



where people come first

EVALUATING INTEGRATED HR & PAYROLL SYSTEMS

*Some notes that might help
from
Software for People*

EVALUATING INTEGRATED HR & PAYROLL SYSTEMS

Release 1.0 July 2005

Copyright © Software for People Limited 2005.

WHAT IS AN INTEGRATED HR AND PAYROLL SYSTEM

It appears generally to be agreed that an integrated HR and payroll system is a good idea but there is less agreement on what an integrated HR and payroll system actually is!

Some of the definitions of "integration" are:

"It must be one program rather than a number of programs linked in some way."

Well, it's results that count and if the user has a convenient way of working and reliability doesn't suffer we can't see what it matters how many programs are involved.

"It must use just one database."

It may be more convenient for the programmers of a system if it uses one database but, so long as the program presents the data to the user in the way the user wants to see it, what does it matter where it gets it from? The IT department might prefer one database (if they have to do the back-ups) but the user shouldn't notice any difference.

"Data must be entered only once."

This is getting nearer to the nub of the matter. Certainly, it should never be necessary to enter the same data, separately, into the two parts of the system but is that enough?

The definition we use is more along the lines of:

"All parts of the system must really be able to use the data regardless of where it is entered."

and that's what we want to talk about here.

The problem with integrating HR and payroll is that they work on completely different views of the world.

HR systems work on a continuous view of life. An employee exists over a substantial period of time and always has an annual salary although it might change from time to time. You can look back to see what it was some time ago and you can probably look forward to see what it's going to be some time in the future. You can see what it was when he or she joined the company and, in some cases, what it will be when he or she leaves the company.

Payroll systems work on a snapshot basis. They want to know what the employee's salary is at payroll processing time and they don't care what it was or will be at any other time. Payroll systems don't generally acknowledge that the world actually exists at any other time.

Somewhere along the line the information about a person's employment - salary, bonuses, benefits, time sheets, expenses etc. accumulated in the HR system have to be converted to "what shall we pay this person today". Someone has to perform that conversion and if you aren't careful, it might be you.

It may appear that you aren't putting the same data in twice but, if the data you are putting into the payroll system could have been calculated from what already exists in the HR system, you really are putting it in twice.

Let's look at some examples.

Most payroll systems allow you to enter an annual salary and will pay one twelfth of that salary a month.

How many allow that salary to have an effective date and how many allow you to enter a new salary, with its effective date, before it becomes effective?

If the payroll system doesn't allow that, then you have to make sure, on the day that the payroll is run, that every employee's salary is currently set to the correct value for that day. You might know that some salaries are going to change a couple of days later but you can't put that information in because the payroll would use those salaries instead. You have to wait until after the payroll has been run.

You really do have to do that, you know, in many payroll systems!

What happens if an employee started part-way through the pay period or left part way through the pay period or (heaven forbid) had a pay increase part way through the period? Simple. You have to calculate how much to pay that employee and override the standard payment.

Does that count as "putting the same data in twice" or not? It isn't *really* quite the same data is it? (Honest gov., would I lie to you?)

Now, let's be realistic. In an integrated HR and payroll system you would have already put in every employee's salary with a date from which it was effective and you would want the payroll system to work out which of those salary records were current during the pay period and, for any salary record that did not cover the entire period, to calculate a pro-rated payment for the part of the period that it did cover. For a mid-period salary change you would want it to calculate two pro-rated payments and add them together.

If it can't do a simple thing like that it isn't an integrated system regardless of how it's built.

Pro-rated how?

- By days worked versus days in the period?
- By week-days worked versus week-days in the period?
- By working days worked versus working days in the period?

Payroll systems can't do that last bit. They wouldn't have the employee's working patterns available to do the calculations - why should they?

The HR system should have the employee's working patterns available so an integrated system should be able to pro-rate that way!

Suppose an employee has two or more jobs. An HR system should be able to cope with that. He or she probably has two or three salaries at the same time. Any or all of them might start or end or change value during the pay period so the system must be able to find them all, pro-rate any of them that don't cover the entire pay period and add them all together for payment.

Payroll systems, generally, cannot do this sort of thing. They don't hold enough data to make it possible because they have no real interest in what is happening over time - just what the situation is on pay day.

HR systems generally do hold this sort of data.

An HR and payroll system should hold this data and, if it is to be called an integrated system, it *must* be able to use it in this way.

So, we conclude that an integrated HR and payroll system is nothing to do with the way it is built or where it stores its data but rather with how and where the data gets entered and whether it can be re-used in other parts of the system without further user intervention.

Case Studies

Let's look at a few other examples that we have encountered during our activity in the HR and payroll arena.

1. Helicopter Pilots

Envisage an organisation providing helicopter services to North Sea oil rigs. The organisation employs pilots who are paid a salary but also a bonus for each type of helicopter they are licensed to fly. To maintain their licences they have to take tests from time to time.

You would expect the HR system to maintain records of all the helicopters each pilot can fly with the amount of bonus and details of when the licence expires and to produce reminders of licences that are due to expire in the near future.

Any HR system worth its salt should be able to do that.

The payroll system has to pay the bonuses so how does it know how much to pay?

It *must* be possible for the payroll system to retrieve all the licence information for each pilot, detect those that are current at any time during the pay period, retrieve the bonus amount for each one, pro-rate those that start and/or end during the period, total them and pay the total bonus payment.

It *cannot* be an integrated system if the user has to perform any part of these calculations. All the information needed has already been entered into the HR system and if the payroll system cannot use it how can anyone pretend that the two are integrated?

Of course, it doesn't have to be pilot's licences. It could be a list of training courses a trainer is qualified to give or a list of foreign languages an employee can speak.

2. Timesheet and Expenses Authorisation

Suppose employees are submitting weekly timesheets and/or expenses claims. They might be entered directly by the employees using an employee self-service system, or by line managers for groups of employees or by someone in a central office from paper forms.

No matter. They get into the system somehow and they have to be paid.

The payroll system must be able to extract the information from those timesheet and expenses records and work out what to pay each employee but it isn't as simple as that.

Surely someone has to authorise payment of each of these timesheet and expenses records - particularly if they have been entered directly by the employees.

So the payroll system has to be able to distinguish between those that have been authorised and those that have not.

The payroll must then record which ones it has paid so that it can avoid paying them again next period.

At each payroll run, it must look back, possibly several months, to find records that were not paid before but have now been authorised so that it can now pay them.

So, again, we have the situation that payments to be made by the payroll system this period need not have been entered during this period but may have been entered at completely random dates and, indeed, by completely random people.

We also have the requirement that the payroll system must not only be able to retrieve pay-related records from anywhere on the database but must also be capable of updating them.

3. Monthly Timesheets

People like to work to real months but payrolls are usually run at some time before the end of the month. This means that a monthly timesheet cannot have been completed for the month in which the payroll is run by the time the payroll is run.

At least, if it has, the entries for the period between the date on which the payroll is run and the end of the month must be somewhat speculative.

What should the payroll system do in such a case? Well, it's obvious isn't it. It should pay that part of the timesheet from the start of the month to a nominated cut-off date that is a day or two before the payroll processing date. Next period, of course, it should pay the remainder of this period's timesheet and the first part of the next period's.

Isn't that what you would want it to do? It allows employees or managers to enter the timesheet data in the way they look at it - a monthly record - while the payroll system actually pays it at the time when it is sure that it has the correct data.

4. Hourly-Paid Contracts

Consider the situation, common in educational establishments, where a person is given a contract to perform a number of hours work on a given day of the week for a number of weeks. Possibly, the person has several other contracts for other days of the week or for other groups of weeks.

The creation of these contracts and the monitoring of them would be the responsibility of the HR department or, most likely, of various departmental heads but the payroll system has to pay them.

Again, the data is there in the system so the payroll system has got to be able to use it!

Should it pay every week of every contract that should have been completed before a certain cut-off date?

Or should it only pay those that have been authorised for payment?

If it pays all of them it must certainly cope with the possibility that someone realises that certain weeks that were paid were not actually worked and the money paid for them should be recovered.

In either case, at each payroll run, it must look back many months to see if there are any weeks that were not previously authorised for payment but now are or that were paid but have now been marked as to be recovered. It must then mark those it pays as "paid" and those it recovers as "recovered" so that it does not pay or recover them again next period.

Configurability

No two organisations will have the same HR requirements so no two will have exactly the same database structure.

Any decent HR system must have the ability to be configured to accept and present data in the way best suited to the user's way of working and the corollary to this is that data will be in unpredictable places and in unpredictable formats.

If the payroll system is to be able to extract the data from these unpredictable locations the payroll system must also be configurable.

There really is no point in giving an organisation an exact replica of the timesheet it has been using for years only to find that the payroll system can extract data from four other types of timesheet but not from this one.

So you have another criterion to check when you consider whether an HR system and a payroll system are truly integrated. Can the payroll system be configured to match changes to the HR system or is the HR system limited by the capabilities of the payroll system?

Getting Data Back

So far we have talked about getting data from the HR system into the payroll system. We should also look at how to get data back from the payroll system to the HR system.

In fact, we have mentioned, in some of the case studies above, that the payroll system must update various records from which it has to derived data so as to avoid deriving the same data again. This is, itself, useful data to the HR department. It enables answers to questions such as "How much has this contract cost us?", "How much is it going to cost us in future?", "How much would we save by cancelling it?" to be answered.

There are many other similar items of useful information, calculated by the payroll system, that ought to be available to the HR department. For example, if an employee has been granted relocation expenses of, say, £85 per month up to a maximum of £500 it would be nice to know how much has been paid to date. The payroll system will have calculated this (otherwise it couldn't be sure how much to pay next month) so the HR system should be able to see it.

Really, though, we are concerned here with the great mass of data that will have been derived by the payroll system as part of its processing. It will have calculated every single item that makes up the employee's gross pay and every single amount that should be deducted to create the employee's net pay and it isn't clear to us why that information should not be available to anyone authorised to see it. In particular, in an employee self service environment, it should be available to the employee.

HR systems usually retain historical data indefinitely so that questions and disputes can be resolved and because analysis of historical trends can be more revealing than looking at the current status.

Payroll systems also retain data historically but frequently do not make it readily available. The "year end" process in a payroll system often means closing this year's database and starting a new one for the next year. Analysis of data over two years (or over two different payrolls) is often difficult and sometimes impossible.

If the HR department has been used to looking at an employee's contract, job, salary, pension etc. details all the way back to the day the employee joined the company it isn't likely to be very happy if it can't do the same for the employee's gross and net pay details.

There's another criterion for an integrated system - consistency between the way the two halves operate from the user's point of view.

Conclusion

There isn't one. There can't be.

Our purpose has not been to tell you whether one HR/Payroll system or another is truly integrated. It hasn't even been to give you a checklist by which you can decide.

Our only purpose has been to give you some hints as to what questions you should ask (or, preferably, what you should ask to see demonstrated) when you are evaluating an allegedly integrated system.

Your requirements are known only to you and they may not be very similar to those described above but they almost certainly have something in common. You have to decide exactly what your requirements are and find out from your proposed suppliers exactly how they intend to meet them.