

**We get up early so that you don't have to.**

## Review: ActiveBatch 5

**ActiveBatch 5.0, starting at \$2,995**

Advanced Systems Concepts

Parsippany, New Jersey

(973) 539-2660

<http://www.advsyscon.com/products/ActiveBatch/ActiveBatch.asp?mode=overview>

There are plenty of products out there to help you automate routine processes on your computer, starting with the lowly Windows batch (or command, depending on whether you're a server type of person) file. If you're looking for an enterprise-level product that can take that automation and extend it across an entire network of computers featuring heterogeneous operating systems, you need to look at a product like ActiveBatch. You pay a lot more, but you also get a lot more.

While centered on Windows, ActiveBatch also support Unix, Linux, and OpenVMS. It can manage and schedule individual jobs, but it also has the concept of a "job plan", a collection of jobs. Job plan can include complex dependencies and involve multiple machines. For example, you could have a job plan where Job A kicks off at a preset time, and if it succeeds Job B runs on a different machine (but only if that machine has 100 MB of free disk space), while if it fails Jobs C and D run concurrently on two other machines. That only scratches the surface of the management possibilities that plans allow you; learning how they work (and how the graphical designer lets you set them up) is the major part of the ActiveBatch learning curve.

Jobs can be triggered by, well, just about anything. They can be launched based on dates in any syntax from fixed time to "last fiscal day of the quarter". Jobs can be triggered by the success or failure of other jobs, by system startup, by the creation of files, or by any event that you can monitor through WMI. Jobs can run on multiple machines simultaneously, and you can manage an entire ActiveBatch installation across a network from a single administrative console. You pay for the console, but then you can deploy the clients on as many machines as you require.

ActiveBatch features a fine-grained and full security model that is completely integrated with Windows, right down to Kerberos support if you're using Windows 2003 in your network. It can hook into external data sources like SQL databases and WMI, letting you set up dependencies on just about anything you can dream of as constraints on execution or as parameters to jobs. You can also use variables to pass information between jobs, so that information from one job can be used to affect the way that the next job in a plan is executed. There are an excellent set of management tools that let you see what jobs are currently running as well as what jobs were running at any time in the past, as well as what (if anything) failed. There's also a nifty real-time graphical debugger. Good documentation will walk you through the process of setting everything up - the documentation uses little baby scripts for all its examples so it can concentrate on the ActiveBatch plumbing, on the presumption that you know how your own business processes work better than they do.

Like most enterprise-level products, ActiveBatch is not something to be purchased lightly. Serious customers can request an evaluation kit, and clearly this is the sort of thing you should evaluate before sinking money into. They've also got a developers program that offers lower-cost access for those who want to piggyback their own solutions on the ActiveBatch engine, which makes all of its functionality available through COM interfaces.

*[Mike Gunderloy](#) is the lead developer for Larkware and author of numerous books and articles on programming topics.*