PC Database Systems - Present and Future

Philip A. Bernstein
Microsoft Corporation

PC DBMSs are growing up. No longer just single-user B-tree access methods, some products now have remote DB access, query optimization, and multi-user capability. We can expect much more from them soon. What will future products look like? What new technical challenges arise? How can DB researchers contribute?

Successful PC DBMSs can fund a similar amount of engineering as traditional server-oriented DBMSs. Thus, PC DBMSs could evolve quickly into powerful, yet easy-to-use challengers to traditional DBMSs. Could a large number of PC DB servers compete with DBMSs designed for larger machines? In the medium-to-long term, what features will distinguish high-end DBMSs? What fraction of DBMS market revenue will they have?

With tens of thousands of DB users managing independent DBs, "heterogeneous distributed DB" takes on a new meaning. There will be Newton size PDAs to glass-house mainframes in the same transaction. So it won't be a world of cooperating equals. Some DBMSs will be storage providers. Others will have modest local stores, used mainly for personal files and temporary data built during a transaction. How will this affect transaction management and query optimization?

We currently see much customization of PC DBs: variant tables, methods for data access, etc. This heterogeneity makes sharing very hard. How will users cope? What disciplines of metadata management will be needed to make sharing possible? What DBMS features would make this discipline attractive to users?

Soon, PCs will be as connected to DBs as workstations are to FTP files now. Users will expect to search for information, without knowing where the information might be stored. They will expect to run PC applications that can update some of those DBs, by connecting their UI to a server agent that mediates transactions. PC applications and server DBMSs will play in an open network, publishing their services and safely coordinating usage. This presumes standard interchange of data and metadata, standard protocols and well known ways to publish services. Users will see a richer and more unified ocean of data. The PC may present those DBs through some uniform metaphor, a virtual DB.

About 40% of the PC's sold for businesses in 1993 were portables. So PC DBMSs must cope with disconnected operation. Is the ideal paradigm that of Lotus Notes, where independent conflicting work on shared data can be coordinated and resolved at a later time? Mobility is a problem discussed in many recent papers. PCs may also need to run long, speculative transactions where work should proceed while disconnected and completed when communication is re-established.

Commodity PCs will make fine DB servers. They will be bought for single purposes, in quantity to increase capacity and performance, and locally to spread throughout an organization. How will databases evolve to exploit this? Soon, every branch or department will be able to afford local replication of most corporate data used by that office, along with ownership of much original data generated by the office. Big computers at head office will probably be the servers for the head office, not central computing for the whole company.

So, VLDBs in the PC era may be dominated by issues of: replicating data among many peers; caching data local to users; publishing, connecting, and reconciling independent suppliers; intermittently connected servers and clients; ever changing topologies, routes, and population; querying with degrees of confidence and completeness that depend on how far you go beyond local cache and servers; local indexes to remote data; and ensuring that capacity scales linearly with number of servers. It seems certain that powerful commodity PC DBMSs will redraw the DB landscape.

Acknowledgement: The author gratefully acknowledges much help from Pat O'Neil and Tanj Bennett.

Author's address: One Microsoft Way, Redmond, WA 98052. philbe@microsoft.com.

Proceedings of the 20th VLDB Conference
Santiago, Chile, 1994