



Scottish Arrestment Orders - a Developer's Perspective

Elsewhere in this issue of Payroll Professional, you'll read about the changes to Scottish attachment orders which were recently announced at very short notice and subsequently withdrawn after a clamour of protest from the IPP and others.

The amended regulations, made public on the 13th of March with an implementation date of the 6th of April, would have made a fundamental change to the calculation of deductions under an Earnings Arrestment.

One of the objections was that many software developers would have been unable to make updated software available in time. Employers would therefore have faced the prospect of calculating these deductions manually until updated software became available.

We thought it might be useful to give some insight into how software developers operate, and why the proposal caused so much fuss.

The first point to be made is that the proposed changes were not particularly complex or difficult to implement. Neither were software developers reluctant to implement them. The issue was entirely with the very short notice which was given.

Software developers like to operate to planned timetables in terms of releasing product updates. In Midland's case we aim to have two releases each year. One contains the updates for tax year end and the new tax year, while the second has general product enhancements. Other software developers may arrange the frequency and content of updates differently, but all will have a schedule which their business is geared around.

Short-notice changes therefore throw a spanner in the works as they require a deviation from the plans. This is inconvenient, but can be accommodated provided we're given enough time to do the necessary work. Consider the tax code changes in September 2008: these also involved producing an out-of-cycle update but software developers accomplished this without too much difficulty, since several months' notice was given.

Making this kind of change to software basically involves three tasks: decide what needs to be done, do it, then test it. In the case of the Scottish proposals, the testing phase would have accounted for most of the work. This is because the change affected the payroll calculation routine and after amending this we need to not only ensure that the changes we've made work properly but also that there have been no undesirable side effects.

This means testing gross pay calculations, statutory and occupational absence payments, tax, NI, student loans and a whole host of other things which contribute to the calculation of a payslip. All this testing takes time but is vital when such a critical software component is affected.

Finally, after the changes have been made and tested we need to make the software update available to our clients. In our case this would have involved production of a



'patch' to be downloaded from the internet but other software developers may have needed to produce and distribute CDs.

All this is time-consuming enough if only one software product is involved. In practice, however, most software developers will be supporting more than one product or version at any point in time and the changes need to be made to each of these. In Midland's case, we would have needed to develop and test the Scottish changes six times in all.

Hopefully you can see that asking software developers to do all this with just thirteen working days' notice was a bit of a tall order!

The other perennial problem with 'out of cycle' changes is getting everyone to install them in a timely manner.

Software developers publicise the availability of important software updates to our clients in a number of ways, such as via customer internet portals and newsletters and by using email alerts. However many of us have hundreds of clients and it's impractical to personally ring every one to make sure they're aware of the need to update their system.

Sadly, despite our best efforts we inevitably find that a small proportion of employers fail to install the updates in time. It's surprising just how unaware of statutory changes some payroll departments are. A number even managed to fail to appreciate the need to implement the September 2008 tax code changes, even though these had been front-page news for weeks beforehand.

This is, of course, an area where payroll departments employing IPP members definitely have an advantage. IPP members really have no excuse for not being up to date!

Reproduced by permission of- The Institute of Payroll Professionals