

# Suits you sir!

**Most computer software has features that allow off-the-peg packages to be tailored for a best fit. Steve Lee explains**

**T**he fact that computer software can be customised has great potential in the accessibility arena where an individual's needs may not be catered for by existing software. A little effort in tuning can reap rewards in usability and productivity.

For example, an appeal in the pages of *Ability* for help in giving a Microsoft PowerPoint presentation by Ross Gravell, Ford's disability officer, led to my developing a program called Power Talk.

The challenge was that Gravell had dysphasia and the audience he would present to included partially sighted people. I opted for a solution that involved having the computer automatically speak text during the presentation.

PowerPoint and Microsoft's SAPI speech system use a technology that allows them to be controlled by other programs (called automation). This can be achieved with a small amount of 'plumbing' program code, the code listens to messages that PowerPoint broadcasts when new text appears and passes the text on to SAPI to speak.

I decided to write the code using the Python programming language which provides simple, clear code with a quick turnaround. However, I soon discovered that PowerPoint presented problems that blocked all progress, but because Python is open source (see sidebar), I was able to draw on a community of developers to provide help.

The developer of the features provided a fix and PowerTalk was ready. PowerTalk was then completed with the addition of an

installer and given an open source licence to ensure it could be adapted by others.

The techniques used to create PowerTalk are fairly advanced and require some programming experience, but much simpler means also exist. In fact there is a spectrum of techniques for customising programs and operating systems:

- Instructions that list actions for a user to follow
- Skins that provide an alternative appearance
- Plug-ins to extend functionality
- Settings that change the behaviour of text or the user interface
- Data declarations that define the behaviour and appearance of documents
- Macros for recording user action sequences and playing back and editing them
- Scripts – text programs that run within a host program or the operating system
- Collections of programs called bundles
- The automated generation of programs from components and options
- Builds that enable an original program to be created from source code

PowerTalk demonstrates how a combination of these can be applied and how a specific solution can be generalised into a product that can then itself be further customised.

Once any customisation is applied to base software it becomes necessary to provide support. This creates problems because user

needs change and the user may become reliant on the new features.

Changes may fail when applied to an unknown system configuration and there is the need to merge official updates of the underlying software. The creation of an installation program or script eases deployment to other systems. Backups are needed to allow the undoing or reapplication of the changes.

As a final thought an interesting possibility is to provide software modifications as part of an existing service that provides AT assessment, selection and customisation. □

PowerTalk may found at [www.fullmeasure.co.uk/powertalk](http://www.fullmeasure.co.uk/powertalk)

## The open source alternative

Free and open source software (FOSS) provides a proven alternative model of software ownership, licensing and development. The idea is that software can rapidly improve if the program source code is freely available. This contrasts with the proprietary model which protects ownership of the program by hiding the source code.

Contributions from people at all levels of skill and experience are actively encouraged. Larger FOSS projects are developed by global on-line communities, which provide rapid development and ready support.

Cost of ownership of FOSS is often low, with developers generating revenue by means other than charging for the program. For the user, FOSS can mean high quality software at a good price and with excellent support and quick fixes. To the developer, FOSS can mean having mature initial code to work with and a large pool of contributors.

See [www.opensource.org](http://www.opensource.org), [www.gnu.org](http://www.gnu.org) and [www.affs.org.uk](http://www.affs.org.uk) for more information.