Adaptive Server IQ Multiplex

from Sybase

An extract from the Bloor Research
Product Evaluation Series

Bloor Research

Fast facts

Sybase Adaptive Server IQ Multiplex (IQ-M) is a relational database aimed at data warehousing environments. The unique feature of this product is that it is predicated upon a column-based approach rather than the row-based approach used in relational databases. This gives significant advantages within a query environment, not least in terms of performance and scalability. In addition, Sybase's Bit-Wise™ indexing is designed to take advantage of the inherent benefits of using columns.

Key findings

In the opinion of Bloor Research the following represent the key facts of which prospective users should be aware:

- Sybase Adaptive Server IQ is a column-based relational database which is intrinsically more suitable for query processing than a row-based approach.
- By using columns, data compression is much easier to implement than in conventional approaches. In fact this is so good that Sybase claims that its database, with all of its indexes, never exceeds the size of the original data. This represents an improvement of many times when compared to conventional treatments.
- Sybase's Bit-Wise indexes, which can also be compressed, represent an effective means of accessing the database. There are a number of other indexing techniques that are supported.
- Both multithreading and 24 x 7 high availability features are available. In particular, separate read and write nodes mean that these procedures can be conducted in parallel, without effecting one another.
- IQ-M can offer significant performance advantages when compared to conventional approaches. In particular, it supports Rcube flat schemas that provide significant advantages over conventional Star schemas.
- It is noteworthy that IQ-M is focused on providing support for as many queries as possible to be running in parallel, rather than being concentrated on using parallelism to optimise the performance of any particular query.
- In a multi-node environment IQ-M still lacks some features that we would like to see, the most important of which is load balancing.

The bottom line

Sybase Adaptive Server IQ Multiplex has been engineered specifically for query processing. This is unlike standard relational databases, which have only been adapted for that purpose. It should come as no surprise then, that it can offer significant performance and total cost of ownership advantages, when compared to rival products. These benefits will be most obvious in environments where the query load cannot be predicted and where there are many ad hoc queries. Such environments are becoming the rule rather than the exception with the growth of Web-based business intelligence processing and eCommerce.

© Bloor Research. All Reproduction Prohibited

Vendor Information

Background information

Sybase web address: www.sybase.com/bi

Product availability

The current version number of Sybase Adaptive Server IQ Multiplex is version 12.4.2. However, it should be noted that this numbering is only in order to keep it synchronised with Adaptive Server Enterprise. In fact, Adaptive Server IQ only first appeared in 1996, with version 11, replacing the earlier Navigation Server.

The product runs under both Windows NT and Unix operating systems. In addition, in a multi-node environment it is possible to combine NT and Unix systems, though this is currently limited to Intel NT processors and Compaq Tru64 Unix systems. This allows the stability of a Unix platform for writing to the warehouse, while NT systems, with their better price/performance, can be used for reading nodes. As far as we know, no other vendor can support this combination of platforms.

It should be noted that Sybase is (successfully) targeting the Application Service Provider (ASP) market with IQ-M.

Introduction

Sybase takes an approach to data warehousing that is fundamentally different from that of the other relational database vendors. This is not simply a marketing distinction based on the fact that its Sybase Adaptive Server IQ Multiplex is distinct from its mainstream database, but goes right to the core philosophy that underlies IQ-M. Sybase has concluded that conventional relational approaches to ROLAP are inefficient and can only provide adequate performance at the cost of a significant, and otherwise unnecessary, investment in additional hardware. It has therefore developed what might best be described as an inverted relational database. That is, it uses a conventional relational structure and a similarly familiar terminology, but is column oriented rather than row oriented.

When we first reviewed Sybase Adaptive Server IQ (as it was then), we stated that, in general, we found the company's arguments in favour of a columnar approach "quite convincing". With the passage of time, the increased demands made upon data warehousing by the increase of Web-derived data and users, and the experience of customers, our opinion has hardened. We now drop the qualification: we are convinced that IQ-M offers a superior approach to those of traditional relational vendors.

Architecture

IQ-M stores data in tables just like an ordinary relational database. However, instead of storing and accessing those tables by row, it does so by column. While this would obviously be inappropriate for a transactional environment, in which a transaction is effectively equivalent to a row, it is entirely sensible within a query processing environment, since queries are generally selected on the basis of defining columns.

© Bloor Research. All Reproduction Prohibited

In particular a major advantage of this column-based approach is that, in effect, the entire database is automatically indexed, in that it is by column that you define selection criteria in a query. In fact, it is not quite as simple as this, since there a number of ways in which IQ-M supports these indexes as columns, which are discussed below.

Another consequence of the use of columns is that IQ-M is much more efficient than traditional approaches when it comes to data compression (according to Sybase, up to five times better). This is because, needless to say, all the data fields in the same column have the same datatype. This means that each column can be compressed with optimal efficiency. By contrast, if you read across a row, each field is likely to be of a distinct datatype. In such an environment it is impracticable to keep changing to the optimal compression algorithm, which means that any compression offered will tend to be of the lowest common denominator variety.

Sybase argues that the columnar nature of Sybase IQ provides so much better performance than normal ROLAP approaches, that it does not need to support hardware parallelism in the same way that its major competitors do or that Sybase Adaptive Server Enterprise (for which Bloor Research has written a separate product review) does. In particular, the company points out the problems that are associated with the data partitioning needed to support hardware parallelism. While it is certainly true that partitioning, no matter how implemented, can create problems (not least of which is the additional maintenance that usually ensues), it nevertheless opens the way to substantial performance improvements. Sybase would further argue that this is simply a compensating mechanism for the poor performance inherent in a row-based approach.

While there is a lot of truth in Sybase's arguments, this does not mean that Sybase eschews all forms of data partitioning. However, rather than implementing horizontal partitioning, it instead implements vertical partitioning. That is, partitioning by column rather than by row. One of the advantages of this approach is that partitions can never become unbalanced, since there will always be the same number of fields in each column of a table. This significantly reduces the maintenance requirement of managing partitions and should eliminate the database re-organisation that may become necessary when partitions become unbalanced and start to impair performance.

Supported Schemas

IQ-M supports conventional relational schemas including both the normalised schemas used for transactional processing, as well as the star, snowflake and constellation (a collection of stars) schemas that are commonly used in data warehousing. However, its columnar approach opens up the possibility of using what Sybase refers to as Rcubes. These represent a flat schema architecture.

The distinction between a flat schema and a constellation schema is illustrated in Figure 1. As can be seen, it is much simpler to use Rcubes than to employ multiple star schemas. To be specific about this particular example: there are 8 fewer tables (and, therefore, 8 fewer Extract, Transform and Load operations to be completed) and there are 25 fewer joins.

However, perhaps the biggest impact of using Rcubes is that they are simply much easier to understand and, most particularly, that they offer much more efficient navigation for cross-functional purposes than constellation schemas.

© Bloor Research. All Reproduction Prohibited

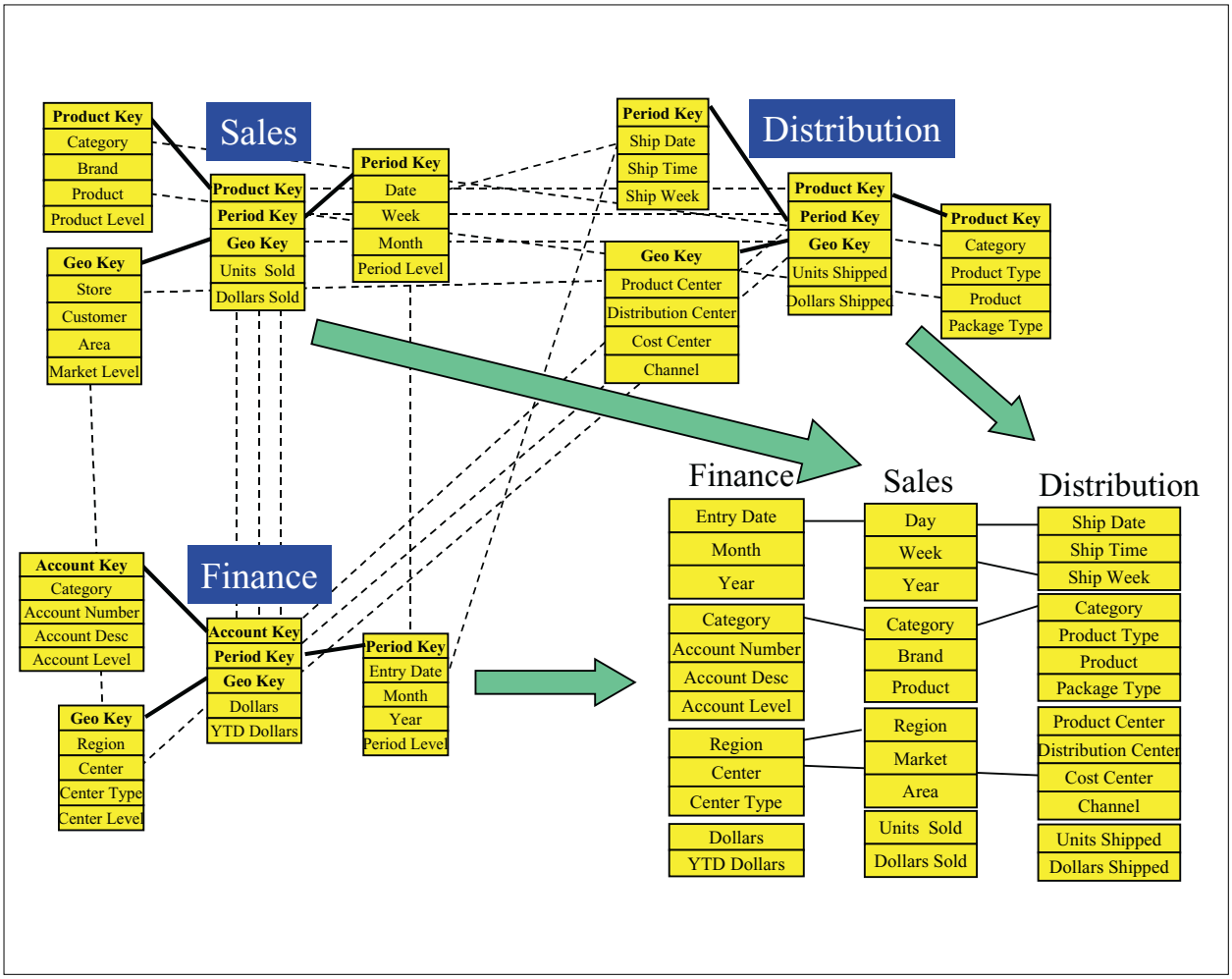

Fig. 1: Different schemas

Indexes

IQ-M in practice uses four different indexing techniques: low cardinality indexes, Bit-Wise indexes, high group indexes, and column stores. These need some further explanation.

Low cardinality indexes use a process known as tokenisation. Using this process the data is converted into a token and then the tokens are stored rather the data. This is particularly useful for reducing the quantity of redundant data. Suppose, for example, that you are a supplier with a large customer base throughout the UK. For each customer you would have to store the customer's address. This would mean a very large duplication of county names. So, rather than having hundreds of instances of "Banffshire" for example, you might replace each county with a numerical value. So, as Banffshire is the 5th county in the UK in alphabetical order (after Aberdeen, Armagh, Avon and Ayrshire) and it might therefore be assigned the value 5. Where a column consists of a numeric value anyway, that value can itself be used as the basis for the tokenisation. Once the tokens are established (which will be an automated process), a bit-mapped index is created to reference these tokens.

© Bloor Research. All Reproduction Prohibited

Tokenisation is particularly useful when there are a large number of replicated data entries. Typically, this will apply when there are a limited number of possible data values. This is why Sybase refers to these as low cardinality indexes since they are typically only used for fields that have less than 1,500 unique values.

For high cardinality fields, where the number of possible values exceeds 1,500 (for example, monetary values), Sybase uses a patented technology known as Bit-Wise indexing. These are particularly useful where you want to combine calculations with range searches for example "what is the total revenue and number of units sold, where the price was less than £50".

High group indexes are, in fact B-trees. However, the principle here is that you only define these indexes when you know that several columns are likely to be used in a group, in particular when you want to combine low and high cardinality searches. An example here might be an enquiry about product item sales and value (high cardinality) by store (low cardinality).

Finally, the fourth indexing type is simply the column store itself. If you know that you are always going to retrieve an entire column of data then the fact that storage is columnar means that you can project that column into a report or enquiry without having to explicitly define any index at all. This is useful, for example, in "where" clauses.

A number of extended facilities are supported to allow the use of these indexes in a variety of circumstances. These include index compression to reduce disk (or memory – bitmaps may be cached) requirements, the ability to use different types of index in combination, and the facility to filter the bit arrays using Boolean operators such as AND and OR. These features mean that the indexing in IQ-M overcomes a number of the traditional drawbacks of bit-mapping, namely that they are not suitable for joining tables or aggregating data.

Database operations

The way that IQ-M is used, in practical terms, is illustrated in Figure 2.

As can be seen, IQ-M includes an SQL API that allows SQL-based access. This is SQL-95 compliant and is the same SQL that is used in Sybase Adaptive Server Anywhere and (with a few exceptions) is also compatible with the syntax employed in Sybase Adaptive Server Enterprise (that is, T-SQL). It also supports both ODBC and JDBC call-level interfaces. Alternatively, IQ-M also provides Java capability and this language can be used for writing stored procedures and for creating user-defined functions.

Enterprise Connect is the name of a middleware family of products that is reviewed in more detail in Bloor Research's "Sybase Database-Oriented Middleware". Briefly, the different products within this family provide interoperability between Sybase products and other popular database products. As may be imagined, there is a range of options within the family and you obviously deploy only the ones that you need, both to support combined queries which cross IQ-M and external databases, and to populate IQ-M databases in the first place.

Enterprise Connect includes special features for supporting Oracle and DB2 databases that are worth noting. In particular, you can import DB2 and Oracle table definitions into the Sybase Catalog to assist in the interoperability process. This is especially important for Sybase as it is not just

© Bloor Research. All Reproduction Prohibited

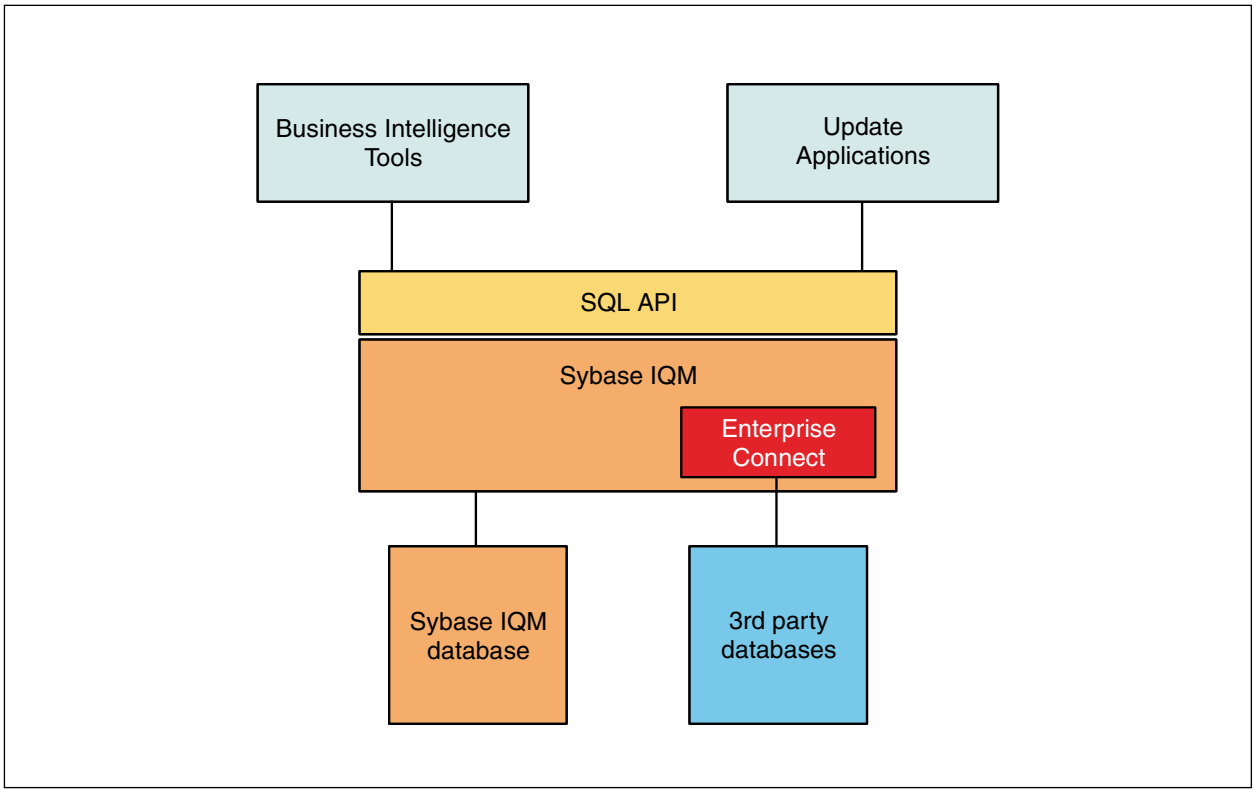

Fig. 2: Operation of IQ-M

aiming IQ-M at existing Sybase clients but also at other mainframe users who may prefer the IQ-M paradigm.

IQ Multiplex

The Multiplex component of IQ-M adds the ability to support multiple SMP machine nodes within a single Sybase IQ environment. This is illustrated in Figure 3.

Here, four nodes (there could be any number) are connected to a single IQ-M database (again, there could be many of these, including mirrored options for 24x7 operations) by means of a fibre channel. One node must be designated to own, manage and update the database, while all the other nodes have read-only access to the database. Since there is only ever one write instance there is never any need to lock records, so there is no contention between the read-only instances.

Should any node fail, including the updating node, you can switch users or responsibilities to another node. However, this is not automatic and there are no in-flight failover mechanisms. Of course, failover is supported in hardware terms but any active queries or updates would have to be restarted. There are load balancing capabilities across nodes, though this is not automated but, rather, is under the DBA's control. This has the advantage that the DBA can define dynamic resource allocation based upon business needs.

© Bloor Research. All Reproduction Prohibited

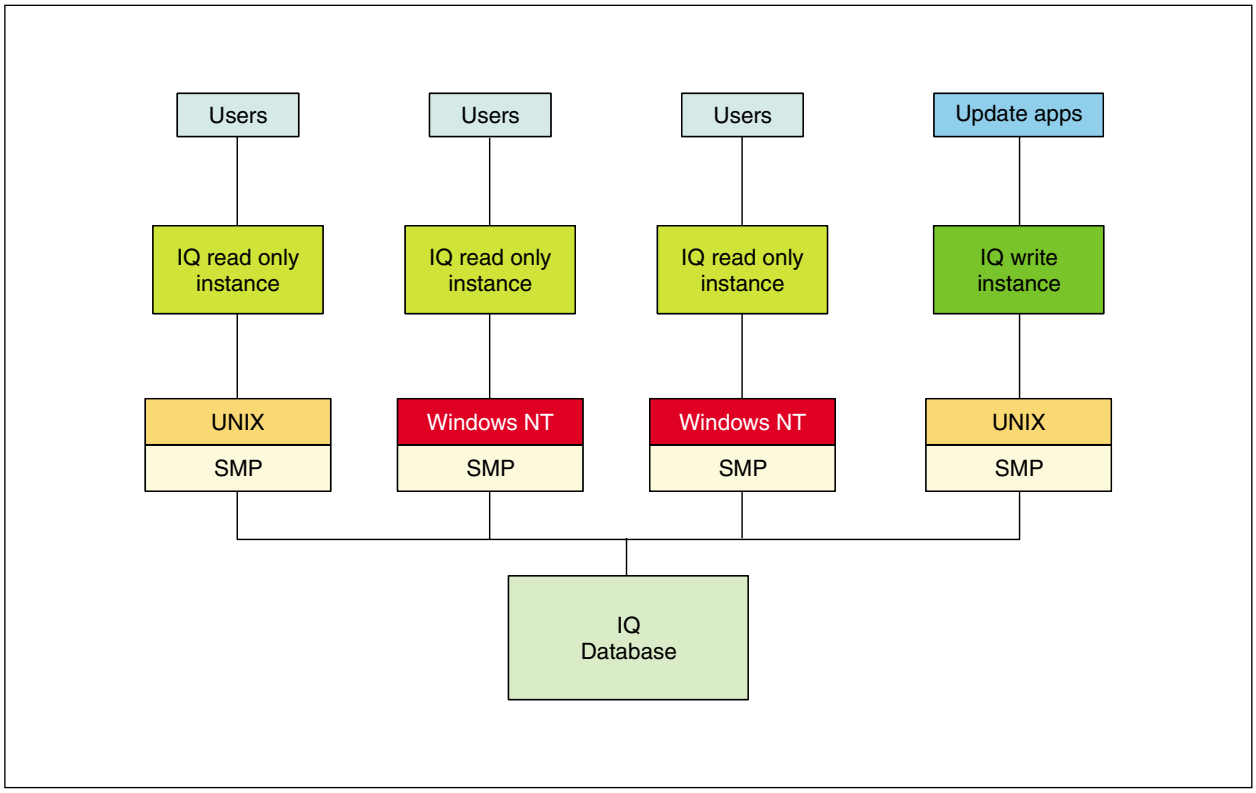

Fig. 3: Multiple SMP nodes

It is worth noting Sybase's strategy here, as it is at odds with most of its competitors and arguably determines the circumstances in which Sybase IQ will be most suitable. Most suppliers have focused their efforts, particularly with respect to parallelism, primarily on improving the performance of individual queries. The basis for this is that if each individual query performs better then this will also reap benefits in terms of the number of concurrent queries that can be supported. However, this is not necessarily valid. It is relatively easy, for example, to see how a parallel database might use data partitioning in order to optimise a particular query but, at the same time, cause a deterioration in the performance of a second query.

Sybase's stance, on the other hand, is that its product is intrinsically designed for individual query optimisation. Thus it leaves its parallel facilities, as instanced by the Multiplex component, to focus on supporting multiple queries rather than enhancing individual query performance.

Finally, it should be noted that IQ-M supports in-flight maintenance operations. During database maintenance, a query user does not see any updates that take place during that session (because of the separation between read and write nodes) but only when he re-connects to the database in a subsequent session.

Scalability

While it has always made much of IQ-M's performance advantages, Sybase is particularly targeting the Very Large Database (VLDB) market. The multi-node architecture which, according to

© Bloor Research. All Reproduction Prohibited

Sybase, offers near linear scalability, has already been discussed but there are other features worth noting. For example, the product employs lightweight multiple operating system threads that underlie each process. This multithreading significantly reduces processing and memory overheads.

With its 64-bit processor support, IQ-M can genuinely claim to support VLDB systems with size limitations that include:

- 281 trillion rows;
- 28 gigabytes of main memory cache;
- 9 petabytes of data;
- 16,000 columns per table;
- 16 million terabytes of addressable main memory;
- 32 kilobyte character fields.

It seems to us that it is unlikely that anyone will wish to breach these limits in the near future.

© Bloor Research. All Reproduction Prohibited

Summary

In our view there is a lot to be said in favour of the Sybase IQ-M approach. It will be particularly suitable for ad hoc and partially ad hoc query environments where neither the type of query nor the resulting load can be predicted. Moreover, there is a substantially lower maintenance requirement with a column-based approach, particularly as Rcubes are used, which should prove attractive even in traditional query environments.

Add these performance features to the scalability addressed through Sybase's focus on VLDB users, and the arguments in favour of IQ-M begin to look not just convincing, but compelling.

© Bloor Research. All Reproduction Prohibited

Copyright Information

An extract from the **Bloor Research** Product Evaluation Series. All rights reserved. No part of this publication may be reproduced by any method whatsoever, without the prior written consent of **Bloor Research**.

Bloor Research is Europe's leading IT research and publishing organisation. Research is available on many subjects such as Data Warehousing, RAD, 4GLs, Development Tools, Client/Server, CASE, Software Testing, Databases, Object Databases, Parallel Databases, Networking, Computer Hardware, IT and IS Strategy, Desktop Strategy, Rightsizing and Object Orientation.

Bloor Research
Challenge House, Sherwood Drive, Bletchley,
Milton Keynes, MK3 6DP,
United Kingdom
Telephone: +44 (0) 1908 625100
Facsimile: +44 (0) 1908 625124
E-mail: mail@bloor-research.com

Web Site: www.bloor-research.com

L01585