

Q&A: Save Time and Money with Automated Data Warehouse Development

Technology can eliminate time-consuming, repetitive tasks like mapping data sources, scheduling jobs, and generating documentation — and can automate coding.

By Linda L. Briggs

“What’s pragmatic about data warehouse automation is that it works with the concrete reality of what’s already there,” according to **Mark Budzinski**. “You aren’t ripping and replacing. You aren’t blindly automating,” he explains in this interview. Instead, “you’re leveraging the strengths of your existing IT systems and using automation tools to bridge gaps and eliminate impediments.” Budzinski is the president of WhereScape USA, which offers products for agile, rapid development of data warehouses and related data-driven projects. Budzinski’s career includes management positions at Intel, Sequent Computer Systems, RadiSys, and Applied Microsystems. In this interview, he addresses data warehouse automation issues.

BI This Week: What are the key trends and issues you hear in conversations with customers?

Mark Budzinski: I talk with a lot of CIOs, and a common theme is that they have too many projects, too short a timeframe, and too few resources. In addition, the business is absolutely exasperated with IT. In many cases, they’ve given up on IT and developed their own solutions via a “shadow IT” organization. In response, the good IT organizations are now trying to beef up their level of service; IT needs to prove it’s a good-faith partner to the line of business.

Do you think established vendors do a good job of addressing these issues?

No, because they aren’t *interested* in addressing them. Look at legacy ETL tools. Legacy ETL requires you to put a staging area right in the middle of your BI and analytic production line.

If you were a manufacturer, would you really add a complex, costly, time-consuming, unnecessary step to your production line?

OK, you used to have to stage data so you could prepare it before loading it into the warehouse, but today’s systems are so powerful that the warehouse can easily do that work itself. You can automatically load data into the warehouse and do your data prep right there.

In this case, you’re moving data once instead of two or three times. Data warehouse automation addresses situations just like this. It’s a way for you to optimize your production line by eliminating superfluous actions.

What is data warehouse automation and why is it useful?

I think it would be appropriate to use TDWI’s own definition. In its data warehouse automation course, TDWI defines data warehouse automation as “using technology to gain efficiencies and improve effectiveness in data warehousing processes. Data warehouse automation is much more than simply automating the development process. It encompasses all of the core processes of data warehousing including design, development, testing, deployment, operations, impact analysis, and change management.”

First of all, automation isn’t a license to just blindly automate everything. That’s impossible. We think that if you’re pragmatic about how and why you use automation, you can deliver BI and analytic projects at the speed of your business. You can govern, you can have clean, consistent data, and you can also be agile, resilient, and flexible. What’s pragmatic about data warehouse automation is that it works with the concrete reality of what’s already there. You aren’t ripping and replacing. You aren’t blindly automating. You’re leveraging the strengths of your existing IT systems and you’re using automation tools to bridge gaps and eliminate impediments.

With data warehouse automation, you’re using technology to accelerate repetitive and time-consuming tasks, but your ultimate goal is to enable people to get stuff done better and faster. Ultimately, the best way to do that is by getting people involved. I’m talking about

collaborative design and development, and about business people and IT people working together to design, build, and test BI and analytic apps. That’s another key part of data warehouse automation.

What steps in the data warehouse development process can be automated? When we talk about “automation,” what we’re really talking about is eliminating time-consuming, repetitive tasks — those tasks that don’t require human intelligence or creativity, but which, for whatever reason, are still being done by human beings. That includes things like mapping data sources, scheduling jobs, and generating documentation. You can automate all of that.

You can automate coding, too. Companies have legacy ETL tools that can do 80 or 90 percent of what they need, but they still have these gaps, so they use human coders to plug those gaps in the data warehouse itself. A core goal of data warehouse automation is to completely eliminate hand coding. Human beings shouldn’t be doing this kind of thing. It can be done faster, better — and at considerable cost savings — by software.

What are the benefits of using software instead of humans? The primary benefits are rapid time-to-value and an ability to quickly respond to changing business conditions. You’re delivering new BI and analytic functionality in days, as opposed to weeks or months. You’re giving business people what they want when they want it, as opposed to what they asked for six months ago.

You’re getting the biggest benefits of self-service BI — user empowerment and quick time to delivery — in a model that’s governable, manageable, and scalable.

Those sound largely like business benefits. What are some of the technical benefits of adopting data warehouse automation?

How about vastly improved manageability, scalability, and resilience, as well as significant cost savings? If you’re coming from mostly

hand-coded ETL or from a legacy ETL tool, you're going to save a lot of money. By the way, everybody knows hand-coded stuff is hard to scale, but legacy ETL can be tricky, too. As conditions change, you'll find yourself doing more and more hand coding or writing more and more scripts just to keep things going. Compared with data warehouse automation, neither approach scales very well.

Another benefit is more productive IT workers. They don't have to write and debug scripts or code. The software does it automatically. IT doesn't have to resort to the "master spreadsheets" companies use to keep track of source mappings and other items. Workers aren't robotically documenting what they're doing. They have one solution to manage all of this.

What are some of the barriers and resistance to adopting data warehouse automation?

Fear, uncertainty, and doubt. A lot of that is whipped up by vendors with a vested interest in the status quo. There's a lack of awareness, too, about what automation is and about just what you're automating.

Another huge barrier is culture. If you're a company that's obsessed with controlling every aspect of project management — for example, by exhaustively gathering and defining requirements in advance — you're not going to be a good candidate for automation.

Assuming it's a good fit, how do companies get started with data warehouse automation?

Start by educating yourself. TDWI offers a full-day course on data warehouse automation. If you can, attend the class. If you

can't, do some research. There are plenty of resources online — starting with wherescape.com. Check out what we do and what our competitors do. Be critical, be skeptical: I love it when people challenge us. You owe it to yourself to do your due diligence. I think you'll find that automation can only make what you have even better. And if what you have isn't working, there's a good chance automation can help with that, too.

Critics say that automated data warehouse development can never replace custom hand coding. What is your response to that?

Actually, we have dozens of customers who have eliminated hand coding completely. They use WhereScape to generate SQL, procedural code, and even scripts. Is hand-coded SQL somehow "better" than automatically generated code? Given infinite time and an unlimited budget, a SQL-coding genius could probably match what a product like WhereScape RED can do, but the irrefutable fact is that no company — none — has an infinite amount of time and an unlimited budget. Besides, SQL-coding geniuses are a hot commodity: they're out there, but there's an extremely limited pool of them.

I can't speak for my competitors, but we look at WhereScape RED as a kind of SQL-coding genius: when RED's loading data into a Teradata target for example, it uses Teradata's parallel loading technology, and it generates code that's optimized to take advantage of Teradata-specific enhancements to SQL.

Hand coding works for one-offs, but as soon as you start talking about multiple instances of one-off stuff, you have a headache. It doesn't scale, it can't be managed, and it's ungovernable.

Specifically, what does WhereScape bring to this discussion?

WhereScape RED, our flagship automation tool, generates platform-optimized SQL code or — if necessary — procedural code, which we embed in user-defined functions. For scoping and modeling a data warehouse, we have a tool called WhereScape 3D. You use 3D to interactively explore and model data sources and warehouse targets. When you're finished, 3D automatically generates the mappings, project design documentation, and other artifacts.

This is a great example of "pragmatic automation." Human beings provide the intelligence and imagination, and 3D takes care of the boring, repetitive stuff. WhereScape RED, too, automates the movement, loading, and transformation of data. It schedules, monitors, and manages data flows between systems. It automatically generates metadata and documentation, but RED is also an agile data warehouse development and management tool. When I talked about business people collaborating with IT people to build BI and analytic apps, they're using RED's drag-and-drop design environment to do that. The developer points, clicks, and builds something, and the users see if it works and if it's to their liking. If it doesn't, if it isn't, they iterate quickly until they get it right.

About the Author

Linda L. Briggs writes about technology in corporate, education, and government markets. She is based in San Diego.

lbriggs@lindabriggs.com