SECTION

CONTROLS

Today, when custom code is used to interface two products, every time one of the products is upgraded a new interface needs to be written. This can be difficult, costly and time consuming. For software producers, the answer to this problem lies in a concept rooted in the hardware side of the business: plug-and-play.

Plug-and-Play Opens the Door To Interoperability

BY MARTY WEIL

According to Bob Bloom, director of strategic marketing at Norwood, MA-based Intellution Inc., "Cooperative Automation is a business practice as well as a product development practice—it's ultimately a philosophy. On all fronts it represents a commitment to interoperability." The most visible manifestation of such commitment among writers and producers of automation software has been a movement to develop open applications, systems and tools, but the promise of easy communications made by these emerging products has not been matched by the reality of their implementation.

"The idea of 'open' is often presented as 'Here's a programming interface. We publish it. Now you can write some code to it'," says Bloom. "But the practicalities of this approach present some significant problems."

Originally used to describe the ability of hardware to configure itself for operation within a system by simply "plugging itself in," plug-and-play software provides tools that facilitate functionality across applications. "If everyone plays by the same rules and writes to some common interfaces, things will just plug together," Bloom notes, "and a consideration of emerging products and technologies indicates that the plug-and-play world is nearly upon us, if not already here."

Not surprisingly, the technology expected to enable plug-and-play components on a broad scale is being pioneered by Microsoft Corp. of Redmond, WA. The company's Object Linking and Embedding (OLE) operating system provides component-object models that serve as a foundation for binding applications and functions late in the systems process—by integrators as opposed to product developers—thereby allowing the best products or the best parts of products to be incorporated into flexible solutions. The result is the creation of pluggable components, with end users more easily putting together and sharing information across different applications.

But does such plug-and-play really exist in today's marketplace? The answer is yes and no. OLE is a strategy that Microsoft and key industry vendors are committed to, and the broad adoption of OLE (considered a fait accompli within the industry) will result in the componentization of applications and genuine plug-and-play. In this scenario, different applications will communicate in the richest set that they possibly can (i.e., you won't just pass ASCII data like Network DDE does, but will be able to negotiate interfaces that allow for much more comprehensive integration of data between applications). But, while this world is highly anticipated, it is yet to arrive—but a glimpse of it can be seen today in Intellution's Plant TV.

Plant TV is a universal data viewer—a single application under Windows that can connect to and display data from a very wide range of sources. With Plant TV, everyone in an organization can in-

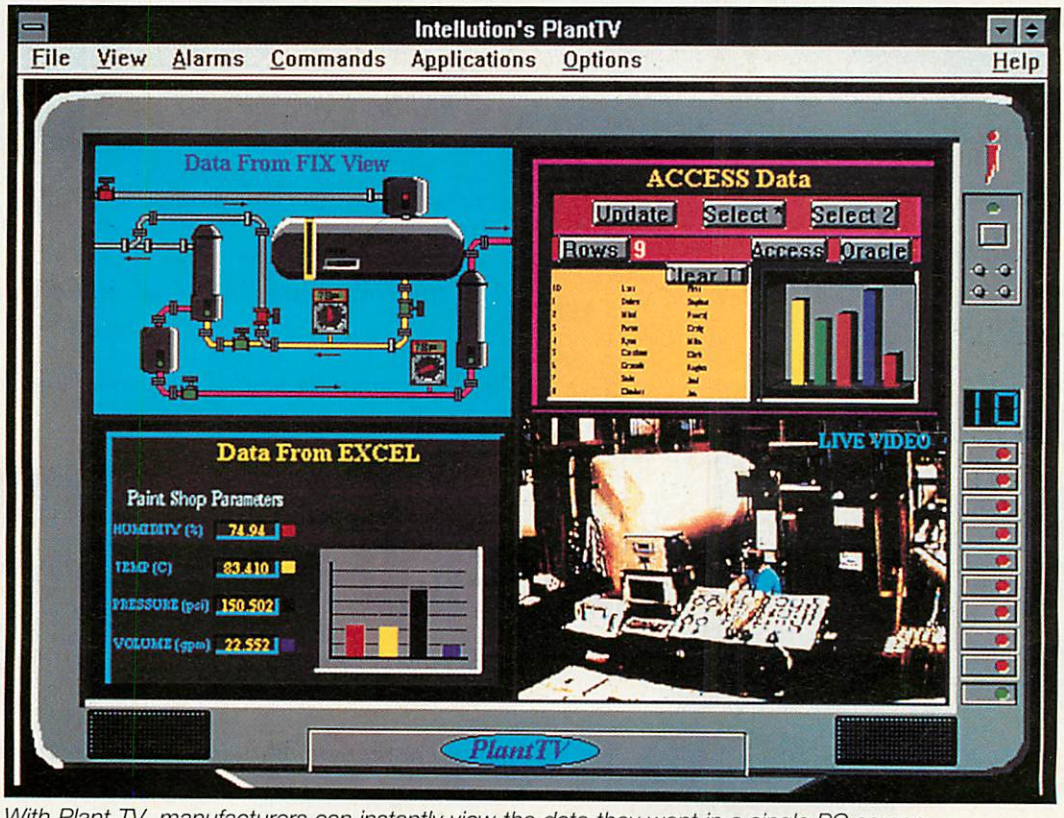

Photo courtesy: Intellution Inc.

With Plant TV, manufacturers can instantly view the data they want in a single PC screen.

Deep inside every overstuffed corporate database, there are priceless insights and brilliant business plans struggling to get out. How do you unearth them? With Alpha Warehouse.™ data warehousing on Digital AlphaServer™ systems. Their 64-bit power muscles through complex queries over a hundred times faster than the competition. Holds up to 14 gigabytes of data in memory. Expands to multiple terabytes of storage. And generally makes 32-bit systems look like 98-pound weaklings. Alpha Warehouse solutions are enormously scalable, too. Their open architecture and ultra-compliant Digital UNIX® platform work effortlessly with other systems. So future growth is all gain, and no pain. Digital's partnerships with key players like Oracle, Informix and Red Brick give you a huge choice of database applications, and familiar access and reporting tools like Prism and SAS. And Alpha Warehouse is already hard at work around the world, doing trend analysis, profitability modeling, risk management, inventory control, supply chain planning and other heavy lifting. Want to know more? You don't even have to gamble a stamp. Just punch out 1-800-DIGITAL, press 4, e-mail to moreinfo@digital.com or see us at http://www.digital.com.*

RECORD-BREAKING PERFORMANCE†
AlphaServer 8400 running Oracle7™ v 7.3
tpmC: 9414
$/tpmC: $316

APPLICATION IMPROVEMENTS*
Complex queries (5-way join): 251.9 times faster
Ad hoc queries (3-way join): 95.4 times faster

digital™

ABI

*Performance information, benchmarked by Oracle Corporation, reflects 32-bit vs. 64-bit versions of Oracle7 running on Digital's AlphaServer 8400 system with Digital UNIX. †TPC results, October 27, 1995. TPC is a trademark of the Transaction Processing Performance Council. The TPC-C benchmark is the industry standard for measuring On-line Transaction Processing (OLTP) performance. ©1995 Digital Equipment Corporation. Digital, the DIGITAL logo, AlphaServer and Alpha Warehouse are trademarks of Digital Equipment Corp. UNIX is a registered trademark in the United States and other countries, licensed exclusively through X/Open Company, Ltd. Digital UNIX V3.2 on Alpha is an X/Open UNIX93 branded product. Oracle7 is a trademark of Oracle Corporation. All other products are trademarks or registered trademarks of their respective companies.

MES Becomes the Ultimate Plug-and-Play Receptacle

The arena most likely to benefit from the plug-and-play evolution is that of manufacturing execution systems (MES). Most companies who have implemented MES systems have not achieved the success they anticipated, while many MES vendors have also struggled in realizing the promise of MES technology.

Consider the paradigm.

The earliest depiction of MES systems were pyramids with plant floor systems at the bottom, MES in the middle, and planning at the top. While this is a very good model, one of the problems that is obvious is that integration of the plant floor with planning now has two boundaries; previously, if you didn't think of MES between those functions, there was a single boundary.

From a technical point of view, the number of interfaces needed has been doubled (and information must pass these boundaries going both up and down the pyramid); so in many cases, the assumption of MES has complicated operations rather than improve them.

This is where the direct functionality of plug-and-play system technology becomes a truly powerful solution, as MES-level solutions are very encompassing—everything from warehousing applications and batching applications to integration of inventory information.

Most current (i.e., open) MES packages provide some aspect of MES solutions, but MES is about bringing together all different levels of information within an enterprise, and therein lies the greatest need for the universal, direct access, which is the ultimate goal of plug-and-play.

At the very least, MES users will need an open system to allow communication across applications, but the best MES scenario would be plug-and-play—as if one universal application were being used, and however functionality is expanded or extended, it seems invisible to the user.

stantly view the data he or she wants in a single PC display screen. Whether the data resides in relational databases, plant floor MMIs, lab and ASCII files, DDE servers, or multimedia (live or prerecorded), Plant TV plugs the employee into the universe of organizational data and plays back what is needed in real time.

It is useful to look closely at relational databases to clearly understand the technical leap from open ideology to practical plug-and-play.

Traditionally, the method for communication with relational databases is Structured Query Language (SQL). One of the problems with Structured Query Language is that even though it is supposedly a standard, every relational database vendor has its own version. Therefore, it's not really a standard, and its facilitation of communications is problematic.

Microsoft has successfully addressed this problem with Open Database Connectivity (ODBC). ODBC builds a standard method to communicate via SQL. So, for example, Intellution supports ODBC. Someone developing with Intellution's sequel module has a tool that communicates ODBC to any ODBC-compliant database—which essentially includes all major relational databases such as Oracle, Ingres, Sybase, Access, SQL Server, and even non-relational databases like dBase and Paradox.

This is an example of genuine plug-and-play (and how Plant TV accesses its relational databases), because it allows any ODBC server to attach to it an ODBC-compliant client, regardless of who it is.

Dynamic Data Exchange (DDE) is another example of this. DDE allows data to be shared among applications, so you can reference something by its DDE name in order to get data without having to know a lot about the source of data.

Companies seriously pursuing the development of plug-and-play applications are well-served by strong system architecture, in addition to support of protocols such as DDE and ODBC.

True distributed architecture systems—where a piece of data is defined only in one location within a system, and where that data is accessible to every computer on the fixed network—facilitate effective plug-and-play implementation by being inherently more direct than architectures where duplicate copies of tags are required throughout a network.

Systems based on the latter structure would have great difficulty in effecting the direct, real-time response of an application like Plant TV, as these architectures would have to redefine all the data.

Corollary developments of the movement to plug-and-play are the user's ability to pick and choose specific elements of an application for current or anticipated needs, and, as an option, the ability to extend applications by embedding user-specific information into them.

Intellution's Dynamos, part of the Plant TV product, are good examples of both developments. Graphic Dynamos are an Intellution technology—objects the user can build within the graphics system and attach specific intelligence to. For example, this intelligence could be specific to a motor, pump or valve—virtually anything a manufacturer would want to model within a factory. Simple prompts are attached to these graphics, which can then be stored for later usage. So, as an engineer builds a system down the line, he can pull from these graphics and simply drop them into the new design—dramatically speeding the generation of process graphic screens.

Database Dynamos, on the other hand, allow users to embed functions into the Intellution database, so that they appear as if they're part of the original system. For example, a user may have a proprietary algorithm that needs to be run. This can easily be embedded into the database, and doing so in this one-step operation significantly reduces configuration time and start-up errors.

Conceptually similar to object-oriented programming, such tools can be plugged in or plugged into, depending upon the user's need. "Customers should begin to expect more from their interface," says Paul Vanslette, director of research and development at Intellution.

Given the technologies now becoming inherent in MMI/SCADA (man-machine interface/supervisory control and data acquisition) software, interfaces historically limited to interaction with programmable logic controllers or some other sensor on the plant floor are now working with virtually anything in an enterprise system.

Perhaps the greatest or most encouraging indicator of how close we are to really integrating the idea of interoperability comes from a public now ready to take it out for a spin. People are coming to accept the elements available on their PCs, an acceptance facilitated by technology's effective response to the speed and power issues that have been barriers to these elements. The message is getting out on a broad consumer level, which helps prepare the industrial marketplace for advanced tools like Plant TV.

Products and pieces of systems will be much easier to tie together, according to Bloom. This process will be accelerated in all arenas (MES, MMI/SCADA, and Data Viewers) by the technologies incorporated in Microsoft's Windows 95 operating system, but especially in its OLE 2.0.

Vanslette's enthusiasm for the latter is typical of the industry: "We feel that the concept of OLE Automation is key to what's going to be happening in the future. We support that today, and we will be supporting that even more in the future. The plug-and-play components inherent in OLE will result in genuine interoperability for the industrial arena. This will enable applications to be integrated and installed much more simply and, more importantly, this will allow users to make best-of-breed selections tailored to their specific environment."

This freedom of choice will foster continuous process improvement, stronger supplier/customer relationships, and stronger enterprises. Perhaps this is something we can all plug into. MA

EDITED BY FRANK E. BENASSI. He can be reached at 212-629-1513, fax 212-629-1559 or e-mail benassi@manet.com.

Intellution Inc.
Plant TV.....RC# 87
Microsoft Inc.
Windows NT.....RC# 88

Smart Factories are created one piece at a time.

Touchscreen displays show real-time operating conditions • PLCs with shared programming language • Flexible, easy-to-install, open networking solutions • Sensors for accurate detection under complex conditions • Machine vision capable of 3600 inspections per minute • In-line inspection solutions yield flawless manufacturing • Temperature controllers with accelerated response from fuzzy logic • Best of all, they are available locally and worldwide. As a total automation solution or one piece at a time. Find out more. Get the SmartFactory Sourcebook. Call **1-800-55-OMRON** or visit our web site at http://www.oei.omron.com.

OMRON®

BE SMART. LOOK TO OMRON.

READER CARD NUMBER 29