Research Techniques

Alan Dix

School of Computing
Staffordshire University

Context

This document was originally written to support a lecture on "Research Techniques" given to final year undergraduates while I was at the University of Huddersfield. Since then I have used the basic material with some variations to undergradutae, masters and PhD students. Althogh these notes were written with the original audience in mind, I hope that some of the ideas will also be of use students elsewhere including those undertaking MSc projects, PhD studies or doing research in general.

Contents

1. Introduction – what is research 1
2. Other people's work – what they write 6
3. Other people's work – what they say 9
4. Other people's work – what they make 10
5. Your own work 14
6. And finally 20

1. Introduction – what is research

Final year projects cover such a wide range of areas and each will have its own specific techniques, methods and tricks. So, if this document dealt with specific techniques, such as experimental method or simulation, it would only be of interest to a small number of you. Instead I shall look at some of the general principles that can help in your own process of research, innovation and development.

We will first ask what research is and whether, as final year project students, you ought to be doing it anyway. After this, and forming the bulk of the document , we shall look at the process of research itself.

Research?

- what is it?
- should you be doing it?
- how do you do it?

Pause for a moment, think of the word 'research' — what images come into your mind? Don't try to define it, just think about what it means to you. Write down a few ideas below.

My images of research

Definitions of research

"Systematic investigation towards increasing the sum of knowledge"

(Chambers 20th Century Dictionary)

"an endeavour to discover new or collate old facts etc. by the scientific study of a subject or by a course of critical investigation."

(The Concise Oxford Dictionary)

Look at the two dictionary definitions above. Which one, if either, is closest to your images of research? The first is more lofty — looking for totally new knowledge. The Oxford definition also includes the collation of existing knowledge. The image which immediately springs to my mind when I think of research although not what I do) is of Brains on Thunderbirds — the white coated scientist bent over a bubbling test tube. This fits well the Chambers definition. In contrast, the 'collate old facts' definition suggests heads bent over old manuscripts in the British Library Reading Room.

These are just two types of research, we'll consider a few more ideas, but you may have thought of something different again.

The first of this list , the scientist, is perhaps the archetypal image of experimental research. Of course, the white coated technician is less well respected today than in the late 60's when Thunderbirds was first screened. A more acceptable modern image might be the botanist in the Amazonian rain forest, observing and discovering new creatures before they fall under the axe and are consumed by fire. Behavioural and experimental psychologists would also fall under this general heading.

The social scientist's methods are different from the laboratory (although quite similar to the botanist). They include interviews, observation and the collection of data. All of which will be needed at very least during your requirements elicitation from your client. Notice how the world of the social scientist is far less controlled than that of the laboratory or even the botanist. You can put a beetle in a 'bug box' and examine it, but social situations collapse when dissected too closely. The ecologist has similar problems.

Historical research corresponds to the British Library image, reading original sources and other peoples books and papers. Of course it does not stop there. The aim of the historian is to understand the historical processes as well as to record it. One of the key things a historian has to recognise is that all the sources are biased — written from a particular perspective by a particular person for some particular purpose. You will be faced with similar issues, albeit from modern sources.

Types of research

- the scientist
- the social scientist
- the historian
- the journalist

? R & D

Journalists operate in a somewhat similar fashion. They do not expect to generate new knowledge (although they may occasionally generate fiction). Instead, they cull from exiting sources: people, books, other newspaper articles etc., in order to write about their chosen subject. Note the range of sources they must draw on. Also note that they will not attempt to thoroughly understand the subject they are writing about, nor do they attempt total coverage of the area. They have a goal, to write an article, and they find out just enough to do that job. The academic must take a deeper and wider perspective than this, but do not underestimate the skill of the journalists. When some event happens they have to find out enough within a few hours to be able to write cogently about it.

Finally we have industrial Research and Development. What is the research element in that. Well, some firms do have 'blue skies' research laboratories whose job is to find exciting new things, rather like (but better resourced) than a University research atmosphere. However, most do not have sufficient spare resources to use for this sort of enterprise. Instead, the job of the commercial researcher is to draw on existing knowledge and bring it to bear on a particular problem. This may involve creating something new, but usually by some adaptation of an existing solution. Like the journalist, the industrial researcher is very goal directed, but has to do more. The journalist merely has to gather enough information to write the article, the industrial researcher must understand the information in order to apply it to a new situation, that is, the product under development.

Clearly, the R & D situation is closest to your project as you too have a client and a product to produce. However, the situation is not identical. Your aim is not only to produce a product, but also to obtain a degree. Although your time may hardly seem leisurely, you do in fact have more 'leisure' to reflect upon the work you are doing, taking a more academic angle. In particular, this might mean being somewhat more broad in your searches for information and considering more alternatives to a problem, even after you have found a solution which works.

So, given these definitions of research and examples of researchers, should you be doing research in your project, or should that be left to the PhD students and academics. And, if you should be doing research, which of the above types should it be.

The project

- integrative
- independent
- interesting
- intellectually challenging

? innovative

not necessary but good

Let's think about your project in terms of some 'I' words. First it should be integrative, bringing together knowledge from different areas. In most of your courses you consider some particular aspect of computing, business or mathematics. In your project you must use knowledge from a variety of areas and sources. Some things you may already know from a course you have done. Other things might need further investigation.

The project is an independent piece of work which is interesting to you. As it is YOU doing the work, you are only expected to produce what is reasonable for a final year undergraduate.

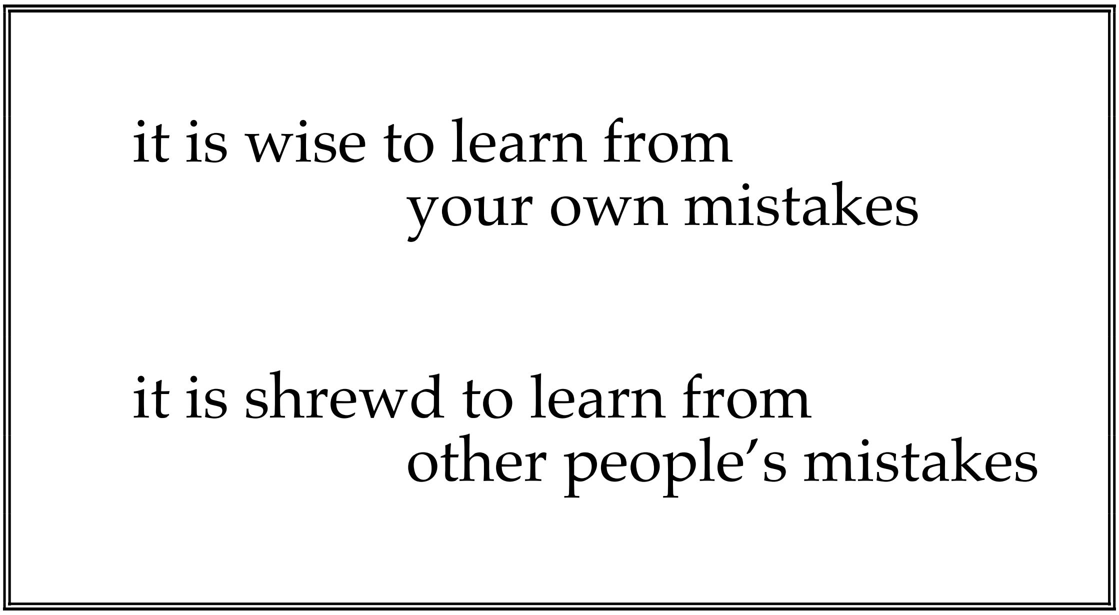
However, as it as an academic project, part of an honours degree, it must also be intellectually challenging. Again, not the minimal solution to a problem, but one that involves significant academic work before hand and analysis of your results after. One would hope that this will also contribute to making the project interesting.

To be integrative and intellectually challenging the project must clearly involve research in the sense of 'collating old facts'. That is, aspects of the British Library image combined with the focused attitude of industrial R & D.

But, the crunch, should the project be innovative — breaking new ground, extending the sum of human knowledge, generating new and novel solutions?

Well, it would obviously be nice to develop some new algorithm or discover some new fact about IT, and the best projects will involve some level of innovation, but this is an undergraduate and not a research degree, so it is not necessary. On the other hand, it would be hard to apply even standard techniques to a new problem without there being something novel about it. Every situation is slightly different and you will have to use a level of ingenuity (another 'I' word) in dealing with it.

So, given you should be doing some research, how do you go about it? You all know the old saying: "you should learn from your mistakes". Indeed, this will be an important part of your final report. You will have to reflect upon what did and did not work. You will be expected to diagnose your problems and learn from them. However, you do not have enough time to make too many mistakes, so you should avoid as many as possible.

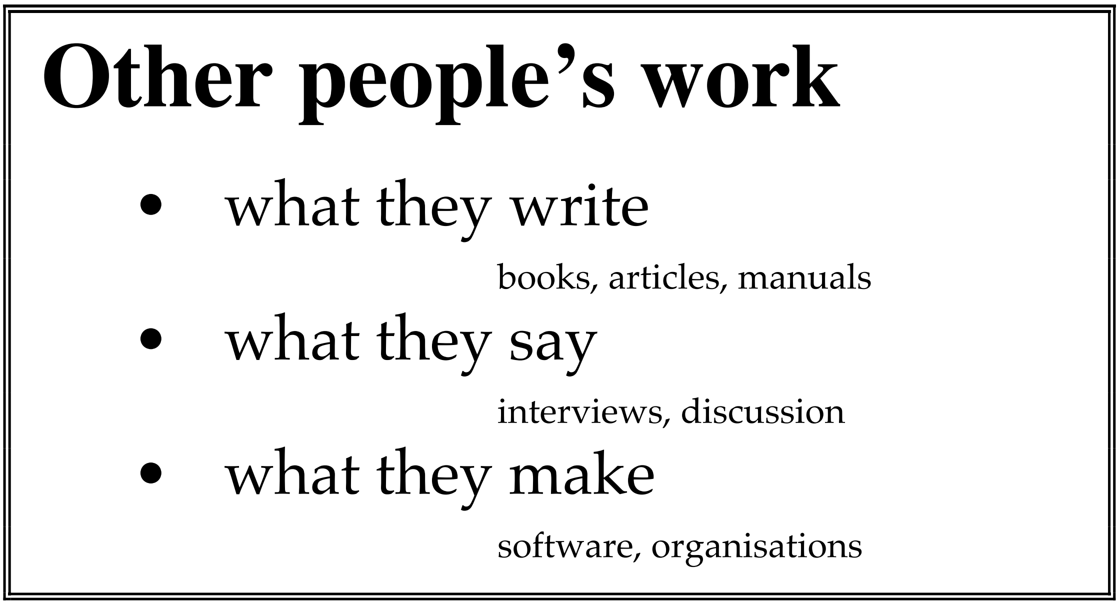


it is wise to learn from
your own mistakes

it is shrewd to learn from
other people's mistakes

How do you avoid making your own mistakes? Well although it is good to learn from your own mistakes, it is shrewd to learn from other peoples mistakes. Find out what other people have done right and done wrong before making the same mistakes or even working hard to discover the same good things.

To do this you must study other people's work before embarking on your own — that is, more research. But it is not all of the British Library (or even Huddersfield Library) kind.



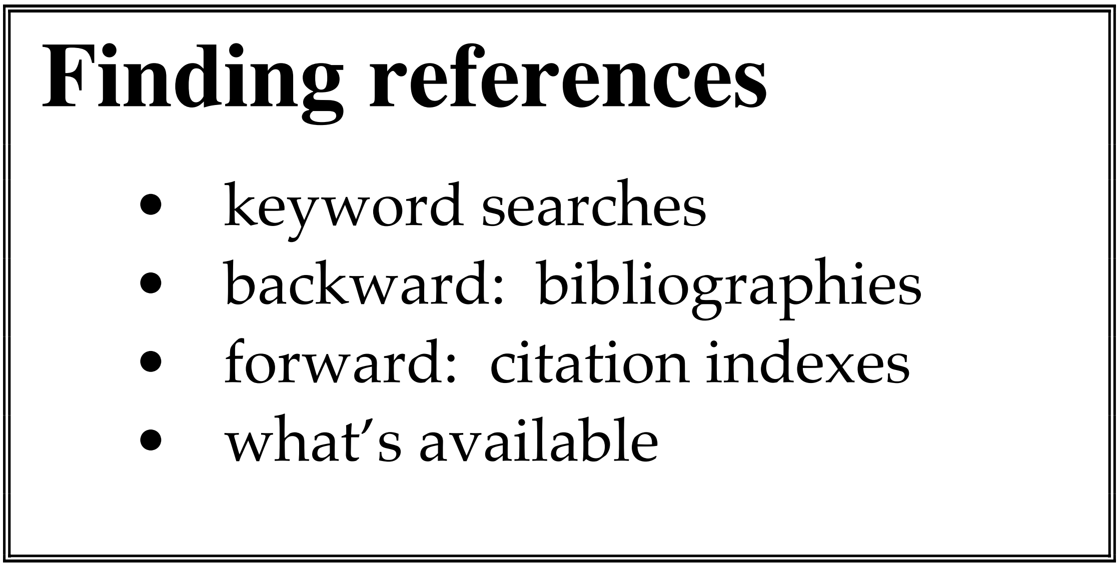
Other people's work

- what they write
  books, articles, manuals
- what they say
  interviews, discussion
- what they make
  software, organisations

You obviously need to read what other people have written: books, academic papers, newspaper articles etc. In addition you need to consider what they say by interviews, discussions etc. Finally, examine what they make — in your context primarily software, but also possibly the organisational structures, paper based systems etc.

2. Other people's work – what they write

You have already found out, or can find out, from the library staff how to use CD-ROMs and other information sources. So I will assume that you can handle that level literature searching. Instead, we shall look at some of the general issues.



Finding references

- keyword searches
- backward: bibliographies
- forward: citation indexes
- what's available

First of all you need to need to find relevant literature. You do not usually start from nothing. Either you have a few books or references to start with or at very least know what you are (initially) looking for. CD-ROMs and databases such as Inspec allow some sort of keyword based searches. That is you enter a few keywords such as 'parallel computing' and the database then finds and lists all the articles it knows about which have those keywords. Depending on he database this might find articles with the keyword in the title "Computing tidal drift using parallel rulers". Alternatively, the article may have a series of keywords associated with it and you find articles where these match your search. It is in fact quite a skilled procedure and the first time you use such a system you are likely to end up with either thousands of articles or none. Happily, most databases allow you to refine your search so you can start off with a very broad query and gradually hone it down to a manageable number.

There are two things to remember about such databases:

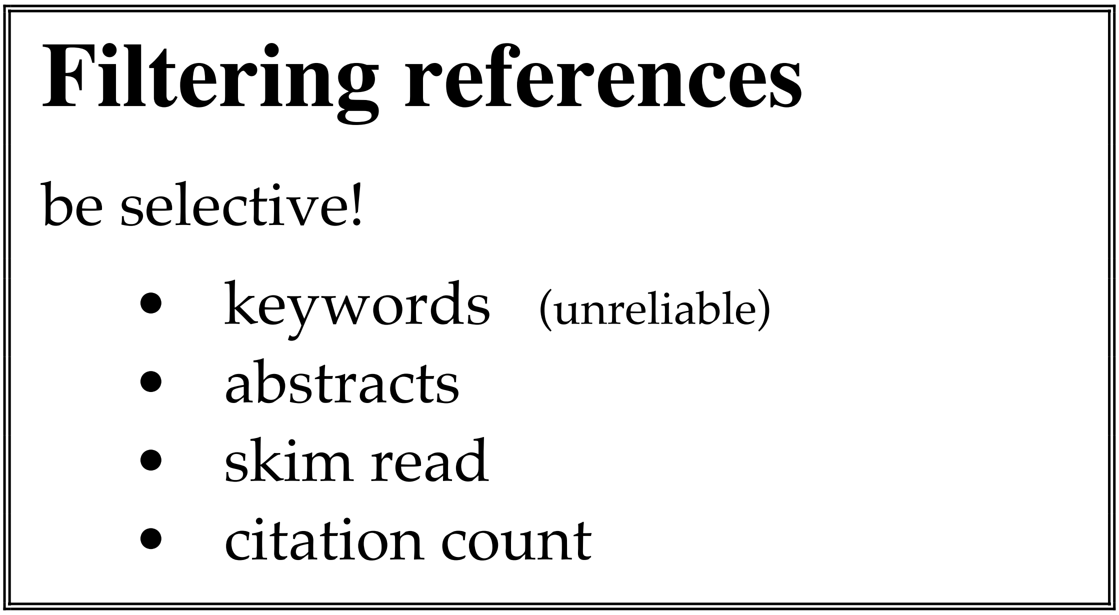
- They can only find articles that they know about. So, for example, some databases include large numbers of journal articles but no conference proceedings. So, just because the database doesn't know about something, it doesn't mean its not there!
- The keywords that are entered in database may have been generated by various people: the authors, a journal editor or the compiler of the database. If the keywords came from the original authors, then different authors may use the same words differently. On the other hand, the database compiler would use a uniform scheme, but it might be based on a cursory reading of the article.

For both problems you need to know what sources and methods the database uses.

Having obtained one or more initial articles or books, you can then spread your search. First of all you can move backwards in time using the article's bibliography. So you have a 1991 textbook on parallel computing and it references 'F. Bloggs 1983'. You look this up in the bibliography, order it from the library and then when you get it you find a reference to 'J. Smith, 1961', you obtain this and look in its bibliography etc. Notice how you get older and older articles. Obviously no article can reference a newer article than itself (at least only rarely).

If you want to look forward in time it is somewhat more difficult. In this case, you use a citation index. This tells you what other articles and books reference the one you know about. For example, imagine that 'J. Smith, 1961' is the seminal work on parallel computing. However, you want to find more recent work. You look up 'J. Smith, 1961' in the citation index and its tells you that F. Bloggs referenced 'J. Smith, 1961' in an article in 1973. You can then continue this process by looking up 'F. Bloggs 1983' in the index.

Finally, you can simply rely on what is easily to hand, what is in the library, on your supervisors shelves etc. If there are two books which cover the area you are after, one is in the library and one needs to be ordered and may take several weeks, then (at least at first) look at the one in the library. You will almost certainly find yourself with far too much to read and it is at least efficient to start with what you can easily get hold of.



Filtering references

be selective!

- keywords (unreliable)
- abstracts
- skim read
- citation count

If you use the processes described for finding references you will quickly end up with an enormous pile of books and photocopies which you will never get through, or which will take up too much time and divert you from other things. You need to be selective as well as broad in your reading. This is not so great a problem when chasing references backwards using bibliographies as the context in which the reference occurs tells you quite a bit bout it. However, citation indices and on-line databases give you less context and it is easy to end up with fairly useless papers.

First of all you can simply check that the title and keywords sound sensible. So having performed your CD-ROM search and found 20 references to parallel computing (actually it would be nearer 20,000) you don't just order them all, but check the details and thereby avoid ending up with an article on coastal navigation through inter-library loans! In addition, there may be an on-line abstract which can be read before the item is ordered.

Similar techniques apply when looking through books on the library shelves, or searching journals or conference proceedings for relevant articles. You can use the title, keywords and abstract as a guide, and then perhaps skim read papers which look promising.

Another crude way to reduce the mount of book and articles you order is to see how many times a particular paper is cited. If all the books you have cite one paper, then it is likely to be important, if only one or two cite it, then perhaps it is marginal. Not a very reliable heuristic, but its better than nothing!

Recording references

what

- details (title etc.)
- keywords (your own)
- mini abstract
- key points

where

- card index
- word processor file
- standard database
- bibliographic db

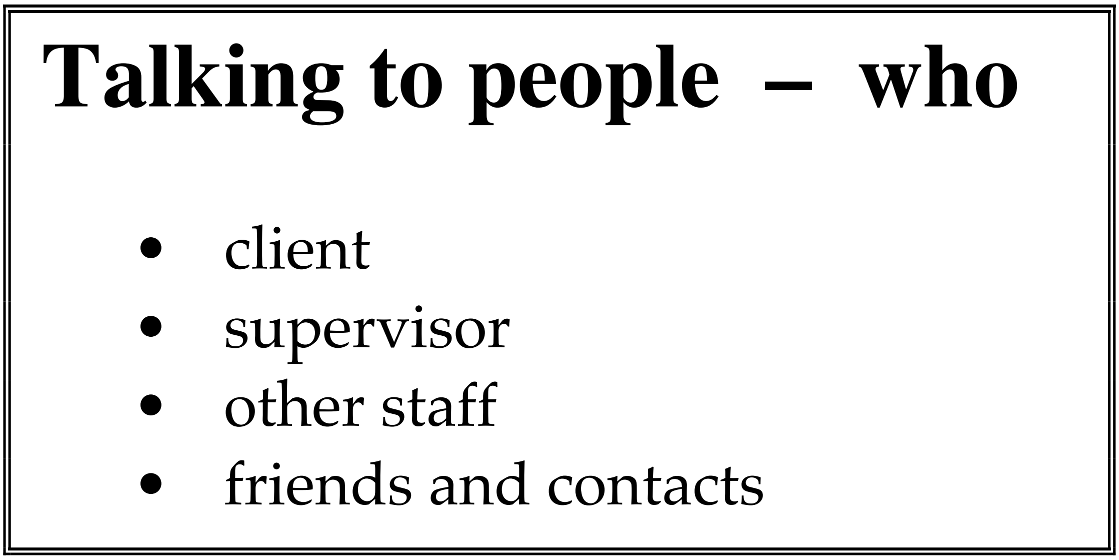
Having read something you need to record it. Books need to be returned to the library and you can guarantee that they will be out when you need them at writing up time. Even when you have a copy yourself it is surprising how difficult it will be next May to find the particular thing you remember reading about in November.

First of all, what should you record. Well obviously you need basic details: title, author(s), date, where it was (if in journal or collection), the publisher, perhaps the library class mark. I say obvious, but it is easy to leave out relevant information. You can't rely on having the article to hand when you write your report, so record at least enough information to be able put it in your bibliography. Be especially careful when you photocopy an article from a conference proceedings or chapter from a book: immediately write down on it here it came from. Sometimes the information is in the header or footer, but not always and it is so frustrating to have an article and have no idea where you got it from.

In addition to these basic details you can also record some information about the contents. You think you will remember everything now ... but in six months time. You can add keywords (ones that mean something to you, not necessarily the ones on the article itself). You could also add a mini-abstract in your own words - just a few sentences, and what were for you the key points. Note the difference. The abstract summarises the whole content of the paper, the key points extract just those features which were particularly relevant to you.

So, where do you put this information? Some people just keep a notebook and write things own as they read them. An alternative is to use 3"x5" filing cards (as beloved by law students). The advantage of these is that you can spread them out on a table, sort them into groups etc. However, this is probably advice for the extremely methodical! Given you are all computing students you might prefer an electronic equivalent. You can simply build up a word-processor file with the information in. You can then simply include bits of this into the bibliography of your report. Indeed one type of bibliography includes annotations about each paper. The database buffs could use a standard database package to set-up a bibliography. However, note that most databases are poor at handling unstructured fields such as abstracts. Also note that there are lots of different types of references (books, conference papers, journal articles, newspaper reports etc.) and they can be difficult to map onto a standard relational structure. The alternative is to use a special purpose bibliographic database, such as EndNote on the Macintosh. However, there are none available on the student machines, so you would have to buy your own! Those using TeX would of course use a BibTeX format 'database' and troff users (if any) 'refer'.

3. Other people's work – what they say



Talking to people – who

- client
- supervisor
- other staff
- friends and contacts

You will have access to a range of people during the course of your project. Of primary importance will be your clients. Although there may be 'objective' information about your clients' needs, a lot will depend on the conversations you have with them and also possibly your observation of them.

Whereas your client is your major source for domain knowledge, your supervisor will be your start point for academic study. However, the project is independent work, so don't expect your supervisor to know it all! When your supervisor cannot help another member of staff may. Note especially because of the integrative nature of the project, it is likely that you will need to obtain deep expertise from more than one area. Your supervisor and other members of staff are likely to have magazines, books or papers which can start you on your literature search. Finally, don't forget your friends who may be doing a project in a related area and who may have found useful references (and with whom you can discuss your problems in the bar) and other contacts, such as those from your placement year.

Talking to people – what

- you don’t know what you’re doing

professional — does it
academic — knows about it

This document was very hard to prepare. When I first thought about the topic I thought “I don’t know any research techniques” yes, sure I do research, but I haven’t the faintest idea how I go about it. No this is not due to particular stupidity on my behalf, but is common. You will all of heard of the centipede which, on being asked how he managed to walk with 100 legs, thought about it for a second and then immediately fell over. We do so many things without knowing how. When you ask people things you must remember that. If you ask your client what is wanted, or what is done at the moment, do not expect an accurate answer. You will have to do the probing which goes beneath the surface. One of the differences between a professional and an academic in an area is that the professional can do things very well, but the academic knows about doing them. It is insufficient to simply do the work of your project – that is make a good product. You must also know about it, understanding the choices you have made and relating it to wider issues.

4. Other people’s work – what they make

Academics write a lot: paper and books and lecture notes. The fruits of industrial knowledge are usually locked inside people’s experiences, or where written are usually confidential. However, the products they produce are an important source of knowledge. Your client’s request may be “I have a system that does such and such, but doesn’t do it very well, I want a better one”. In this case, the existing product may be one of your most important sources of domain information. In addition, you can look at similar products, both marketed ones and those produced as the product of previous research (including previous projects).

As mentioned earlier, there are other things that people make as well as software products: procedures, organisational structures, physical things as well. I will use the world ‘artefact’ to cover all these.

things people make

Artefacts Embody Experience

Whenever you consider a word-processor, database package, aeroplane or any other artefact, they all in some way embody the accumulated experience of their makers. This is particularly obvious in traditional hand crafts, but is also true of more industrialised products. Often this knowledge is unwritten and as discussed earlier, they may not be able to explain it when asked.

If you ignore the experience locked up inside existing artefacts, you are likely to make the same mistakes that the previous systems made or at best correct some, but make other things worse. To unlock this implicit knowledge you need to analyse the artefact.

- what is good about it?
  why is it good?
- what is bad about it?
  why is it bad?
- why do it this way?

First ascertain exactly which features are good or bad. Usually a system is not all bad! having done that ask yourself why. For example, given a database interface, you might decide that it is bad because there are some queries which it cannot perform, but that the graphical interface is pleasant to use. If there is a bad feature look twice at it. Why is it that way. Did the designers simply not notice or is it due to some other factor. It may be due to some constraint that no longer holds, or due to something that you might have otherwise not considered. Finding out why something is wrong is the first step to making it better.

Artefacts
Embody
Theories

e.g., mouse ⇒ hand/eye control better than typing

These why question are very important, and the deeper you ask, the more you understand the situation. Implicit in any system are the theories which either support or refute it. For example, we might say an interface is good because it uses a mouse. Why? Well we could consider that this is because hand/eye control is better than typing commands. Of course, we should then ask what we mean by 'better': faster, easier, less error prone ...

deep understanding helps

- combine ideas
  avoid the crocophants
- change context
  e.g., interfaces for the blind
- improve and correct

Deeper understanding really does help produce good designs. Imagine you are an animal designer. You notice that crocodiles are good because they have strong jaws and you notice that elephants are good because that are big and strong. In pursuit of a new better animal you put a crocodile's head on to an elephant's body – getting a crocophant, good for nothing. In computer systems this happens all the time, understanding why something is good or bad can avoid these crocophants.

Also, if you understand why something is good at a deep level, then you may be able to find something which achieves the same end in a different context. For example, you think that graphical user interfaces are good, but want to design an interface for the blind.

You ask yourself why a graphical interface is good. There are various reasons including the rapid feedback of your actions. So, you consider alternatives so that an unsighted user can experience the same rapid feedback, perhaps speaking back menu item as a mouse is moved.

In general, the deeper your understanding the greater your ability to focus on those area which need improvement and correction. Also the greater the chance that your ideas for improvement will work.

Artefacts
Embody
Assumptions

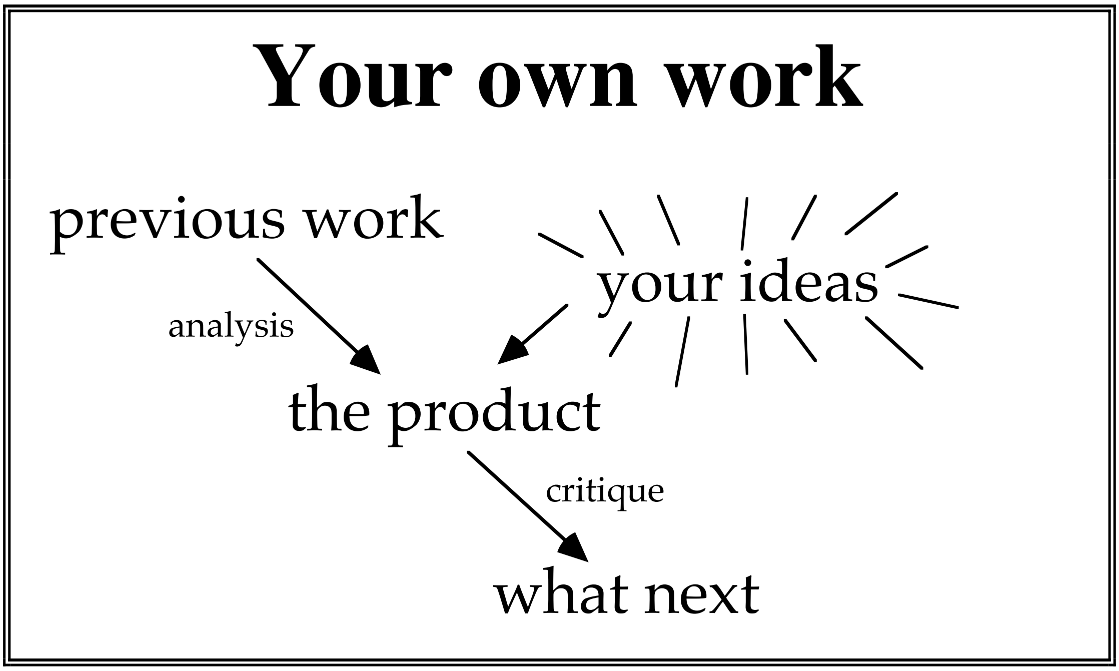
solutions depend on context
e.g., speed vs. space for algorithms

Finally, when considering existing systems remember that artefacts embody assumptions about the context in which they were developed. You might find an existing piece of software which appears to satisfy all your requirements. However, if it was designed with a different context in mind it may be inappropriate or need modification for your context.

One example of this is the trade-off between speed and memory requirements in computer algorithms. For example, in database system you might improve the speed of queries by storing lots of indices which consume a lot of disk space. Many current applications assume that RAM and disk space is cheap and so typical Windows, X and Mac applications consume many mega-bytes of memory. These may be good applications, but if you wanted an application to run on a hand-held computer then clearly the change in context requires a radical redesign.

Knowing that artefacts can tell you about the context can also help when analysing your problem domain. You ask why the existing system behaves in a particular way and the answer may uncover some aspect of the domain you had not previously noticed.

5. Your own work



Your own work

previous work

analysis

your ideas

the product

critique

what next

Having gathered all this information about other people' work, you need to analyse it, and apply it to our own problem and product. Somewhere along the line you will hopefully also bring in your own ideas.

However, your job does not stop there, you will also need to look back at your product critically. Part of your final report should detail where you feel the faults and limitations of your work lie and in what way the product could be improved.

Consider first the analysis stage. You could, of course, simply gather your background information, let it permeate into your brain and then start hacking. Whether or not you succeed, you won't get many marks! Instead you should aim for some sort of structured review of the existing work. This will enable you to relate it to your own product (in your report), but also should help you to develop a better product.

One way to analyse the existing work is to simply to classify it into areas. If you are developing a database interface for the motor industry, you may have studied some papers and books on HCI, some on databases, and some articles about cars. If you have used filing cards to record your references, you could actually sort them into piles, if you have used a database, you could enter this classification as an extra field and use it to sort your bibliography. However, unless you have vast numbers of references simple hand sorting will be fine.

A more sophisticated alternative is to build some sort of taxonomy, sub-dividing your classes into sub-classes, sub-sub-classes etc. For example, you might divide the database systems you have studied into object-oriented vs. relational.

Analyse existing work

- classify
- taxonomise
- multiple perspectives
- matrices

	single processor	multi-processor
non premeptive	ea or other	somat else
premeptiv e	thing amibob	whatsit

Beware however, of classifications (like those above) which follow too closely the standard assignment into computer science courses. You need to classify in a way which is pertinent to <u>your</u> project. One way to look for alternative classifications is to start off with the 'obvious' one and then force yourself to re-sort using some other criteria. In general, the more perspectives you take on a problem the more you will understand it.

If you have classified using several criteria then one way to organise your material is in a matrix. Use one classification along the vertical axis and one on the horizontal axis and simply place systems and papers in the relevant cells. If a cell is empty this might suggest that this is a particularly hard combination, or alternatively that it is a possible area to generate some fresh work.

Alternatively, you may notice that most of the cells are empty. This might suggest that there is some connection between the different classification criteria. The reason for this might become obvious as soon as you think about it, or you may find that you have discovered a previously undiscovered law (well it's worth being optimistic).

For the purposes of developing a new product, the most important feature of such a classification scheme is that it gives you a way of looking at <u>your</u> context. Consider your own product and where it ought to fit in the matrix. Is there an existing system/paper there? If so then it is a potential solution. If not (which is normally the case!) then you have to think a little harder. It may be that it is a particularly hard or even impossible situation, in which case, you need to consider ways of simplifying or altering the problem to make it soluble. More often it is simply that no-one else has done exactly the same thing before. In this case you look to previous work which share most of the same features as your own problem. In the simple two classification matrix this would mean simply looking to those systems in the same row or column as your own. Bringing features of these related systems together is a good first step to solving your problem ... watching out for crocophants on the way!

Understand your context

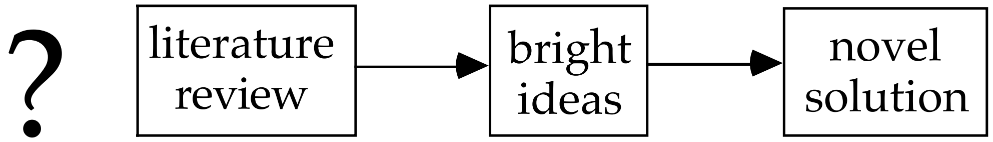
	single processor	multi-processor
non premeptive	eha or other	somat else
premeptive	thing amibob	✗

matching solution?

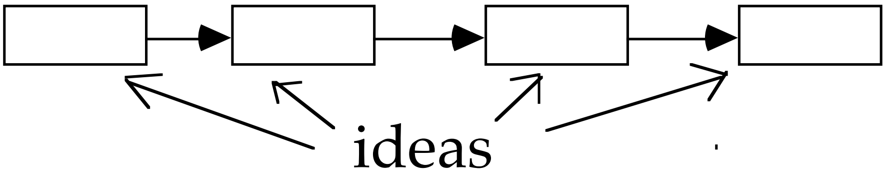
yes — then use it
no — synthesise

So much for using existing work to drive your own. What about all those bright ideas of your own. Well the first rule is don't rely on them. In the first Pert chart below, the bright ideas are on the critical path. This is a risky strategy. If you don't get the ideas you can't make the product and you will fail your project! However many bright ideas you normally get, don't rely on them coming when you need them.

Planning for innovation



Don't rely on it!



However, the other extreme is to take a risk-free strategy. This is usually a minimax plan, which minimises your maximum losses. Unfortunately minimax plans for losses tend to be also minimax for gains. If there is no room for your own ideas in your project plan, then you can get by, but are not going to get really good marks. The ideal plan looks like the second pert chart. A risk free strategy which is still open to using your ideas when they come. A tall order, but not impossible.

Finding new ideas

- abstraction
- analogy
- lateral thinking
- challenge yourself
- silly ideas

e.g., the munchman interface

So, now the hard part, how do you come up with all these bright ideas to feed into your work. Of course, there are no easy answers otherwise they wouldn't be bright ideas. However, there are some heuristics which can help. One thing to do is to abstract your problem. Look at it and think of a more general problem of which it is an example. Once you have done this you can see other problems which are related to it. You can then take solutions to the related problem and apply them to your problem. This process of abstraction and analogy is typical of many academic disciplines, but can also be intensely practical.

As an example of this I recently published a paper which related a range of problems including burning the toast and waiting for email replies. The common theme was that there was some event (the toast being cooked, or the reply arriving) which happened after some delay from the thing you do to start the process (put the bread under the grill , or sending the original message). Seeing the common general problem enabled me to consider general solutions: aide memoir, electronic reminders etc. and then to apply the analysis to disparate case studies: air traffic control and car radios.

De Bono's lateral thinking is one of the accepted, but hard to implement, pieces of advice for thinking up novel ideas. The aim is to think in a way which is different from one's normal thought patterns. You can read De Bono's books for his advice on this, but note also that we have already discussed several ways to encourage this. For example, the search for different classification schemes was a way of forcing different perspectives. Also asking very deep 'why' questions is one way of pushing you away from the standard solutions.

Another way to encourage new ideas is to constantly challenge yourself. You have had one idea, OK so what's wrong with it. Even if you think it is a good idea take it apart and see what's good and bad about it. In doing this you will be forced to think again. Imagine you are going to be given £10 for each fault you find. Then argue back to yourself.

It is hard to be suitably critical with your own ideas because they are your ideas and you like them! You obviously need to believe in what you are doing, but the emotional commitment to your ideas can prevent you from seeing the problems and hence find better ones. One way to get round this is to think up silly ideas. The advantage of silly ideas is that you don’t intend them to be good ideas and so can feel happy criticising them. However, you can also look for good points. Just as you did with other people’s work, take the silly idea and pretend that it is serious. Why is it good, why is it bad? The very silliness of it can help to make new connections and hence inspire new ideas.

Imagine a world famous expert on computer networks (now in his late 70s) has come to give a lecture. He is the inventor of the Internet, the World Wide Web was his brainchild and it is widely rumoured that Novell employed him as chief consultant. Despite his age he is still enthusiastic in his presentations and producing exciting new work. he stands up to speak: “Ladies and gentlemen, the problem with computer networks today is that they are like spaghetti bolognese without the Parmesan”. As he speaks his excitement rises and then with the word “Parmesan” he suddenly clutches his chest and expires. The hall is shocked and when the doctor arrives he is pronounced dead. However, after the shock of his death subsides the audience begins to ponder these final words of wisdom from the master.

Think about it yourself, he said like spaghetti bolognese, not lasagne mind you, so he was obviously not thinking about OSI layers! Think of your own problem — is it like spaghetti bolognese without the parmesan. Think of a reason why it is. Now think of our own silly idea. They do not have to be quite a silly as that. I once was worrying about the problem of type-ahead and how you could tell whether the computer still had keystrokes to process. I suggested (as a silly idea) that the characters could appear a the bottom of the screen as they were typed and then a munchman could gobble them up as the application processed them. A silly idea, but capturing some good points.

Evaluation

- testing
- simulation
- proof
- statistical
- anecdotal

You now have had your ideas and produced your product. The job is not over. The time has come to evaluate your product. This is clearly important from the client’s perspective — does it do its job? However, you can also consider the whole project as a bit like a white coated scientist’s experiment. In doing the experiment (making the product) you have tested your ideas and will then know more about them. Of course, without effective evaluation you won’t know how good you were this time and so won’t be able to do better next time. Also unless you evaluate your product effectively you won’t get good marks.

Depending on the kind of product, various forms of evaluation might be appropriate. You might use test data or possibly run some sort of simulation to test your system. You might even attempt some sort of proof that it works. This proof might consist of a careful argument, or might even be a formal mathematical proof. In some cases, some statistical experiments or tests might be used, possibly in conjunction with simulation data, or data collected about use. Finally, do not underestimate anecdotal evidence, the comments of your client on seeing the product, your own experiences etc. In many areas it is impossible to obtain reliable objective evidence and anecdotal evidence is likely to be more useful, especially when considering future action.

Critique – the good

good points

- say what they are!
- what is novel, interesting?
- how does it relate to theory?

So, we have come full circle, from analysing other peoples work to producing your own. Now, you can use the same techniques to analyse your own work as you used with other people's work. However, you have the added job of presenting your work in the best light to your examiner. This is bit like a marketing exercise, but with some important differences.

First of all, consider the good points. Identify them and report them. Tell the reader of your report what is good and why. Your product may be excellent, but you ought to be able to say why and relate your success (or failure) to the theory you used. If the examiner has to work this out then he or she is doing your work and you will be marked accordingly. If you don't say what's good, how does the examiner know it is not simply a fluke?

Equally important is to analyse the bad points. As I've said before, this is an important part of your project report. If your product has faults which your examiner notices, but you have not mentioned, then you will lose marks. If, on the other hand, you report any weaknesses and analyse why they happened, how it could be dealt with and what you have learned from the experience, then you are likely to gain marks.

Assuming you have identified the bad points in your product (and also in your project management etc.) you will need to examine the reasons for them. More why questions. It may be that you were limited by the available resources: software or hardware. Of course, in a commercial project you would be resource limited, so this is no excuse for sloppy planning! Your most critical resource will be your time and it may be that within the time available you can only produce a reduced form of the product you would like to have developed. This is something you should be addressing during the first semester as you produce your project plan and specify the scope of your project. However, you will almost certainly find that things do not go entirely to plan.

Critique – the bad

what is wrong and why

- resource limitations
- ran out of time
- lack of experience
- hindsight

It may be that you simply lacked experience in some area and so things went wrong. Hopefully, your supervisor will be able to spot some of these problems before they happen, or they may become apparent as part of the first semester assessment. However, the project is your own independent work and as such you will make some mistakes. Remember that the project is part of your degree course and is a learning experience, you are not expected to know everything before you begin. If something went wrong because of lack of experience and you have identified the problem — then that is evidence of learning and will be to your credit.

Finally, there may be factors completely beyond your control. You may choose to use some software package which purports to perform all the functions you require, but half way through your work you find it is not up to the job. Faced with such difficulties you are expected to cope as well as you can: find workarounds, perhaps modify your aims. The important thing is again that you identify these problems and describe your methods of dealing with them.

6. And finally

This is an undergraduate project. The amount and nature of research which you carry out and the level of innovation in your product should be commensurate with that.

Having produced and assessed your product you will have built up a body of practical experience which should complement that obtained during your work placement. In addition, your reflection upon that process and the research you do towards it will build on your academic skills which will also help you in your future career.