

Efficiency Through Document Automation

BY ROBERT DELWOOD

Today's tight budgets and reduced staff mean we need ways to work more efficiently. For office and knowledge workers that should include document automation. Yet organizations overlook a powerful tool they use every day: Microsoft Office. Office automation offers a powerful set of capabilities, both out of the box and with programming, that can transform the way you create and manage documents.

Microsoft Office automation uses built-in features either to place, reconfigure, or format document passages, or to automatically run sequences of commands. The simpler end of the automation continuum includes built-in features such as copy and paste, the "repeat last" command, and document linking (reusing text among different documents), all examples of tasks that can run directly from the Office ribbon. Advanced users may record macros (linking a series of commands) or manage macro libraries to shorten tasks. At the other end of the continuum, programmers create custom applications that manipulate data in specific ways. The intent is the same: to be more efficient by increasing throughput and reducing opportunities for error.

ORBIT®

The advantages are clear, but implementation can be an issue. If it's the job of individual contributors to improve their processes, why doesn't it happen more often? For one thing, automation tools, though often not complex, do require users to change the way they think about their work. There is also often a mismatch of expertise:

- Users understand their own procedures but may not know what can be automated.
- Automation specialists know how to implement Office automation but may not know users' tasks.

The solution is to get the two together. We assign an automation specialist, typically a programmer with Microsoft Office automation experience, to a team to learn users'

procedures. Both parties then brainstorm to come up with automation suggestions. We call the process ORBIT, an acronym of five steps:

Observe. Initial observation focuses on the most repetitious procedures to develop a list of candidate processes. In some cases, an automation audit has the team members explain their work step by step, down to the key stroke. The specialist then has the opportunity to ask questions and investigate options. Typically, an audit of no more than thirty minutes can yield several candidate processes for automation. At other times, the observer actually sits with the team and observes their work.

Reengineer. The automation specialist and users work on new procedures, adapting tasks to automation. This may be as simple as recording macros, but sometimes procedures may have to be changed to accommodate automation capabilities. This step also provides an opportunity to eliminate procedures that are no longer needed.

Build. The specialist creates the application or macros to automate the clients' needs. The solutions and user interface must fit their working environment. Builds may also include a prototype or progressive versions of the application to show its proposed look and feel.

Implement and Test. Specialists test the application and interact with the clients. Some features might be reworked

GETTING STARTED WITH AUTOMATION.

You choose your level of involvement with automation. A simple start is the “repeat last” command, or the **F4 key**. This repeats the last command you invoked. For instance, if you need to make lots of text bold, save that operation until last. After bolding the first text, highlight the next instance and **press F4**. You can go through an entire document quite quickly.

If you want to record a macro in Microsoft Word for Windows 2007 or later, start by clicking **View/Tools > Macros > Macros > Record Macro**. Give it a name, **press OK**, then go through all the steps needed to complete your task. Stop the recording by clicking **View/Tools > Macros > Macros > Stop Recording**. To play the macro, click **View/Tools > Macros > Macros**, select the macro’s name, and **press Run**.

or redesigned based on feedback. Once the clients’ requirements are met, the specialists release the application.

Transfer. We deliver the application to users. Some applications can be modified to accommodate more general requirements for other groups. We place these tools in a public download site and inform potential users and other contractors of their availability.

Sample Cases

Some NASA cases illustrate this approach. At the Johnson Space Center, our DQA (document quality assurance) team, charged with enforcing document quality and style consistency, had unrecognized outdated procedures. One procedure in particular caught the attention of an automation specialist. He noticed that a DQA team member was going through a lengthy printed document page by page. Repetitive actions like that should always be evaluated for automation. The team member was reviewing the manual placement of eighty pages of graphics. The graphics, up to 150 diagrams, were a parts manifest list for the space station in a read-only format. They were routinely and manually inserted into documents. Each diagram had to be printed from its source document, scanned, converted to a TIFF file (tagged image file format), then opened in Adobe Photoshop, where it was cropped and occasionally had text removed. Then it was inserted into the target document and its size possibly adjusted. A single document took twenty to fifty hours.

The process followed clear, repeatable guidelines, which made it ideal for automation. Scripting could print pages from within Word one at a time, create TIFF files by using Microsoft

Office’s Document Imaging driver, and place them in the target document, resizing them with mathematical precision. We questioned the requirement to remove text and found it to be an old constraint, not needed any more. The resulting solution was a Microsoft Visual Basic 6 standalone Windows application (an executable or .exe file), since it would involve multiple documents, and the code wasn’t conducive to being in a document template.

This automation reduced processing time to less than ten minutes with a zero defect rate. Conventional, top-down process improvement would probably have overlooked this task, since it wasn’t considered broken. The user who spent so many hours doing the work didn’t know it could be automated; it took an experienced automation specialist to identify the opportunity.

Knowing that automation can help with the smallest details, DQA team members began suggesting processes themselves. One was to create tables. New tables have up to fourteen requirements, including multiple font styles, column and row widths, and header formatting. A macro was recorded to create them. The time and quality savings were clear, but that wasn’t the real draw of automation here. Of greater concern were existing tables. Having been modified by different authors over time, they were prone to inconsistencies and each had to be manually checked prior to release. This was a good automation candidate since the procedure was manual and repetitive with clear, objective requirements.

We created a macro series to either check each criterion individually, or check all table criteria in the document at one time. The scope later expanded to address user-entry mistakes, replacing line returns with paragraph marks, removing double spaces between words, removing spaces used as line returns,

WHAT MAKES FOR A POOR AUTOMATION PROJECT?

Not all document tasks are good candidates for automation. The following often reduce automation benefits or prevent automation entirely.

- Unclear or changing requirements
- Projects that include too many manual decision points, or interventions
- Subject-matter experts who “just know” what the right action is but who can’t or won’t explicitly describe the logic behind it
- Projects that have too many rule exceptions
- Inconsistently formatted documents, often the result of many authors modifying the document and/or modification over a long period of time

Finally, some processes just don’t convert well to automation steps, either by needing features not available or not supported by Microsoft Office, or because automation isn’t practical in a particular environment.

holding certain words, and removing empty table rows. The greatest benefit was not time savings, although they were significant, but the consistency and assurance that the tiniest of details were fixed.

Our last example is of a complex application, initiated by management to address a specific, long-standing problem: acronyms. Many NASA documents require a list of acronyms and their definitions. In some cases, spelling out the first use of acronyms is required. Acronym management had been a time-consuming and often inaccurate manual process at the agency.

Given the ambiguities and complexities of acronym usage, this became a formal information-technology development project. The key was using Word’s automation functions. With a combination of its wildcard find and replace, thesaurus, custom dictionaries, and other features, it was possible to reach nearly 100 percent accuracy. A separate list maintains customized acronym terms. In addition, the application lists words that might be previously unidentified acronyms. Rounding out the application are systems to add or modify terms and definitions, selecting from multiple definitions for the same term, producing and saving terms lists, and generating a new acronym appendix ready to copy and paste into the target document.

The resulting tool had three benefits. The immediate one was that it reduced document processing time for that step from a high of twenty hours to about thirty minutes, along with a dramatically higher accuracy rate. Second, after studying the process in detail, we concluded that authors and editors who supplied documents to DQA could use the tool to catch mistakes earlier and correct those mistakes faster. Lastly, the tool was so effective that we modified it for use by other groups or other

contracts and made it available as a free download. It’s now used routinely for contract proposals, white papers, and presentations.

With two or three highly visible projects completed, the teams became more confident and started suggesting automation ideas on their own. To date, we have provided more than a dozen tools. Document turnaround time dropped from the required twenty days to just three days. In addition to saving time and money, these tools can reduce stress. Some of our teams found that reviewing document changes in Microsoft Word is awkward and doesn’t lend itself well to group meetings. We wrote a tool that cataloged all revisions in a document to a separate list along with user-requested information. One meeting organizer wrote, “This is the coolest tool ever and why everybody doesn’t use it is beyond me. I love this thing!” ●



ROBERT DELWOOD is a senior systems analyst for Barrios Technology.