The TimeIQ Program

Presented by

Howard Bomze

FAME Information Services, Inc.
Challenge

Evolve FAME’s technology while protecting our customers’ investment in current development projects and operational software and provide them a smooth transition path from our current technology to the new technology.
What changed since the last conference?

▲ Potential competitors became partners:
  ▲ Oracle, Informix, Sybase

▲ Internet technology became the central element of data distribution and presentation strategy for a majority of our customers

▲ Java burst on the scene as a viable, platform-independent programming language and application distribution mechanism
Goals for the New Technology

- Interface to multiple database engines
- Enable distribution of analytical processing between:
  - Server
  - Applications Processor (optional)
  - Client (thin or fat client)
- User extensible data models and analysis
Goals for the New Technology

▲ Provide component based visual application development tools
▲ Make it Web enabled
Technologies Utilized

Emerging database technologies:

- Object-Relational DBMS partners:
  - Oracle
  - Informix

- Object-Oriented DBMS’s

Interface to relational DBMS’s for small scale and legacy time series data storage
Technologies Utilized

▲ Object-Oriented development languages:
  ▲ C++
  ▲ JAVA

▲ CORBA object request broker for:
  ▲ distributed processing
  ▲ language independence between code layers
  ▲ machine architecture independence
Technologies Utilized

- JAVA Beans based presentation components:
  - Leverage 3rd party offerings and make them time series enabled
TimeIQ Model

- Oracle DB
- FAME TimeSeries DataCartridge
- FAME TimeSeries DataBlade
- Informix DB
- O-O DBMS
- Generic RDBMS

Data Model, Analysis and Manipulation

Application Framework, Presentation and Business Components

SQL3 Applications
Potential advantages brought to FAME by other database engines

- Full DBMS management facilities
- Replication
- Multi-threading, SMP and MPP support
- Finer grained transactions and roll-back
- Enhanced locking
- Additional navigation methods
- Finer grained security (not necessarily at the observation level)
But First

- They must perform at a comparable level to current FAME database technology
- They must be as reliable as the FAME database engine
Persistence Layers

▲ Provide a common core interface to the Middleware layer

▲ Have extensions to the interface to take advantage of features unique to a database type:
  ▲ code not portable if extensions are used
  ▲ similar to ODBC model
Object Relational Databases

Object relational databases are an extension of relational database technology that allow the user to define complex or abstract data types along with methods to utilize them within SQL queries. The user defined type may be a combination of base data types or a binary object to which user defined methods add structure and properties.
Object Relational Databases

Object Relational databases are expected to dominate the DBMS market within the next several years as the following occur:

- ORDBMS products mature - stability and performance
- SQL becomes standardized to handle ORDBMS - SQL3
- Clients needs for handling of complex data types increase
Object Relational Databases

FAME will provide object-relational add-ins and schemas to model the data for access by:

- FAME’s TimeIQ time series components
- SQL (enhanced for ORDBMS)
- 4GL
- C HLI

A schema has been designed to map FAME databases into ORDBMS/RDBMS environments
Distributed Middleware

▲ FAME will implement an OO library superset of the 4GL (except presentation):
  ▲ C++ and/or JAVA depending on performance of JAVA

▲ Data models, analysis and manipulation for:
  ▲ Time series
  ▲ Case series
  ▲ Scalars
  ▲ Additional data types
Distributed Middleware

Distributed processing model based on CORBA technology. Computation can occur on any combination of:

- the server (DataBlade, DataCartridge etc.)
- an on-line application processor (OLAP)
- the client

Users will be able to extend the middleware layer with their own analysis functions
Application Components

- The components and framework will be JAVA and JAVA Beans based
- It will consist of:
  - data presentation components (e.g. charting, reporting, publishing, navigation)
  - business components to capture domain expertise
  - An application framework to tie the components together and build applications
Development Tracks

▲ Development is being carried out in several tracks that are implemented in parallel with each other.

▲ Each track results in deliverables that add functionality to the FAME environment.
Development Tracks

▲ Track 1 - Build persistence layers for several database engines and modify C HLI and 4GL to utilize persistence layer

▲ Track 2 - Build O-O interface to 4GL and C HLI:
  ▲ Use that interface to develop application components. The interface will be modeled after the Middleware interface

▲ Track 3 - Build the Middleware layer
TimeIQ - Track 2

Oracle DB
FAME TimeSeries DataCartridge
Informix DB
FAME TimeSeries DataBlade
Oracle DB Persistence
Informix DB Persistence
Oracle DB
Generic RDBMS
Oracle DB Persistence
Informix DB Persistence
Object-Oriented DB Persistence
FAME DB Persistence
FAME DB Persistence
FAME DB
CHLI
4GL
4GL
ORB
DB
DA
ORB
O-O I/F
O-O I/F
Application Framework, Presentation and Business Components
Data Model, Analysis and Manipulation