms].[formname].[fieldname])=True)
```

(This means, select the record if the name matches, or if the field was left empty by the user)

B. Documentation of the Access Database

Here are some suggestions that can help you to maintain the database in a way that it is easily understandable for future interns that work with or on the database. Some of it may seem like a loss of time for the moment, but it can help save a lot of time later.

Tables

- Use the description field in the design view of the table to describe non-obvious fields, for example when the field contains codes.
→ This description will be shown in the status line when you enter data in the table.

Queries and Filters

- Use consistent naming conventions (qryXXX for queries, fltXXX for filters), always use queries as sources (in a report or another query) and filters as filters
→ This avoids confusing queries and filters (you could use a filter at the place of a query – however for clarity filters should only be used as filters)
- In the properties of the query, tell what query or table(s) it is based on, and where it is used (what report or other query)
- In the properties of a filter, tell what query it is based on, and where it is used (for which report and what macro opens the report with this filter)
→ This makes it much easier later to find and delete queries and filters that are no longer needed, without the risk of deleting something that is still in used somewhere else

Forms

- Use a *label* to describe the purpose of the form
- Group elements together that belong together. If you are using parameter fields, it should be obvious what button it belongs to. Also, when you have a button to change a filter and one to open the associated report, it should be visible that they belong together.
If the form gets too crowded, consider creating a new form that contains buttons with similar functions, and create a button in the first form that opens this new form (using the macro action *OpenForm*). Take care that the hierarchy of the forms remains logical.
- For fields where the user can enter text (like a report title, or criteria), use the property *Status Bar Text* (under *Other*) to list the queries and/or reports where this parameter is referenced to.
- Most elements, like buttons or text fields, have the property *ControlTip Text* (under *Other*) to describe what it does. This text will show up when you touch the button with the mouse.
For example, a button could have a description like this:
“Opens Report XY, using Filter Z. This Filter can be modified with button AB”
→ Even if the purpose of a button may seem obvious for you, someone else or even you

yourself may not understand later what it does or is supposed to do. Documenting the functions also avoids creating other, redundant buttons with the same functionality later.

Reports

- Tell in the report properties what form opens the report, in particular if there are parameter fields in the forms that are used (for example a title, or the first page number)

Macros

- Group macros together. It is in general good to have one macro object for each form, you can give it the same name as the form, like that it is clear what belongs together.
For the sub-macro names, it is a good idea to use the names that are on the buttons.
- In the comment field of every action, tell what it does (for example what report or query it opens)

Deleting objects that are no longer used

Deleting objects that have become orphaned is important, but also quite tricky. Here are some points that could help doing this:

- Reports are usually the most easy object to delete, because you can open them and see what they do, and decide if you need it in future or not.
If you have it documented properly, you'll know from what form it is opened, so that you can delete the corresponding button and macro there (if there is one)
Before you delete the report, check to see what its record source is. You should then consider deleting this query also, together with its associated filters.
- Deleting queries is trickier, because it can be used at different places. If you just deleted a report based on a query, this doesn't necessarily mean that the query isn't used any more, it may also be used in other reports; That's why it is important to tell in the properties of a query what reports it is used in.
- If you are not sure if you really can delete an object, you can 'mark' it for deletion, for example by putting a star in front of the name. Like that, you can continue using the database, and when the query is needed somewhere, an error will be display because the query is not found. If everything seems to work OK, you can probably delete the query without danger.