SAA Revolution Needs Time For Evolution

The plan is currently pricey and mainframe-oriented; support for client/server, Windows 3.0 and LANs may help

The jury is still out on IBM's plan for providing application standards across platforms. But before the industry can fairly judge the SAA blueprint, it must have time to evolve. Current drawbacks, such as expense and lack of products, may not be an issue over time. With IBM's track record as an industry leader, organizations committed to Big Blue would be well advised to prepare the way for SAA migration.

Much of what is included in IBM's plan for providing application standards across platforms is not new. On the other hand, SAA contains some important, novel concepts.

For example, Common User Access, the "look and feel" style guide for SAA applications, will help create a commonality across all modern IBM applications. It combines the advantage of diverse IBM platforms with the standard look that characterizes Macintosh applications.

Another critical SAA component is Distributed Data Systems Architecture. In the past, the most advanced data management implied the use of RDBMSs. In the future, this capability will be available under SAA, but with implementation over a network of computers. In other words, a single logical database view will be available to applications, independent of where the data is stored. IBM calls this "single-system image." This capability will transcend all of the IBM RDBMSs—as well as the SAA Unix environment (IBM's AIX). The architecture for distributed relational data (DRDA) that will do "single-system image" is open and published.

To date, AD/Cycle—IBM's approach to Case—is the SAA component that has received the most media attention. It is not just another modeling and methodology tool, however. AD/Cycle is a standard underlying platform that will allow a database approach to be applied to application development "objects." AD/Cycle's goal is to allow for the integration of diverse Case tools from different vendors.

THINGS TO WATCH OUT FOR

Expense. SAA hardware is expensive. AS/400s and 3090s are not cheap by today's standards.

Mainframe Mentality. IBM is not really committed to commercial application environments that operate entirely in the AIX (Unix) or LAN environments. While IBM supports many products on these downsized platforms, they must be supplemented with mainframe support and systems for enterprise or strategic systems development.

There are several things that IBM could do to improve SAA. Naturally, the sooner these capabilities are available, the better the computing community's reception for SAA.

Provide Cooperative Processing Tools. Most application development languages provided with SAA are traditional 3GLs. IBM's 4GL, CSP, is also supported and is the designated code generator for the AD/Cycle Case environment. However, none of these languages defines how to build applications that execute across multiple platforms.

Adopt Client/Server Thinking. IBM's thinking about distributed applications has centered on data systems architecture and distributed SQL DBMS. While the concepts of distributed SQL are elegant and sophisticated, the delivery of these capabilities is now planned over the next five years. Client/server SQL solutions are simpler and might be deliverable earlier.

Support Windows 3.0. IBM's workstation standards are built around OS/2EE. This environment requires a large 80386 PS/2, which costs around $10,000. Since few existing PCs have enough power to run OS2/EE, this means substantial added outlays for most potential SAA users. DOS/Windows 3.0 has a GUI that is almost identical to OS/2's Presentation Manager. Support for Windows 3.0 on clients would not require a migration away from DOS and would require less investment hardware.
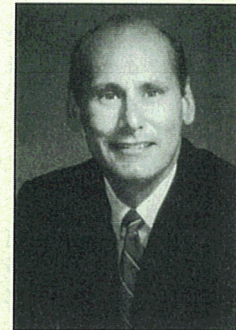
Support Independent LAN Application Environments. IBM's leading LAN product, OS/2EE, includes a GUI, an SQL DBMS and a communications capability. Instead of thinking about a complete application that can run on the PC platform, however, IBM has handicapped the DBMS by not including roll-forward recovery. That limits OS/2EE to read only/decision support applications.

Incorporate Object Development Tools in SAA. Object-oriented languages and tools are paying off in terms of savings associated with software reusability. IBM understands the potential of this technology because it has licensed the NeXTStep software environment from NeXT. It is expected that a version of NeXTStep will appear on IBM's RS/6000 machines.

Much of the recent public criticism of SAA has focused on the notion that three years after its announcement, few firms have committed to SAA compliance. However, important software developments can take the better part of a decade to have major impact. For example, no one would say that IBM's SNA is a paper tiger, yet it took almost a decade to become a communications standard.

SAA should be evaluated within a similar timeframe. For SAA to be successful, IBM's distributed database and Case environments will have to mature. In addition, OS/2EE will have to be widely installed as a basis for CUA. Tune in around 1992/93 and the public's perception of SAA is likely to be much improved.

The real impact of the SAA standards is not likely to be felt until after the turn of the century.■

George Schussel is the president of Digital Consulting, Inc., an Andover, Mass.-based sponsor of computer industry conferences and seminars.