

3 Hands-on Techniques for Managing Operational Change in HRIS Projects

When IT or business managers talk about HRIS implementations, more often than not they soon get to the point that, whilst the implementation was “successful”, it failed to meet all the expectations.

One major reason for disappointing outcomes are users as well as IT staff not being 100% on board. Some may be confused, as they don't understand how their day-to-day processes are supposed to work. Some are disengaged, because they feel something has been forced upon them against their better insights. Some may even actively sabotage the new system, because they feel threatened. Others spend a good deal of their time finding ways around the new system and procedures, because they are convinced the changes just make everything more complicated and inefficient. And finally, some users may actually hit brick walls, when trying to do their jobs, because the system is wrongly designed indeed.

1. Change Management is the Silver Bullet?

As soon as you mention those symptoms, someone will say: “This is due to a lack of proper change management”. Excellent answer. But does it help?

When thinking of change management, most people think of changing culture, values and mindsets, of creating exhaustive stakeholder maps, of large all-hands meetings and workshop series. All these are valuable elements of change management indeed. However, a culture change can't be driven by a software implementation alone. These points all need to be addressed in a larger context. Whilst your SAP HCM or Successfactors project should consider the crucial techniques from the change management textbook, many initiatives aim to high and fail to achieve anything tangible. I'm not saying culture doesn't matter here. You've got to observe cultural fit to avoid disaster – most notably in ESS / MES projects, but changing it cannot be a deliverable of your project.

In this article, I will focus on 3 humble and tangible techniques, which can be used by any HRIS project or programme manager, don't require special change management training, and are addressing specific challenges of a typical ERP implementation.

2. What's changing in the new world?

Most HRIS projects, not just new implementations, will aim at “improved integration” as one of the major benefits. Be it that isolated systems for Finance, HR, CRM, etc. are now integrated into one; local systems of various countries or group companies are consolidated; or that the steps of one process hitherto connected with various interfaces or even partly managed on paper are now

smoothly flowing through the screens of the one chosen system or a highly integrated hybrid solution. In each of these cases a beautifully efficient process allows you to leave all the interface problems and inconsistencies of the past behind – or so it says on the tin.

ERP also promises quick and comprehensive reports and analytics based on the new homogenous database filled by standardised processes.

But this makes for considerable changes affecting almost every user: there is much less flexibility for the individual than there was with isolated systems. A fully integrated ERP always asks for standardised and rigorous processes of data capture. Where, in the old days, fields could be left empty until the month was to be closed, online reporting asks for instant data capture now. Where inconsistencies between local and central requirements could be dealt with manually in the Excel file sent to HQ, standardised processes now need to be followed rigorously. More fields are compulsory and many free text fields have been replaced by pre-configured drop-down lists.

As a rule, more time is saved at the tail end of each process, where analytics and other output benefit from high data quality than at the front end, where this data quality must be achieved. Some users may therefore perceive even a well designed system as cumbersome and less efficient. Self service applications take this to the extreme, where employees formerly not bothered by the system at all, are now asked to capture data, approve requests, etc.

It is also more difficult for one department or subsidiary to have the system changed, as it would now potentially affect the whole organisation in a web of complex interdependencies. If the chosen system is cloud based (HR SaaS), changes tend to be even more painful at first sight, because the coding is not meant to be changed to accommodate legacy processes.

So, users are facing a myriad of often small changes, in many cases involving less flexibility. Here are a few proven techniques to help you

- a) getting the system and processes right
- b) getting users on board, trained, and engaged

3. Quality in -> Quality out

When you expect users to guarantee high data quality, make sure they understand where the data they manipulate goes to and, whenever possible, allow them to benefit from it. Otherwise they often assume the process is unnecessarily complex and take shortcuts, not aware of the problems this is causing elsewhere. There are three actions to take:

- Provide overviews of all end-to-end processes, so that each user can see, where the information they are using is coming from, who is working with their output further down the line and how their work eventually affects customers, employees or other stakeholders

- Provide sample outputs from reports, forms, etc., where users can identify the data they capture in the final result.
- In SAP on-premise, you can sometimes overcome resistance by throwing money at the system, i.e. pay a developer to change it to match the legacy process, or some very bespoke, genuine requirements. In SaaS, you have less freedom. This may be a good thing in the long run and allows for faster deployment, but plan for some of the saved deployment time to be used for change management. Talk to people!
- If users can benefit from the data they capture themselves, provide reports allowing them to use it to the best effect. This technique is most successful, where line managers are asked to perform certain tasks on the system, e.g. capturing headcount planning data or approving employee's expenses. Systems are often designed to just cover the old process more efficiently without really using the potential of the new system. So, if line managers had to approve travel expenses on paper before the change, then the new system asks them to log in and do it online, but doesn't give them back any useful information like analytics of travel cost in their department. Therefore, the system is often seen as a black hole fed with data, but never giving anything back. It is perceived as serving accounting or HR departments at the expense of line managers. Best practise is to include analytics for line managers early on. Don't put it off until the next release, because by then the resistance may have killed the process already. Rather include less process scope in release one, but cover each process comprehensively.

4. Consider local requirements

ERP systems are often implemented using cumbersome methodologies like a V-model or SAP's "asap". They usually involve a design phase of many months or even a year, where requirements are defined with subject matter experts and future users, who don't understand the system yet. Therefore, the necessary drive to harmonise processes across the organisation, often overshoots its mark, when inexperienced users need to trust overconfident technical experts or consultants keen on keeping the initial estimate low while aiming at a rich harvest of change requests later.

To avoid surprises regarding local requirements and losing trust of stakeholders in the subsidiaries late in the process, I highly recommend to give users from each subsidiary the opportunity to give feedback based on playing with a prototype as soon as possible. They will then see, whether certain fields are missing or are irrelevant for their country. Do not wait for the final user acceptance test, although this should also involve representatives from each subsidiary as well.

This technique is more or less built in by design, if you use agile project management methodologies.

5. Keep a detailed change record

The most valuable tool in my opinion is also the most straightforward one. It just requires some discipline throughout the project. This is about keeping a detailed record of process changes caused by the project.

To be clear: this is not about the usual change log, which is keeping track of change requests against the initial specification and used for managing the budget amongst other things. It is also not the high level as-is vs. to-be comparison you find on a business case level. It is really about comparing the new process with the legacy process in detail, to see what users need to do differently in future.

It should be obvious that in order to manage change, it would be helpful to know exactly how this change looks like. However, it is rarely tracked on a level comprehensive and detailed enough, to give a satisfactory answer, if a user asks “What exactly does this mean for me?”.

And it's as simple as that: create a table somewhere to be accessed by your project team and make sure that, if anything is going to change in the way users will do their job in the future, it is tracked in this table. It could be a spreadsheet on a shared drive, but I recommend to use an online tool, which can be accessed from anywhere without the trouble of downloading a file first. The crucial point is that team members immediately record changes as and when they are recognised: in conversations with users, when analysing legacy data,...

The following figure shows a sample from a change record for an SAP HCM implementation kept on a project wiki (based on the Confluence software from Atlassian).

The tool captures:

- The assessment, whether the change is positive (e.g. increase efficiency), neutral or negative
- The process affected
- What exactly is changing
- Primarily affected stakeholders
- Who recorded the change?
- Further comment



Tracking process changes and impact

Added by Sven Ringling, last edited by Sven Ringling on 18.05.2010 (view change)

Edit Add Tools

How to use the template

Only process changes here. No other impact like IT cost etc.
Categories (column "Cat."): + (improvement), o (neutral), - (disadvantage)
The same change can fall into different categories for different groups of stakeholders. In this case, use multiple lines per change.
Column "by": who made this entry into the table?

Example: implementation of a new HRIS

Cat.	Process / Topic	Change / impact	Stakeholder affected	by	remark
+	Calculating working time	old systems error in calculation on night shifts is rectified -> no manual correction required any more	HR shared service team	SR	volume to be specified
+/-	Analytics	Managers run their own headcount reporting online -> immediately available, but need to log on themselves	Line managers	SR	some will like it, some won't. Average time to receive requested report in past was 2 days.
+	Analytics	Managers run their own headcount reporting online -> HR-analyst doesn't have to create and send reports, but can't manipulate any more	HR analyst	SR	talked to the guy. Sees it as entirely positive.
	data entry	Managers run their own headcount reporting online -> high data quality required at any time -> data needs to be captured as soon as it's known, can't wait till next payroll run	HR shared service	JE	lower efficiency as they can't "batch process" large chunks of data any more. Fragmentation of work.

In a real world project, hundreds of these changes will be recorded, but that's fine. After all, this information is invaluable to

- Illustrate the benefit of the new system to users, who often tend to forget very quickly about the weak points of the legacy system and like to complain about the few disadvantages in the new world
- Spot changes with negative impact and possibly find a way to avoid them
- Recognise impact on resource requirements per team or role, rather than assuming an overall equal percentage of efficiency gain
- Identify training needs / compile a "What's new" guide
- Verify that all necessary preparations are in place for the changes to come into effect

It is amazing, what a comprehensive change record can accomplish. This is not a comprehensive change management toolkit. Far from it. However, if you get these three things right your project is already likely to manage change better than many others.

This article was written by Sven Ringling, Director of iProCon Ltd. in London. You can contact him directly via email: s.ringling@iprocon.com or follow on Twitter: @svenringling