# Some thoughts on interface design

**Harry Powell**

MRC Laboratory of Molecular Biology

23rd August 2013

# Before we start

Look at your smartphone/tablet/laptop/PC - which apps/applications do you

- enjoy using?
- even *think* about when you use them?
- use *in spite of* some awkwardness?
- think "why do I have to do that" about?
- use in preference to other (equivalent) software?
- is it "intuitive"? what does that mean?

# Interface design is a *huge* topic

Some thoughts

- Is a gui even necessary?
- Identify your users
- Address the goals, not the tasks
- The computer should do the work, and the person should do the thinking
- Avoid introducing excise
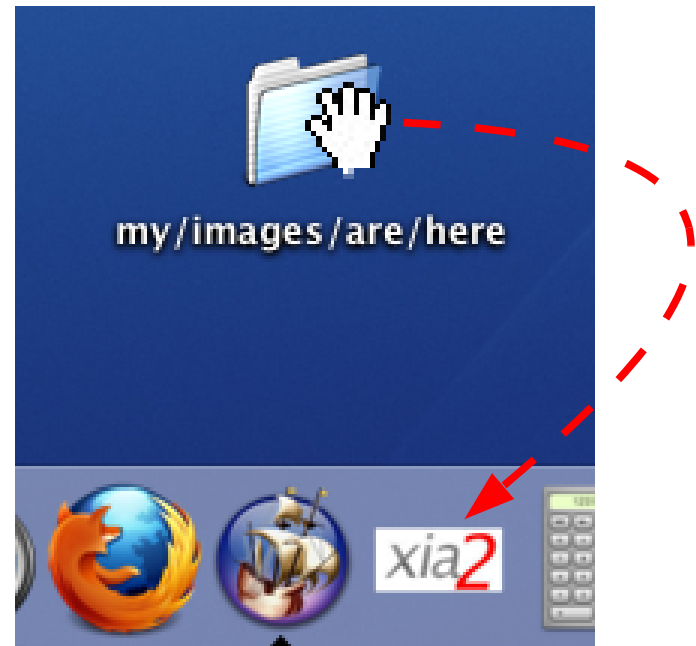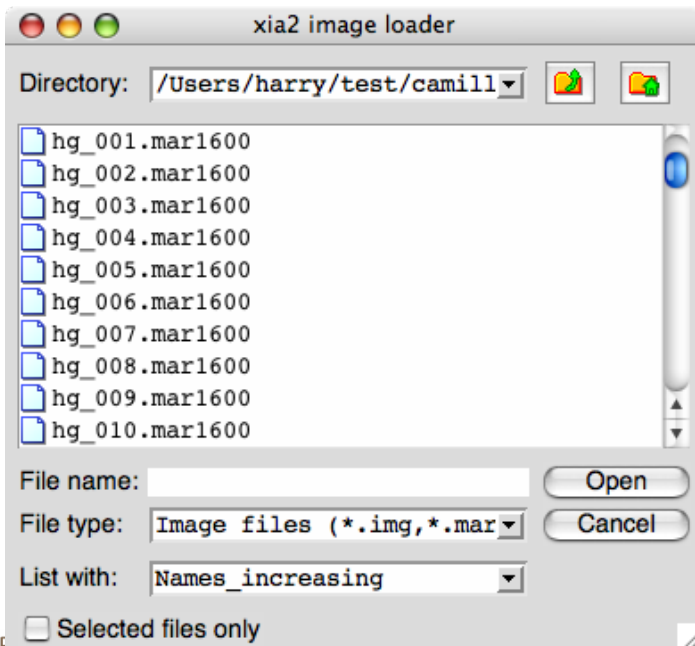
# Technical considerations

- Do we need a GUI?
- Choice of programming language
- Communication between GUI and application
- Who does the programming?

# Is a GUI even necessary?

*e.g.* typing

- `xia2 /my/data/are/here`

will process the images in the directory /my/data/are/here - possibly easier than -

# Choice of language/toolkit

Can be almost irrelevant, but -

- obsolete toolkits are often very stable but
  - may be inflexible
  - may not be transportable
- is it available for (all) your intended platforms?
  - QT (pyQT, Qt/C++...)?
  - Tk (Tcl/Tk, tkinter...)
  - wxWidgets
  - Java
  - Aqua, WinAPI...
- today's "flavour of the month" will be obsolete one day
- GUI design software is probably not a good answer

# Interaction with application

Choices

- GUI and application access the same variables - essentially a monolithic application (*ipmosflm*)
- GUI writes commands and reads output file(s) (*ccp4i*)
- GUI communicates more directly, *e.g. via* sockets (*iMosflm*)
- *e.g.* Application writes html (*via* PHP, Perl, Javascript *etc*.) to produce a web interface

Job for the interface programmer, *not* for the interface designer (these may be the same person!)

# GUIs and applications

A GUI *might* be an afterthought, but is probably the way most people use the application - so it is important

- a poor application will not be improved by adding a GUI

It is best to separate

- GUI design
- Application programming

since they address different problems, but GUI designers and application programmers should work together

Application programmers may have to modify their code substantially for the benefit of the GUI (particularly i/o).

# Application programming *vs* GUI design

An application programmer should know about
- the science and algorithms that they are implementing - the tasks
- what input and output is important

A GUI designer needs to know
- about users
- absolutely nothing about the algorithms
- something about the science can be helpful
- which goals are important
- what input and output is important

# Who will use my GUI?

Basically three groups -

optimise for these

proportion of users

novices

perpetual
intermediates
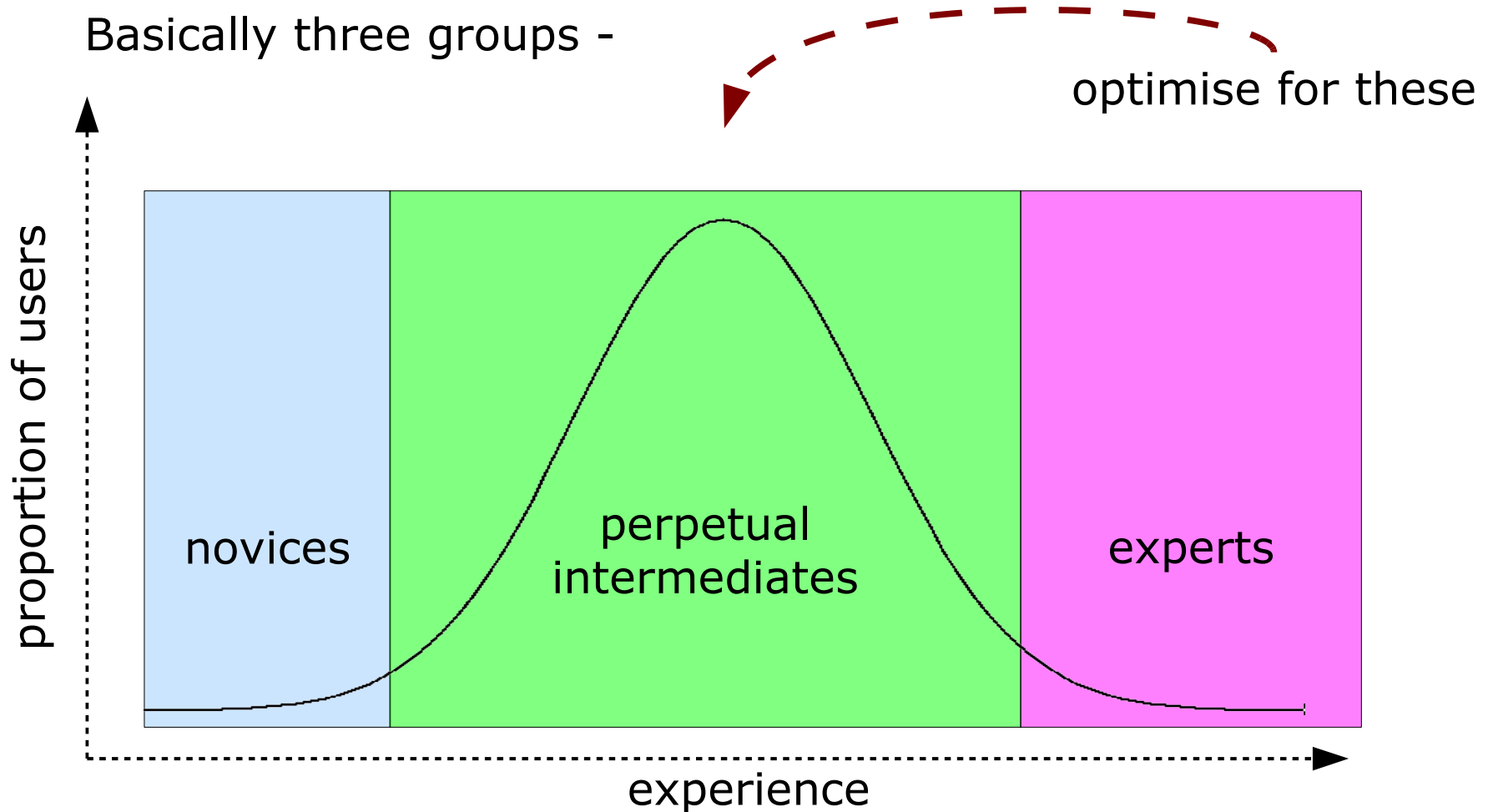
experts

experience

# Novices & experts

- Novices
    - no-one wants to remain one
    - they are intelligent but busy
    - need to know how to operate the product, do *not* need to know how it works
    - will become intermediates if they continue using the product
- Experts
    - Influence novices disproportionately
    - Are trusted by other users
    - need access to "rarely used" functions

These are not typical users

# Intermediates

- Majority of users
- You want your novices to be in this group as soon as possible
- They are not all the same -
  - design for specific user types with specific needs
  - if you try to please everyone equally, they will probably all be dissatisfied

# Caveat venditor! You are not typical

You may be an expert in the use of your software - most of your users are not

- some users *will* be experts, and some will be novices, but you cannot design a GUI primarily for these groups

Find out about what your users do, how they work, etc etc

- ask them
- observe them using
  - previous versions
  - other software to achieve the same goal

Don't base your GUI on *your* needs/wants

# Goal directed design

Users are interested in goals, *e.g.*

- "I want to integrate these images"
- "I want to analyse the active site in this enzyme"

Applications programmers are interested in tasks, *e.g.*

- sort a list of values
- calculate an FFT
- minimise this value *via* the right target function

GUI designers need to bridge this gap

- the user sees the skin of the GUI (goals)
- the body of the GUI connects the goals with the tasks

# Human thinks, computer works

*e.g.* User chooses the images to process, then the program

- reads & interprets all the image headers
- sets processing parameters accordingly
- presents values/output clearly
- updates available actions
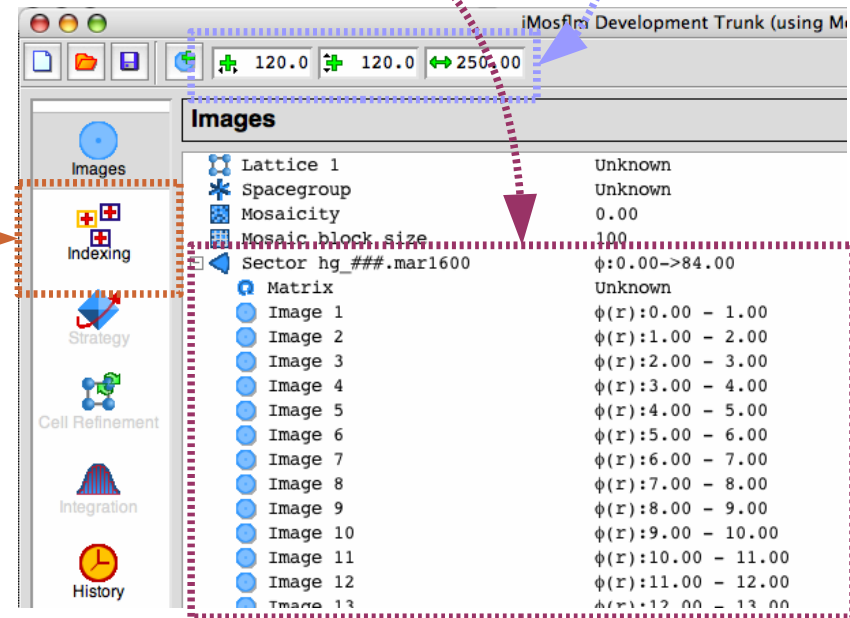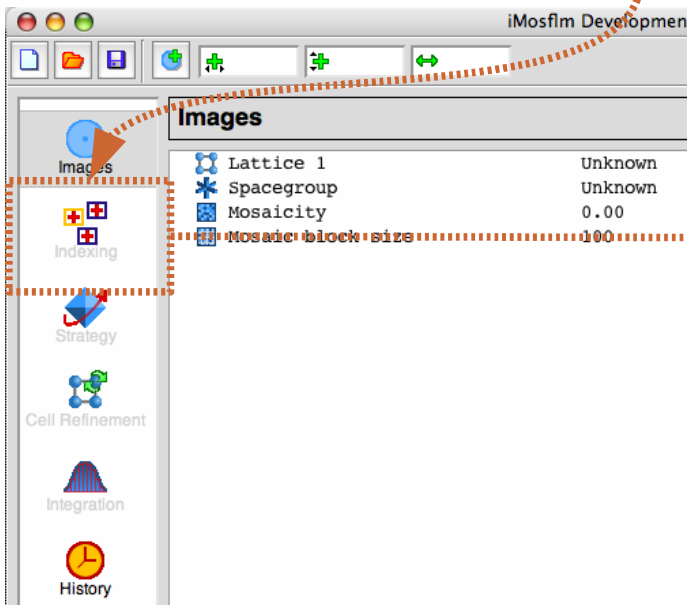
# Main windows

- Should be the first window to appear
- Because they are persistently visible, their appearance should not be jarring
    - relatively neutral appearance
- Navigation between the individual tasks should be straightforward - "intuitive"
- Normal range of major actions
- Typical user is an intermediate
- Access to "advanced" features should be fairly obvious
- Have "sovereign posture"

MOSFLM version 7.0.7 (Unsupported Image Display)

| Edits allowed |
|---|

### Processing params

| | | |
|---|---|---|
| a | : | 0.00 |
| b | : | 0.00 |
| c | : | 0.00 |
| alpha | : | 0.00 |
| beta | : | 0.00 |
| gamma | : | 0.00 |
| PsiX | : | 0.00 |
| PsiY | : | 0.00 |
| PsiZ | : | 0.00 |
| Mosaic | : | 0.000 |
| Divh | : | 0.000 |
| Divv | : | 0.000 |
| Lambda | : | 0.880 |
| Distance: | | 220.00 |
| Beam X | : | 149.66 |
| Y | : | 149.52 |
| CCOMEGA | : | 0.000 |
| ROFF | : | 0.00 |
| TOFF | : | 0.00 |
| YSCAL | : | 1.0000 |
| Pick area: X: | | 11 |
| Y: | | 11 |
| Int threshold: | | 20 |
| Vector scale | | 1 |
| Two theta | | 0.00 |
| Resolution | | 0.00 |

*SPOT SEARCH*

| | |
|---|---|
| Threshold | 5.00 |
| Rmin | 1.95 |
| Rmax | 135.00 |
| X offset | 0.00 |
| Y offset | 0.00 |
| Min X size | 0.50 |
| Max X size | 10.00 |
| Min Y size | 0.50 |
| Max Y size | 10.00 |
| Min no of pix | 6 |
| X splitting | 0.00 |
| Y splitting | 0.00 |

*AUTOINDEXING*

| | |
|---|---|
| Min I/sig(I): | 20 |

Prompts [On]
Update display:
After refinement [No]
After integration [No]

Timeout mode [Off]

| Select item |
|---|

## Main menu

- Read image
- Find spots
- Edit spots
- Clear spots
- Select images
- Autoindex
- Estimate mosaicity
- Predict
- Clear prediction
- Adjust
- Refine cell
- Integrate
- Strategy
- Keyword input
- Find hkl
- Pick
- Measure cell
- Circles
- Beam / mask areas
- Save/Exit

3a2_1_003.image

Min 0     Max 18598     ax1=473, ax2=1021
Overlay [▽] on          Contrast ━━━━━━━●─
Colour [▽] Black on white    Mag [▽] x4   [PS] [Zoom]

### Output

| | | |
|---|---|---|
| Pixel X,Y | 1528 | 1021 |
| XC,YC mm | 229.0 | 153.2 |
| Resolution | | 2.55 |
| Indices | 0   0 | 0 |
| Intensity | | 0 |
| Sigma | | 0 |
| Spacing A | | 0.000 |
| Average | | 4388.0 |
| Rms | | 0.0 |
| Number | | 0 |
| Zoomfactor | | 0 |
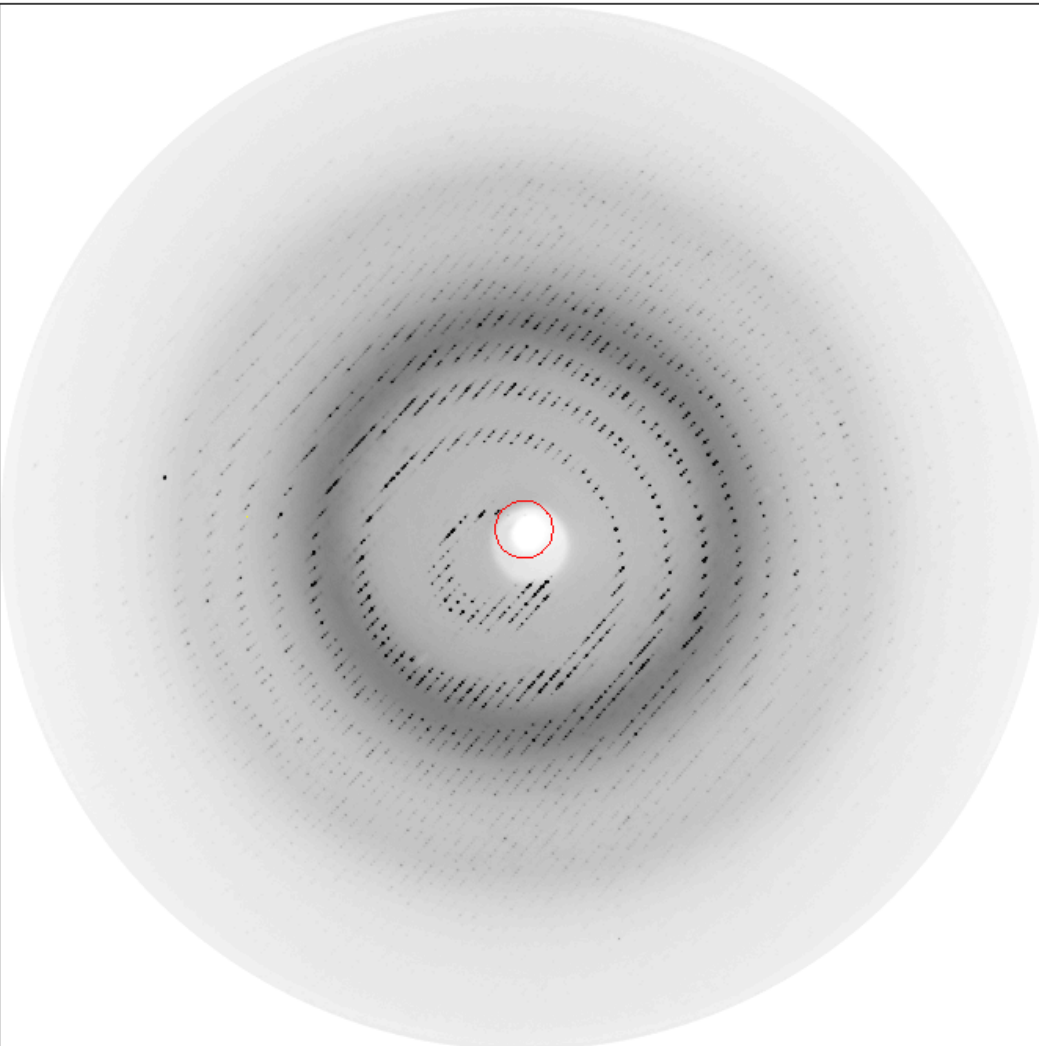
Circle resolution A
  0.0  0.0  0.0  0.0
Phi       2.00    3.50
Missets ThetaX,Y,Z
  0.00   0.00   0.00

# iMosflm Development Trunk (using Mosflm 7.1.0)

## Images

| | | |
|---|---|---|
| Lattice 1 | Unknown | |
| Spacegroup | Unknown | |
| Mosaicity | 0.00 | |
| Mosaic block size | 100 | |

Images

Indexing

Strategy

Cell Refinement

Integration

History

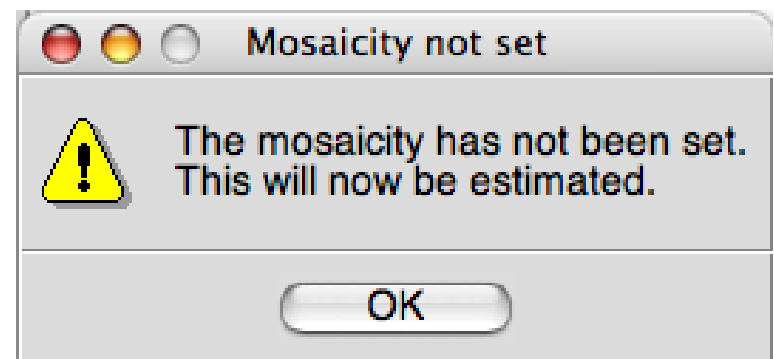Status warnings 0

# Pop-ups, drop-downs & other transients

- Temporarily visible
  - maybe less neutral appearance?
- Transient dialogues
  - for advanced intermediates and experts
  - support sovereign applications
  - enhance functionality
  - more complex options
  - rarely used options
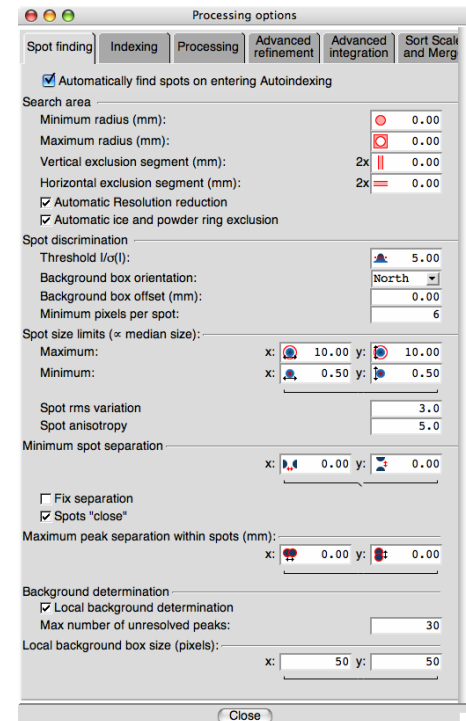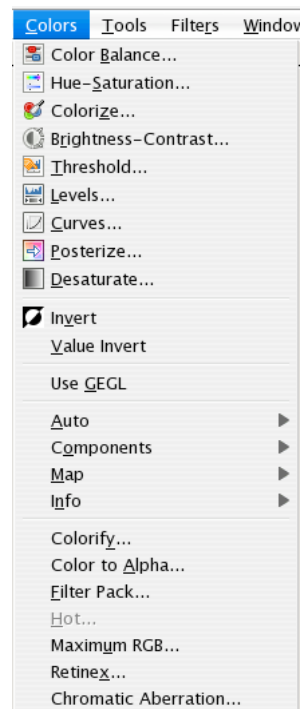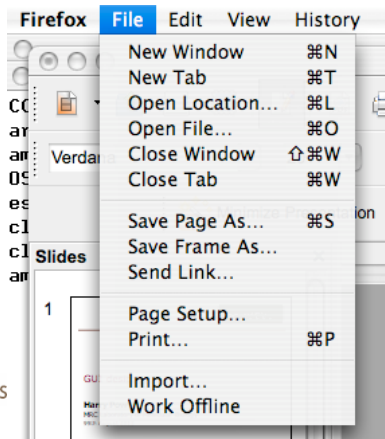
# Avoid adding excise

Actions which are necessary to complete a task but do not contribute directly to its progress *e.g.* the program
- encounters an error
- *knows* what the error is
- *tells* you what the error is
- *knows* how to fix it
- won't let you do anything else until the error is fixed
- *still* makes you push a button

# More excise... 7 clicks to the main window
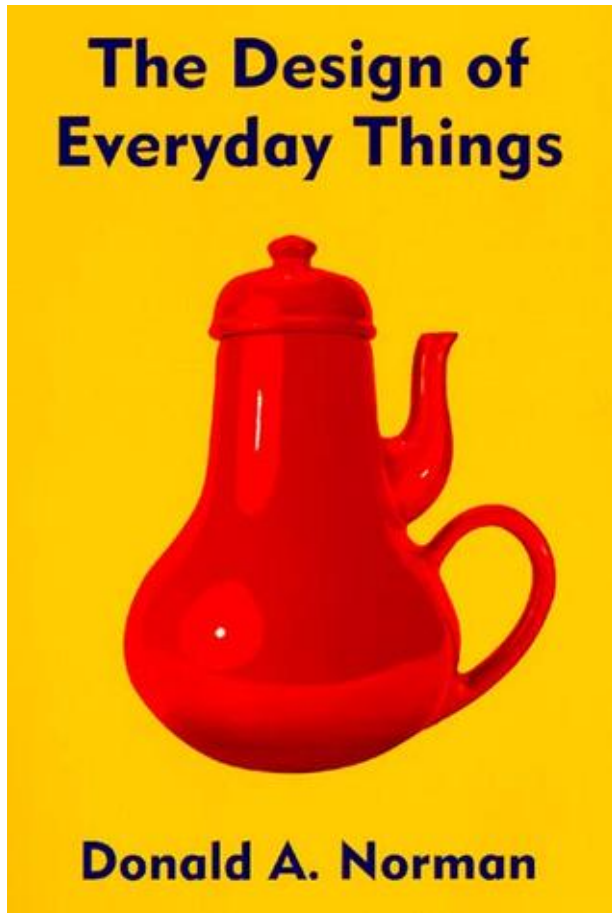
# Serious about GUI design…
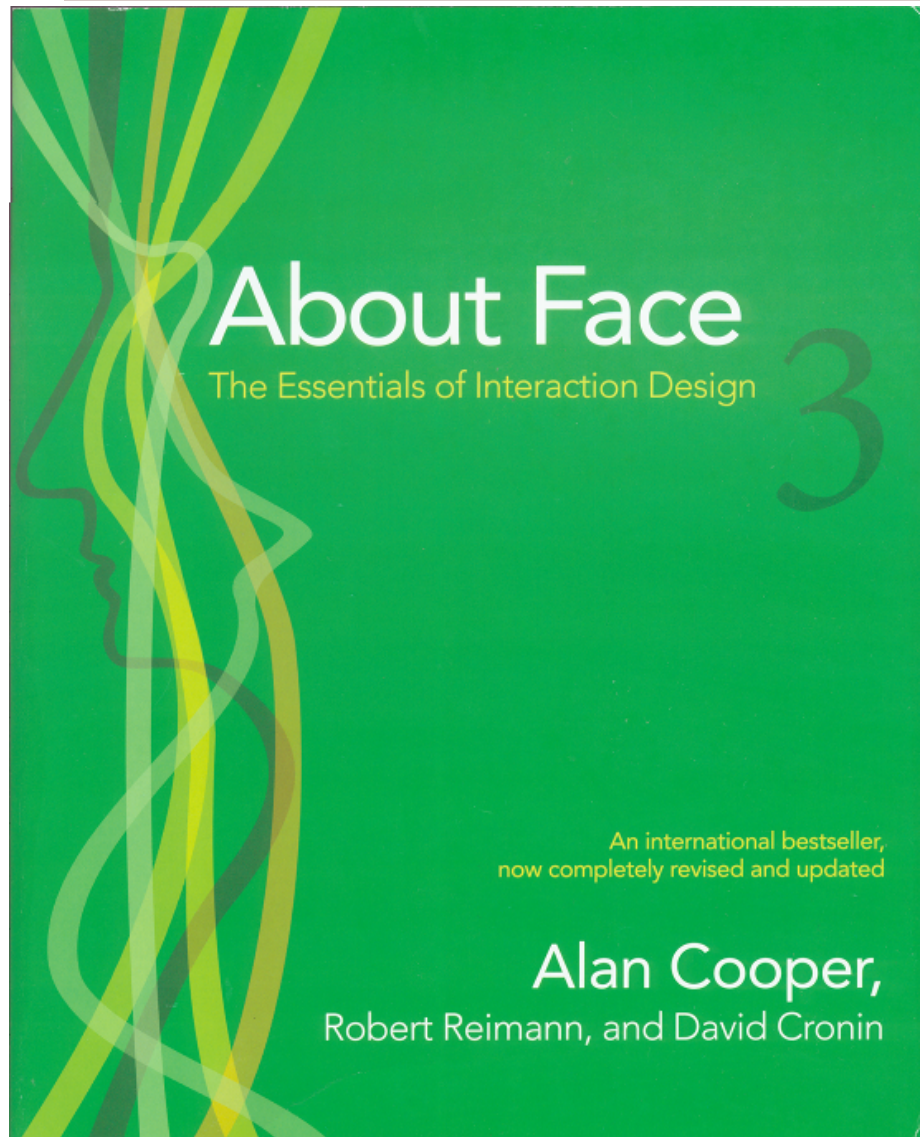


Caution -

- Contains strong views
- Essential reading for the serious interface designer
- It will tell you that your current interface is probably all wrong…

ISBN 978-0-470-08411-3
currently ~€35

# or just want to learn a little?



A classic in the field of the psychology of design

ISBN 978-0-465-06710-7

currently ~€10

# In summary

- Is a gui even necessary?
- Identify your users
- Address the goals, not the tasks
- The computer should do the work, and the person should do the thinking
- Avoid introducing excise