Processes automation with SharePoint: Windows SharePoint Foundation 2010 Workflow or Josh, the Business Process Management system?

by Gabriele Del Giovine (CTO, it Consult) and Giovanni Marrè (CEO, it Consult)

In SharePoint 2010 Microsoft has made a series of significant improvements to the Workflows available with SharePoint Designer, finally making them really usable as opposed to the previous version.

The main innovations are those linked to the "Site Workflows", the "Reusable Workflows, several improvements to the available actions and to the integration with Visio. In this article, we will be concentrating on the reusable workflows, the improvements to the actions and to the integration with Visio; we will then examine the areas and applications where the use of an advanced tool for process modelling like **josh** by **it Consult** is indicated.

Reusable Workflows

For the user who typically is not a developer, whose only development tool is SharePoint Designer and who, in any case, wants to deal with the management of workflows in SharePoint, the possibility of creating "Reusable Workflows" offers a series of possibilities which, in the previous version, were subject to, at best, embarrassing restrictions: a workflow designed with SharePoint Design 2007 remains indissolubly linked to the list it is associated with when created, i.e. in the WSS 3.0/MOSS 2007 architecture it is not possible to develop a workflow with SharePoint Designer 2007 and then reuse it in other lists even if they have the same structure. The SharePoint 2010 Reusable Workflows solve this problem, making it possible to develop Workflows and to reuse them in any list of any site and, above all, to packetize them in a standard SharePoint solution (WSP).

Generally speaking, the Reusable Workflow is independent of a list and contains within it the definition of all of the columns necessary in order to execute it. The columns, thus defined, will be added to the list with which the Workflow will be associated. In addition, a Workflow may be associated with a specific Content Type and be activated only for the elements having that particular Content Type. If it is decided to associate a Workflow to a content type, bear in mind that the association is only with that specific Content Type and not, as may be expected, with all of the Content Types deriving from it.

Improvement in the Activities

Several improvements have been made to the "Activities", the most significant being the process execution identity. While in the previous version, all of the flow steps were executed with the identity of the user who created the workflow, now it is possible to decide whether or not to execute all or part of the flow with the identity of the user who created/associated the workflow with the list. Moreover, among the available actions, the possibility of managing list item permissions has been added. Essentially, these are the only real improvements.

Integration with Visio

Microsoft marketing has emphasised the use of Visio as a design tool for SharePoint 2010 workflows. This may lead to excessively high expectations on the part of the user with regard to this integration. It must be immediately made clear that the possibility of using Visio as a workflow development tool does not entirely correspond to reality: Visio is used exclusively for the drawing part of the diagram which represents the workflow and NOT as a complete workflow development tool. In substance, Visio generates or manipulates XOML files which constitute the SharePoint Designer processes, without – however – being able to modify the parts where the activities to be carried out are specified. The "programming" work of the workflow must, in any case, be developed within SharePoint Designer 2010 using, among other things, a formalism for representing the work flow which is completely different from that used within Visio. Moreover, it must be considered that the workflows available with SharePoint Designer 2010 are exclusively sequential in nature without the possibility of any cyclical activity.

The weak points of the SharePoint 2010 Workflows

For the first time, the workflows in SharePoint 2010 are truly usable by professionals who are not .NET developers with a solid knowledge of the Windows Workflow Foundation, however many of limitations and weaknesses which characterised the workflows in the previous version still remain.

Summarising, they include:

1) Use of the sequential model only, further limited by the impossibility of having iterative cycles, i.e. the impossibility of repeating a step
2) Lack of a centralised infrastructure for controlling the active processes and the tasks assigned to each single user
3) Limited activity set and lack of activities for security settings
4) Visibility of the objects managed by the predefined activities limited to the site in which the flow is executed
5) Highly limited management of the association/initialisation/flow use user interfaces if the InfoPath Form Services infrastructure is not available (part of the SharePoint 2010 Enterprise CAL)
6) Absolute lack of any mechanism for assigning to the users the activities constituting the workflow, if not expressly declared by the operator who manages the association/ initialisation phases of the same flow

The above limits may be partially resolved by using a combination of flow programming and user activity development techniques which are available on the market or which can be

CHARACTERISTIC	SharePoint Workflow	josh (ON SP)
Processes which are NOT only sequential(fork, join, cycles, wait)	**NO**	**YES**
Centralised control infrastructure	**NO**	**YES**
Set of primitives for security settings	**NO**	**YES**
Advanced user i/f for association, initialisation, use	**only with Enterp. CAL**	**YES**
Role-based assignment of user tasks	**NO**	**YES**
Parametric assignment of user tasks	**NO**	**YES**

internally developed with Visual Studio 2010. In some cases (development of activities for security management or cross-site scripting operations) the solution is completely integrated and satisfying, in others the only possible solution leads to a distortion of the tool itself, for example when there are no waiting cycles which lead to a rough simulation of state-flows and to the consequent creation of a number of instances of the same workflow equal to the number of changes in state that have occurred. In addition, the use of the custom Activities reintroduces the same problem that exists in the development of WWF workflows with Visual Studio; i.e. the need to deploy (install) the executable files (DLL) in the Global Assembly Cache of each SharePoint machine, which implicates the arrest of all of the SharePoint web applications within a farm each time a single modification is made to a single Activity.

How to overcome the limits of SharePoint with josh

SharePoint is a formidable tool which covers a wide range of needs that go from the collaborative portal to advanced document management. Over the years it has matured significantly and today its suitability for even the most complex enterprise environments is evident, despite a much lower cost than numerous other ECM (Enterprise Content Management) platforms. What's more, starting from the 2003 version and even more so with the two successive versions, SharePoint has radically changed the ECM market, taking it from a small niche market to become an application typology frequently destined to all of the users of all of the organisations. It is a transformation that has definitively changed the market, also in favour of the other players.

Yet, as we have seen, the workflow components it comes with have several limits and are far from being comparable to those of a modern BPM (Business Process Management) system. However, it is evident that BPM features, integrated with ECM features – if the idea of such a net distinction still makes sense – are necessary for obtaining the performance levels frequently needed in organisational processes and they must be obtained easily and rapidly. The formal description and the automation of the processes makes it possible to obtain consistent and measurable results in terms of costs and effectiveness; and the more friendly a tool is for the user (and for the business side stakeholder) as compared to IT personnel, the better these results are reached. **josh**, in particular, is well integrated with SharePoint and meets each of the above mentioned limits:

1) Availability of a variety of modelling elements, including forks (parallel activities) and cycles
2) Availability of a centralised control infrastructure that allows supervisors who have permissions, to see the active or completed processes, the state of advancement of each, the executor of each task (activity) and the relative times. But, the single users can also graphically see the state of advancement of a process, acquire knowledge and have available coordination information with no additional effort.

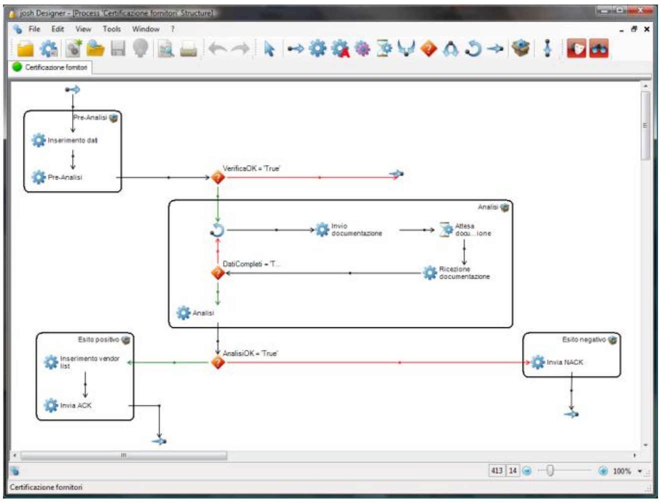
3) Well articulated set of activities, with operations destined to safety settings also usable for changing permissions during the execution of a process
4) Visibility, on the part of a process, of all of the SharePoint objects and not just from one site
5) Management of the user interfaces for association/ initialisation/use of most powerful and customisable processes
6) Availability of numerous mechanisms for assigning activities that form a process to users, in particular based on organisational roles, but also with parametric criteria (for example it is possible to assign a task to someone who has already carried out another in the same process instance, or to the person who, on the flow chart, is in charge of the executor of another given task).

Other characteristics of josh

If, on the one hand, it is clear why, both with SharePoint 2007 and with the improved SharePoint 2010, many public and private organisations have integrated the SharePoint infrastructure with **josh,** frequently there are other reasons for adopting this software platform. Leaving aside the availability of vertical modules which specialise the **josh** platform for several specific needs, and numerous specific technical characteristics which range from the expressiveness of the graphic language to the availability of mechanisms for automatic document production in function of the **josh** process variables (ex. Word documents instanced with typed values or chosen by the user in the course of the execution of a process) as well as a series of “knowledge oriented” functions, including the automatic suggestion of useful documents and the search for experts on the specific tasks, these reasons can be traced to two large categories, that differ from each other most in terms of approach:

- Business processes characterised by a level of dynamism that is so high that it makes the development of traditional applications impracticable, where the frequency with which the processes change make intervention of IT personnel absolutely untimely with respect to the need (as well as very costly)
- Presence of unstructured processes, mostly knowledge intensive, in which many tasks more closely resemble a check list (which, however, provides visibility to management) than a real model with all of the details of the activities

These contexts are generally distinct from each other, but both are marked



A process within josh Designer tool

by features that are more organisation than IT oriented and both seek to maintain a wide (if not total) independence from the IT function in making modifications and adjustments to the behaviour and functionality of their processes. The solution they find in SharePoint alone is not sufficient, especially in contrast to the “self-service” features available for content related settings; however, with **josh** this requirement is met and there is even another method available for facilitating and/or filtering user access to document libraries or SharePoint data lists.

Conclusions

The workflows available with SharePoint Designer 2010 in Sharepoint 2010 can finally be used in spheres which are not “amateurish” or exclusively intended for testing. However, they lack many functions for defining a tool suitable to a use that goes beyond the departmental one. Several of these gaps can be filled through the purchase or development of custom activities. However, it must be considered that these activities reintroduce the problems typical of workflows developed with Visual Studio, i.e. the need to interrupt the functions (even if for a very brief time) of the SharePoint web applications during deployment in addition to the need to modify the very delicate configuration file of the web application (web.config).

Therefore, if the use is not limited to a few people, or if the sophistication of the processes to be modelled and automated are not banal or if the processes are subject to frequent modifications (for example 3 or more times a year), it is advantageous to adopt the **josh** platform by **it Consult**, which makes it easy to extend the automation to a growing number of processes, rapidly writing off the relative license costs.

The integration of the SharePoint workflows with Visio must be considered for what it is: a diagramming tool capable of exporting/importing SharePoint Designer 2010 workflows, however without the capacity of acting on the activities constituting the workflows and introducing, among other things, the use of different flow representation formalisms in function of the context with the risk of generating additional misunderstandings between process analyst and “developer”. Again in this case, the use of a professional BPM tool like **josh** constitutes a response which is closer to business needs and with a greater probability of success.