

# What to look for in a timetabling program

## Screens

The program should:

- be intuitive
- have uncrowded, friendly screens
- use colour to help you to identify items



## Help & Support

The program should:

- have a Help button to give immediate and interactive **Help** on each screen
- have **HelpMovies** (video tutorials) to help you to get up to speed quickly
- have a QuickStart Guide
- have a fully-illustrated Manual
  - which contains **Worked Examples** to cover **all** timetabling problems
  - including complicated 'Sixth Form' patterns
- have a 24/7 Support Centre giving expert timetabling support
- have a dedicated HelpDesk, staffed by **experienced** timetablers



## Scheduling

The program should:

- emulate the usual manual timetabling methods
- be designed by experienced timetablers
- be able to schedule **any** curricular structure, of **any** complexity
- allow quick entry of data,
- **prioritise** for you the lessons to be scheduled, and **show you** the next step
  - using a mixture of algorithms,
  - including heuristic ones, developed from long timetabling experience
- find '**musical-chairs**' solutions, with up to 16-step moves, of a quality that you can specify, and customizable
- include features to obtain not just a 100% solution but ones with good **quality** (good lesson-spread, etc)
- with methods that can be interactive, or semi-automatic, or fully-automatic
  - allowing the User to move between these methods easily, at any time
- find solutions quickly, display them clearly, and allow them to be modified easily



## Printing and Exporting

The program should:

- give 'individual' and 'master' printouts of staff, class and room timetables,
  - with the layout customisable to any of billions of permutations
- export the timetables as HTML web-pages for the school web-site
- export the completed 100% solution to an MIS such as SIMS .net etc
- export using a recommended XML method



# What to look for in an Options program

## Screens

The program should:

- be intuitive
- have uncrowded, friendly screens
- use colour to help you to identify items



## Help & Support

The program should:

- have a Help button to give immediate and interactive Help on each screen
- have a fully-illustrated Manual, with Worked Examples
- have a dedicated HelpDesk, staffed by experienced timetablers



## Entering and Analysing the data

The program should:

- allow quick entry or import of students' Names
- allow the students' Choices to be entered quickly (eg. 5 min per registration group)
- include a Reserve choice if you wish
- check the data against Rules that you have set up  
eg1. You must choose a Language. eg2. If you choose HSC1 then you must choose HSC2 as well.
- provide an analysis (clash table) of the students' choices



## Building a Options Pattern

The program should:

- allow you to set up Rules to be applied  
eg1. Art and Graphics must (or must not) be in the same block.  
eg2. HSC1 and HSC2 is a double option and groups must be populated consistently.
- allow you to design a Pattern of Options Blocks,
  - **manually**, or
  - **automatically** (3 different modes),
  - or a mixture of these, always following the Rules that you set up,
- show group sizes, and report on groups which are too big or too small
- show students' allocations, missing students, satisfaction rates, etc
- allow you to manipulate the blocks and groups easily,
  - with tools to 'equalise', 'shuffle', 'bunch', 'split', etc.
- allow easy addition/deletion of students who arrive/leave later



## Printing and Exporting

The program should:

- give printouts of the blocks, groups, students fitted, students unfitted, etc
- export to HTML, Excel, Word
- print Individual Student Slips, Individual Student Timetables
- print Group Lists for teachers for the start of term



# What to look for in an Cover program

StaffCover  
nearest other

## Screens

The program should:

- be intuitive, with uncrowded friendly screens, using colour



## Help & Support

The program should:

- have a Help button to give immediate and interactive Help on each screen
- have a fully-illustrated Manual, with Examples, and a dedicated HelpDesk



## Entering the data

The program should:

- allow the data to be imported from **TimeTabler**
- allow Absence to be entered quickly, both in advance and at the last minute
- allow events (eg. exams needing cover) to be entered easily
- also allow Supply staff (both permanent and temporary) to be used



## Assigning Cover

The program should:

- organise Cover automatically, taking into account up to 19 parameters  
eg. is it preferable for a Science class to be Covered by a Science teacher, for safety in the lab.
- balance the Cover allocated,
  - accurately and fairly,
  - depending on individual teaching loads,
  - and how recently a person has done Cover,
  - and allowing for 'PPA' time and workforce agreement rules,



## Statistics

The program should:

- store all Statistics (of absence, reasons, cover done, etc),
- for listing or displaying graphically, and printing



## Other features

The program should:

- allow classes to be suspended (eg. during exam time and revision days)
- allow 'Special Duties' slips for extra duties (eg. 'bus duty', etc)
- allow Teachers (and Classes) to be quickly located even though doing Cover



## Printing and Exporting

The program should:

- print 'master' Cover timetable printouts for the staffroom noticeboard, etc,
- with a copy emailed to named senior staff
- print Individual Cover Slips, for handing out to individual staff, and
- email copies to each teacher who is doing Cover

