

ISA | Article Index

1 April 2005

Batch babble begone

Standard interface builds common language.

By Ellen Fussell

What happens when every programming engineer solves the same automation problem differently? Complete chaos, according to Dave Chappell, section manager for batch technologies at Proctor and Gamble in Westchester, Ohio. The methods manufacturers use to automate (opening valves, starting pumps, turning on heaters) might get the job done, he said, but the resulting code becomes a "monolithic monster."

But don't be afraid. ISA's SP88 standards committee is coming to the rescue to slay the code conundrum by beginning work on Part 5 of the ISA-88 batch control standards. The committee wants to define standard interfaces between recipe execution systems and control systems—extending the work OPC has done to include the more complex data needed for batch control systems, said Dennis Brandl, president of BR&L Consulting in Cary, NC. The goal is to also apply the standard to non-batch systems—packaging, filling, and other areas of discrete manufacturing. It will provide a standard interface to OEM equipment without requiring customized programming.

The new addition to the standard started out as a way to define how the procedural products, such as procedural sequencing engines, could interact with equipment phases, the kind of automation that exists in the controllers, Chappell said. As standards writers looked at the benefits, though, they found out it was understandable only to those who created that automation.

"Our goal is to apply the ISA-88 modular philosophy and define methodology and a library of component behaviors that can be used at this low level," Chappell said. "Our expectation is that they'll not only be usable in the machine control environment but the continuous processing and continuous converting and batch environment, so our automation engineers can effectively operate in any environment they find themselves," he said.

Language of equipment phases

The ISA-88 standards are broken down to equipment modules and control modules. The definitions for methods used to integrate the procedural, coordinating, and basic control within the modules are unclear and lead to different implementation approaches, thus chaos.

"We're taking the standards to the next level of completeness," Chappell said. Part 1 defined the structures were there and gave general thinking. "We'll refine that thinking to help guide and constrain the way people create their automation. So instead of a monolithic monster, we end up with more manageable modules," he said.

The sequencing engine translates the recipe. In the procedural side, there are recipe phases. In the control world, there are corresponding equipment phases. Those two need to communicate in the same

way, independent of the products they're using. Part 5 will hopefully describe the method for all the equipment vendors and the procedural sequencing engine vendors to communicate. It's supposed to give a common language that says everyone will talk to equipment phases like this.

"If the Siemens engine spoke, say, only German, when you create an equipment phase, you have to program it in German if you want it to talk to a Siemens sequencing engine," Chappell said. "If I have an English speaking Rockwell system, it's the same thing with the Rockwell sequencing engine; I have to program my equipment phases in English. If I want to use the Siemens German-speaking engine, it doesn't know how to communicate." ISA-88 Part 5 will describe for the entire world, in one common language, how everyone will communicate with equipment phases. "Once we've described that, we can go to Siemens and say you can speak German with your platform, but here's how you talk to everyone else in the world." Chappell said everyone could adopt this language as the language of equipment phases.

Procedures and orchestras

A procedure is an ordered list of recipe phases, the high-level description of how to make a product. "This high level doesn't tell you what to do to make the product, just generally how you're supposed to," Chappell said. "In the control environment, the procedures bring together the what to do to create a product." An example would be adding a thousand kilograms of water and then heating it to 100°F. "You must now add your dye (12 kilograms) and stir it for two hours. That's the what to do," Chappell said. "How to make that happen in the real world is where you have to translate that what to do into the how to do it (open this valve, start this pump, monitor this instrument), until you achieve these readings."

Chappell likened the procedure to a conductor of an orchestra who develops the music. The conductor is directing the people with the instruments, in this case, the equipment. "And his music is now communicating to the musicians—the actual equipment—as to when to play their pieces," Chappell said. "For the conductor to communicate with the musicians, there's a developed language of music and the discipline of what the director is trying to communicate and when," he said. "We don't have this standard language in the batch automation industry today. We have the director's score (the procedure) and the musicians (the equipment) who go out and create music (convert the procedure into a product), but there is no common method the procedure uses to communicate with the equipment." It's all customized for different automation applications.

The similarity is when somebody builds a sequencing engine. "They use their own proprietary methods of communicating with instruments that exist in the control environment," Chappell said. "I can't intermix these equipment phases; the instruments exist down in the control environment. The goal is to be able to buy Rockwell control and have it work with a Siemens procedural engine. We can't do that yet," he said.

Reading patterns

One way to think of Part 5 is it will affect the way engineers deal with solutions based on patterns. "There's a part of computer science called design patterns, and the idea is there are patterns you can use with problems," Brandl said. "There are patterns everywhere, in every engineering discipline, on how you build a bridge. And in software, they're called design patterns. You can reuse these patterns on new problems that are similar."

ISA-88 defines design patterns for control. When the committee developed the ISA-88 Part 1 standard, the concept of design patterns didn't exist, Brandl said. But it's been evolving, and in the past five years, it's become an accepted computer science definition. "We're not going to redo the standard based on that," he said. "But it is a way for engineers to look at the standard. It's not just a dry international standard that's impossible to read. It really is telling the best way to approach some problem."

Part 5 of ISA-88 is part of that effort, he said. "Yes, it's definitely a design pattern because it's trying to say how you organize control within pieces of equipment. In this case, we're talking about small pieces of equipment, such as cutters, driers, bagging machines, and cookers, so they can plug and play together. The only way they can plug and play is if they all follow a pattern."

Who needs it?

Engineers don't like these constraints because "they tell us we're taking away their avenue of creativity," Chappell said. "Programmers are creative by nature, and they like to paint a new picture and create something new to solve a problem versus using something that exists. What we're saying here is rather than you going out and becoming creative and showing your prowess with design and programming, here's a solution for this part of your work."

So who will want to use the standard? The people who manage the business will save money, Chappell said. It will reduce the time to design and construct automation in all the equipment. It will also improve the supportability and make possible changing out components that don't perform well with those that do.

"Today when we want to change one component, we have to change everything," Chappell said. "This will offer the businesses more choices. If they only made one little bad decision, they can fix it. The business owners will be dictating that this is the level of automation they're expecting in their equipment."

Also, end users want plug and play on their equipment, Brandl said. "They want to be able to buy a bagger (machine that puts products into bags and seals) and plug that into a conveyor system and plug that into a labeler without having to program it."

Brandl suggested frustrated engineers use the experience of others and apply their creativity to new problems. Some of those problems coming down the road are radio frequency ID tags (RFID). "Think of them as electronic bar codes," Brandl said. "While Wal-Mart is requiring RFID tags on all material shipped to them so they can trace and track all their products, many pharmaceutical companies are applying RFID tags to medicine and drugs in order to prevent fraud." So RFID is one big area that will be showing up to challenge engineers' creativity. Others are product traceability and tracking. "There's more than enough to keep engineers busy," Brandl said.

Building bridges

The ISA SP95 enterprise control system integration committee has been working with the ISA SP88 committee to get experts together to make sure there is no disconnect between the two standards, said Dennis Brandl, president of BR&L Consulting in Cary, NC. "There is a difference in audience between the two standards," he said. Some terminology can be confusing.

The ISA-88 standards deal with the control engineer's view and the process engineer's view of production and focuses on batch industry. The ISA-95 standards deal with the next levels up: the operational views, the plant supervisors, and production managers or operators. ISA-95 also covers a broader range of industries. So standards writers expand on some words used in ISA-88 in the ISA-95 standards. "That was what led to confusion," Brandl said. "We're just trying to make sure there is no conflict."

Process management, a term in ISA-88, deals with controlling resources in a process cell. Inside ISA-95, there's an activity set called production execution management that includes process management, but it also covers things besides process cells, such as production lines and storage areas. "So all we're trying to do is clarify that with a technical report here's where some of these words might be confusing. They appear to overlap, but they're really different concepts," Brandl said. He said there is really no plan to merge the two, but there is a plan to issue a technical report (TR) that says be aware because there are some of the same concepts in ISA-88 and ISA-95, and here's what they're called.

The committee members in ISA SP95 want to point out the models manufacturers apply to batch can also apply to continuous and discrete manufacturing. "The names aren't quite right for those industries, but the concepts and the design patterns, the patterns included inside the standards, are the same," Brandl said. In some industries, there are requirements to have nonstop production, which means it flows all the way through the production facility. "You don't want breaks in production even as you switch from one product or grade to the next grade," he said. "To do that, we found we could apply the concept of product switchover recipes, and we could use the ISA-88 concept of equipment modules and control modules in this discrete manufacturing project." With these models, any engineer looking at these systems will come up with the same solution. The SP95 committee has done the same thing in electronic assembly and the same in chemical production.

"Take food manufacturing," Brandl said. "You always have more products than production lines, so you're always doing switchovers. You want a minimum break between the times you're doing those switchovers. If we apply the ISA-88 pattern of equipment modules and control modules in how we program the PLC code, it will allow us to use recipes to do the switchovers, to do them in a consistent manner. You get repeatability. You don't get variability based on the operator's skill."

Discrete partnering

Work on Part 5 of the ISA-88 standards started out only focusing on how to communicate with equipment phases. Now it has merged with the OMAC packaging machinery work group, which is working toward the same goal as Part 5 but for the packaging industry, said Dave Chappell, section manager for batch technologies at Proctor and Gamble in Westchester, Ohio. As part of this effort, the ISA committee is looking at automation in the discrete world. That is where the packaging machinery is much like the process. "The packaging machinery world is almost discrete in its speed and mechanism, fast and repetitive," Chappell said. "Some people will tell you their pill-making equipment is discrete. OMAC has their parts that address discrete machinery. So we'll leave that up to OMAC," he said. (See OMAC story on page 14.)

The OMAC package workgroup (OPW) is a subset of the Open Modular Architecture Controls Users Group. The mission of the workgroup is to enhance the value of packaging machinery and systems by promoting the use of digital motion control and OMAC guidelines for open control architectures. The workgroup is a work in progress. The first version is primarily informative without imposing particular requirements on users or original equipment manufacturers (OEMs). It will be useful for users, OEMs, and technology providers to begin to use the guideline to normalize vocabulary, establish definitions of architectures, networks, and line types, and to set expectations for forthcoming guidelines.