At first, trying to compare these two products seems a bit like comparing apples to oranges. In a sense we are, because Microsoft® Office SharePoint® Server (MOSS) is designed to be a portal, a collaboration tool, an application framework, and other things as well. In contrast, JobTraQ® is designed to be one thing, an enterprise grade task management and workflow tool. The question of contrast arises, when one considers the task management and workflow capabilities of MOSS against those of JobTraQ®. Here we will compare the products from a task management perspective in terms of ease of use, implementation time and complexity for developers as well as licensing costs.

In a recent survey, most respondents reported that SharePoint® Server is not easily utilized out of the box or its purpose understood by end users. [1] To be useful, it requires intensive time consuming customization. The main task management problem with respect to ease of use with MOSS is its limitation in tracking all of your projects and tasks simultaneously; it only allows viewing one project at a time. It does not allow visibility of all the tasks and projects in the system. This makes the needs of higher level program management a challenge, as well as making it difficult for individual contributors or team leaders to get a macro look at the requests and requirements they are expected to fulfill.

Criticism on the web abounds on the implementation difficulties associated with SharePoint® development even as a portal and collaboration tool. To implement custom workflows with MOSS, SharePoint® Designer is required and if coding is required, also Visual Studio with two SharePoint® extensions installed.[2] To create the forms associated with workflows, Microsoft® InfoPath, an XML forms editor is required. Developing InfoPath forms is so difficult that even Microsoft® Certified Professionals criticize InfoPath. [3] The ramifications of requiring multiple products for implementing task management with Microsoft® products is that the cost, complexity and implementation time increase. This follows because all of these products must be installed, configured, learned, and maintained. Developers of SharePoint® applications must possess knowledge of a dozen other technologies including IIS, SQL Server, Indexing, XML, ISA server and more.[4]

MOSS workflows are created with SharePoint® Designer. There are many documented issues with developing workflows in SharePoint® Designer as well as Visual Studio. SharePoint® Designer is targeted at business users, but giving business users elevated privileges to make site changes is not optimal. Using SharePoint® Designer can cause your site to become customized, or in SharePoint® terminology “unghosted”. This means that pages are not loaded from templates, which produces suboptimal performance and appearance.[5] Workflows created with Designer are not reusable because they are bound to a single SharePoint® list. There is no ability to debug workflows from SharePoint® Designer, making workflows created with SharePoint® Designer more difficult to get working properly

[1] http://stackoverflow.com/questions/493312/how-is-SharePoint-moss-2007-perceived-in-your-company
[2] http://www.k2distillery.com/2007/10/blackpearl-and-moss-workflow-options.html
[3] http://www.novolocus.com/2007/07/26/infopath-criticisms/#more-58
http://www.thorprojects.com/blog/archive/2007/07/07/dear-mr-infopath-an-open-letter-to-the-infopath- team.aspx
[4] http://www.realsoftwaredevelopment.com/why-SharePoint-portal-server-is-terrible/
[5] http://www.k2distillery.com/2007/10/blackpearl-and-moss-workflow-options.html

and much more difficult to diagnose. SharePoint® Designer offers limited conditional logic, requiring the use of Visual Studio, which is targeted at developers. This means a developer will then be required to build workflows with moderately complex conditional logic. This pushes workflow implementation to developers working with Visual Studio thus defeating the purposes of pushing administration to the line managers. Disadvantages of doing MOSS workflow in Visual Studio include deployment complexity, and requisite custom development of robust audit and metric data. In addition, software to integrate with other platforms must be built from scratch.[6] One evaluator does not believe these concerns with development complexity will be mitigated with Office SharePoint® Server 2010.[7]

To create workflows with JobTraQ®, there are a handful of pages in which technically savvy business users complete various system configuration tasks including creating task templates, fields, statuses and rules. These building blocks culminate to the preparation of custom user facing screens through simple drag and drop forms creation, all completed through a standard web browser. Workflow rules - created using simple pick lists, no developer required - can change the assignee, status or the value of any built-in or custom field using simple formulas or pick list options. Tasks and fields can be secured with powerful role based or individually applied privileges. This is in contrast to having to use InfoPath for custom web forms, SharePoint® Designer or Visual Studio for creating workflow rules and SharePoint® server to publish and view changes and several mechanisms to secure files and forms. Field level permissions and auditing (out of the box features in JobTraQ®) are also very complex to implement and require advanced configuration and programming to accomplish. The same workflows can be created with a handful of pages in JobTraQ®. Changes to tasks, fields, forms, and permissions are immediately in effect; there is no subsequent deployment or publishing required with JobTraQ®, and unlike SharePoint®, JobTraQ® offers a complete configuration audit trail, showing you who changed what objects such as field names, workflow rules, form layout, system configuration, permissions and more, and when they did it.

For more complex workflows that require integration with other information systems, the JobTraQ® Web Services compliant API allows data to be extracted from and inserted into the JobTraQ® task and contact database, tasks to be created and updated, contacts to be synchronized, as well as a complete and robust Web Service delivering any capability available from the JobTraQ® user interface.

To incorporate reporting into your solution the SharePoint® Reporting add-in is needed as well as service packs for SQL Reporting Services for SQL Server, increasing the installation and configuration complexity. There will be no stock reports on task management built into SharePoint®. All reports will have to be written by developers from scratch or schemas for ad-hoc reports will need to be pre-configured by developers and published. Conversely with either the standard JobTraQ® system or with the more advanced JobTraQ® Business Intelligence Module, the setup will install everything needed for reporting. With JobTraQ®, end users can create their own reports eliminating the need for developers to build them. JobTraQ® is also built on the SQL/IIS stack and will scale comparably to SQL Reporting.

[6] http://www.k2distillery.com/2007/10/blackpearl-and-moss-workflow-options.html
[7] http://arnoldit.com/wordpress/category/enterprise/

In effect we are comparing using JobTraQ® for task management to using multiple Microsoft® server and thick client developer studio products. These products are also more costly in terms of price, complexity and solution delivery time. Compared to using the Microsoft® suite, the initiative to deploy task management with custom workflows goes from a project requiring multiple developers with an array of skill sets, Microsoft® server products and advanced development studios to a business analyst trained in JobTraQ®. JobTraQ® administrator training requires one to two weeks. Formal 3 and 5 day classes are available, and a few additional days to become more familiar with the application is all that is required.

Of particular importance to the public sector and organizations concerned about being compliant with the Citizens with Disabilities Act, SharePoint® does not meet Section 508 compliance for end users with disabilities without significant customization. [8] This is a due to the number of ActiveX browser plug-in controls, the use of non-compliant markup, and overdependence on client-side scripting throughout SharePoint®. As of this writing the JobTraQ® user interface is 508 compliant with no customizations required.

[8] http://www.SharePoint2007security.com/guidance/accessibility_section_508_compliance