

Parting is painful, but legacy apps can't always be given a good home - evolving mainframe applications for client/server networks - Report on Client-Server Computing

Frank Dodge

Is it possible to evolve of mainframe applications to client-server? And even if you could, should you?

For the past three years, Dr. George Schussel of Digital Consulting Inc. has run scores of seminars aimed at companies wanting to downsize legacy applications from the mainframe to client-server platforms. In his *Downsizing Journal*, Schussel lists several aspects that need to be changed in order to evolve old mainframe applications to the new technologies. First, the application architecture must be changed because applications will be primarily on desktops and data on shared servers. All of the major client-server DBMSs provide capabilities for stored procedures and triggers. Stored procedures of precompiled code that can be called by the application running on the client significantly reduce network traffic. Triggers are routines that are automatically executed as the database reaches predefined conditions. Stored procedures and triggers are available with some host environments, but very few of the old legacy applications used them. Finally, applicaltions must be re-architected to take advantage of the graphical user interface (GUI).

In the light of all this, is it feasible to evolve old mainframe applications to client-server? Schussel unequivocally says not. Even if it were reasonable and possible to port old applications to the new technologies, would you really want to? Many legacy applications currently running on mainframes were designed and developed 15 to 20 years ago. The major goal of downsizing that we have heard from customers is to replace old applications that aren't delivering the functionality they want, and haven't for years.

Porting these systems to a downsized Unix platform not only doesn't make them open or client-server, but doesn't do anything to change the old inadequate functionality. So which part of the investment in legacy systems can be saved? Clearly, it must be that legacy data. Or must it?

Most old financial applicaltions were implemented when business conditions and requirements were very different from today. Often, legacy data is incompatible with new views.

There will be situations when the discrepancy between legacy data and re-engineered business and data models is so great, companies will choose only to summarize old data at some intermediate level and populate a new database as a one-time project. such situations might include those where it is impossible to migrate old software to new technologies in a productive way; or, where the logic in the old programs is so far out of date that there is little, if anything, worth salvaging.

The best course is to define new business and data models, ensuring up-to-date functionality. Companies can't afford to settle for compromise solutions which will only put them farther behind where they ultimately need to be.

COPYRIGHT 1994 Transcontinental Media IT Business Group
COPYRIGHT 2004 Gale Group