A key technical and business challenge for many large financial institutions is to knit together their many disparate data sources, databases and systems into one consistent framework that can meet the future demands of the business, its clients and regulators. This paper outlines how the above issues can be addressed through Xenomorph’s analytics and data management solution, TimeScape.
# Table of Contents

The Business Issue ............................................................................................................................... 3
The GoldenCopy Approach .................................................................................................................. 3
The Middleware Approach .................................................................................................................. 3
Addressing the Need ........................................................................................................................... 4
Consistent Data Access ....................................................................................................................... 5
Easy Integration Framework ................................................................................................................. 5
All Data Types ..................................................................................................................................... 6
Extendable Data Representation ......................................................................................................... 7
Fine-Grained Control .......................................................................................................................... 8
Easy Data Access ................................................................................................................................. 9
Easy Security Identifier Translation .................................................................................................... 9
Easy Data Analysis .............................................................................................................................. 10
Support for Derived Data .................................................................................................................... 12
Summary ............................................................................................................................................. 13
Appendix 1 – TimeScape Business Flow .......................................................................................... 14
Appendix 2 – TimeScape Logical Architecture .................................................................................. 15
The Business Issue

A key business and technical challenge for many large financial institutions is to knit together their many disparate data sources, databases and systems into one consistent framework that can meet the future demands of the business, its clients and regulators. A typical scenario for how data can be difficult to access is shown in Figure 1 below.

![Figure 1 – The Data Unification Challenge](image)

The GoldenCopy Approach

One approach to addressing the above problem is to create a GoldenCopy of all data needed by the business. This approach, where data is stored within a centralized data warehouse, offers the key advantages of data consistency, quality and auditability. However, achieving a centralized GoldenCopy can be very difficult due to the variety of legacy systems involved. Also, a centralized GoldenCopy is often siloed alongside a particular type of data, a particular type of business function or a particular business unit. Its centralized approach does not deal with the management of data once it is delivered to downstream systems (i.e. once it has ‘escaped’ the confines of the centralized system). Additionally, traditional data management approaches are mostly batch or over-night based, with poor end-user interactivity and lacking in real-time analytical/business functionality.

The Middleware Approach

Often a middleware or ETL-type approach is implemented as an alternative to a centralized GoldenCopy, particularly in an environment with many existing legacy systems and sources of data. While such solutions have their place, they fail to go far enough from a business point of view; they deliver a technical solution to the problem but often do not have a business-focused way of presenting data and allowing more complex business objects to be analysed.
Addressing the Need

What is needed is a more generalized and business-focused approach, able to deal with disparate and distributed sources of data and provide easy end-user interactivity whilst maintaining data consistency, quality and auditability. We could categorise such a system as a form of Distributed GoldenCopy, which would exhibit the following characteristics:

- **Consistent Data Access** – A normalised data model across all underlying systems that brings together all data regardless of physical location or representation.

- **Easy Integration Framework** – Connecting to a range of underlying data sources, databases and systems easily and quickly.

- **Support for All Data Types** – Offering support for both simple and complex data types, with no artificial barriers between support for reference data, market data or more complex model data.

- **Extendable Data Representation** – Making it easy and quick to add new objects and field types.

- **Fine-Grained Control** – Enabling user data access permissioning to a very detailed, granular level.

- **Easy Data Access** – A business-focused data model that represents data in a way that is easily understandable by both business and technology users alike.

- **Easy ID Code Translation** – Enabling a consistent data universe where data objects are easy to lookup and translations between the many naming conventions are easy to handle.

- **Easy Data Analysis** – An analysis framework that is powerful, extensible and simple to understand.

- **Support for Derived Data** – Potential to extend the system with analytics and models to enable the creation and management of derived data regardless of data source.

The rest of this document illustrates how the above can be achieved using Xenomorph's analytics and data management system, TimeScape.
Consistent Data Access

Figure 2 below illustrates how multiple systems, databases and datafeeds can be connected into TimeScape to present a single common view of the data for all downstream users and systems. The database driver components are discussed in more detail in the section below, but are basically dynamic links that map and normalise existing systems into conforming to the TimeScape data model. TimeScape QL+ provides a powerful business analysis framework for accessing these normalised datasources. Using Excel, or a wide range of applications and programming interfaces, the end user can consume any data that has been exposed into the system.

Easy Integration Framework

TimeScape has many component parts. Two fundamentals of the system are its data model for representing data and its vector database engine, XDB, as an implementation of this data model. It is possible to unplug XDB from the rest of TimeScape and use TimeScape’s Data SDK to create “live” data drivers that connect existing databases and datafeeds directly into the system through a thin normalisation layer. These are illustrated figuratively as the orange blocks in Figure 2 above. Each block contains the mapping code necessary to represent the data source within TimeScape. Once created, data from an external system is directly accessible through TimeScape without the need for storing additional copies of the data. Additionally, if new data is made available in an external system the driver connections can be designed to recognise these additions and publish them into TimeScape, avoiding the need for costly and time-consuming integration work as systems inevitably change and develop.
All Data Types

TimeScape offers support for both simple and complex data types, presenting the data in a manner that is easily accessible by both business and technical users alike. “Simple” data types supported include the usual unitary data types found in database implementations, such as integers, real numbers, text, binary data, dates, booleans etc., but also more complex data types such as:

- **Reference** - This uniquely identifies an object within the data model and can be used to create links between objects in the same database, or even across databases.

- **Matrix** - Numerical matrix structure held as a 2-dimensional array of real numbers.

- **Excel Data** - A 2-dimensional rectangular region of an Excel spreadsheet, filtered to exclude any error values that were present during save.

- **Formula** - A Formula Grid defining a data result that contains no formatting information - this is analogous to a stored procedure within SQL.

- **Formula Grid** - A spreadsheet object allowing complex calculations to be performed based upon formulae and queries that are executed to form a data result. This type includes formatting information and is used as the basis of reports and file exports.

- **Weblink** – A live URL to a website or web service.

- **List** - Named, multi-column table containing any of the TimeScape-supported data types. These are very powerful data types allowing the user to easily construct any list-like structure and store it over time e.g. composition of a yield curve, portfolio etc.

When creating a data attribute for an object in TimeScape, the above data types can also optionally be assigned with dimensions of being:

- **Historic** (at daily, tick and when changed frequencies)

- **Multi-Sourced** (so that the same property can take multiple values dependent on source)

Combining the Matrix datatype with historic storage over time enables complex datasets such as volatility surfaces to be stored as illustrated in Figure 3 overleaf. As another alternative example, a yield curve can be defined as a historic list of references to interest rate instruments such as cash rates, futures, swaps and bonds.
Extendable Data Representation

Many data management solutions based on centralised data warehouses have a very rigid data model that is hard to extend. In contrast, many data management systems that implement ETL-type approaches have no real data model to base data analysis upon. In between these approaches, TimeScape offers a data model that is ready to run out of the box with predefined templates for instrument data. Moreover, these templates can be extended quickly and easily to add new properties for existing instruments and indeed new types of instrument from scratch. It is this flexibility that enables TimeScape’s data drivers - illustrated in Figure 2 - to cope with changes in the data and data models of external systems and databases.
Fine-Grained Control

Given the variety of systems that can be integrated into TimeScape, it is essential to have an approach to user access control that is granular enough to cope with the business and technical needs of each institution. Figure 4 below shows a representation of the data model exposed to all end users of TimeScape. At the highest level is the database, followed by the category of instrument (equivalent to class or object type), where each category contains items (object instances) that in turn own data properties (data attributes, optionally historic or multi-sourced).

![Diagram of data model]

Figure 4 – Navigating the Security Hierarchy of Data

The standard user/group access levels to databases within TimeScape are:

- None
- Structure Read
- Data Read
- Data Write
- Data Approve
- Structure Write
- Permission
- Administrator

In addition, when fine-grained permissioning is enabled in TimeScape, the above permissioning can be applied at a category (type of instrument e.g. Bond), property (data attribute e.g. Close) or datasource (e.g. Bloomberg) level. So in Figure 4 a user group could be given access to only closing prices for bonds with data sourced from Bloomberg.
Easy Data Access

TimeScape presents data in an easily recognisable format for end users, a format that is abstracted away from any particular database representation and is focused on business usability. Objects in the system (termed “items”) represent easily recognisable entities such as securities, portfolios and companies. From an interface point of view, it is easy to load a price for a security contained in multiple databases with statements such as:

```
Item(“KDB”, “IBM”).price
Item(“AssetControlDB”, “IBM”).price
Item(“GoldenSource”, “IBM”).price
Item(“Oracle”, “IBM”).price
Item(“FAME”, “IBM”).price
Item(“ReutersDB”, “IBM”).price
```

So such access is based upon simply knowing the data store, the security ID and the data attribute required. If desired, the above logic could be extended further to remove the need to express which specific data store is being accessed (based on a global preference list). However, when implementing such a scheme, it is also possible to implement queries. For instance:

```
Item(“ReutersDB”, “IBM”).price.explain
```

This would “explain” to the user which data store the security is stored in and, optionally, what data the price is recorded on and from which data source.

Easy Security Identifier Translation

Security identifiers are defined as “Codes” (e.g. “AAPLO”) in TimeScape, with each Code belonging to a particular CodeType (e.g. “Reuters”). Code mappings for instruments represented in TimeScape can be handled in a variety of ways; the alternatives depend very much on what data is available. For example, a mapping table that contains a variety of code types for instruments can include both external ones (RIC, Sedol, BBG Ticker etc) and internal ones (RDR, Unique IDs etc). TimeScape can translate between codes and the mappings can also include the location of the item if required.

For example, the following query will load the Reuters ASK tick series of a 2yr EUR Swap rate using a Bloomberg code for an instrument in an Oracle database:

```
Item(“Oracle”, “EUSW2 Curncy”, “Bloomberg”).AskTick(‘Reuters’)
```

Whereas this query will return the Reuters RIC of the instrument given the Bloomberg code:

```
Item(“Oracle”, “EUSW2 Curncy”, “Bloomberg”).Code(‘Reuters’)
```

It assumes that the mapping table in the Oracle database is populated, which is another area where TimeScape’s datafeed connectivity can help in maintaining and managing codes.
Easy Data Analysis

TimeScape QL+ is Xenomorph’s powerful business-focused query language that represents data in such a way as to be easy for both business and technology users to understand. For example, to calculate the 30 day moving average of the difference in yields exhibited by two bonds you would do something as a simple as:

\[
\text{MovingAv}(30) \left( \text{Item("BondDB","US370442AY11").yield} - \text{Item("BondDB","US233835AA55").yield} \right)
\]

QL+ inherently understands time and time series analysis, and through the “?” operator is able to inform a user of the types of analysis possible for a particular item in TimeScape. For example this query:

\[
\text{Item("BondDB","US370442AY11").}? 
\]

would return the following result, showing all the types of functions that are applicable to a yield property:

<table>
<thead>
<tr>
<th>Function or Property</th>
<th>Name</th>
<th>Data Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>FUNCTION</td>
<td>ABS</td>
<td>Variant</td>
</tr>
<tr>
<td>FUNCTION</td>
<td>ACOS</td>
<td>Variant</td>
</tr>
<tr>
<td>FUNCTION</td>
<td>ALIGN</td>
<td>Variant</td>
</tr>
<tr>
<td>FUNCTION</td>
<td>ANNSTDDEV</td>
<td>Time Series</td>
</tr>
<tr>
<td>FUNCTION</td>
<td>AVEDEV</td>
<td>Variant</td>
</tr>
<tr>
<td>FUNCTION</td>
<td>AVERAGE</td>
<td>Variant</td>
</tr>
<tr>
<td>FUNCTION</td>
<td>BETA</td>
<td>Time Series</td>
</tr>
<tr>
<td>FUNCTION</td>
<td>Cells</td>
<td>Variant</td>
</tr>
<tr>
<td>FUNCTION</td>
<td>COLCALC</td>
<td>Variant</td>
</tr>
<tr>
<td>FUNCTION</td>
<td>COLSUM</td>
<td>Variant</td>
</tr>
<tr>
<td>FUNCTION</td>
<td>Columns</td>
<td>Variant</td>
</tr>
<tr>
<td>FUNCTION</td>
<td>CORREL</td>
<td>Time Series</td>
</tr>
<tr>
<td>FUNCTION</td>
<td>COS</td>
<td>Variant</td>
</tr>
<tr>
<td>FUNCTION</td>
<td>COVAR</td>
<td>Time Series</td>
</tr>
<tr>
<td>FUNCTION</td>
<td>CROSSVOL</td>
<td>Time Series</td>
</tr>
<tr>
<td>FUNCTION</td>
<td>DIFFERENCEOUTSIDELIMITS</td>
<td>Time Series</td>
</tr>
<tr>
<td>FUNCTION</td>
<td>EXP</td>
<td>Variant</td>
</tr>
<tr>
<td>FUNCTION</td>
<td>FAILURES</td>
<td>Time Series</td>
</tr>
<tr>
<td>FUNCTION</td>
<td>FIT</td>
<td>Matrix</td>
</tr>
<tr>
<td>FUNCTION</td>
<td>FLATS</td>
<td>Time Series</td>
</tr>
<tr>
<td>FUNCTION</td>
<td>GAPS</td>
<td>Time Series</td>
</tr>
<tr>
<td>FUNCTION</td>
<td>GEOMEAN</td>
<td>Variant</td>
</tr>
<tr>
<td>FUNCTION</td>
<td>KURTOSIS</td>
<td>Time Series</td>
</tr>
<tr>
<td>FUNCTION</td>
<td>LEN</td>
<td>Long Integer</td>
</tr>
<tr>
<td>FUNCTION</td>
<td>LN</td>
<td>Variant</td>
</tr>
<tr>
<td>FUNCTION</td>
<td>LOG10</td>
<td>Variant</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

Table 1 - Context Sensitive Help for a Bond Query
Here is an example illustrating how QL+ also knows how to convert between different frequencies of data:

\[ \text{Item(“EquityDB”,”VOD.L”).tradeprice} \]
\[ .\text{PAnalysis(“10s”,“,”high”, “low”, “count”, “volatility()“)} \]
\[ \text{LOAD = [DATA_SOURCES = Bloomberg : Reuters]} \]

The above would return values for the high price, low price, number of data points and volatility for each 10-second period bucket applied to the intraday price history of Vodafone. It also illustrates a QL+ data rule which allows data preferences, alignments and data filling rules to be expressed at runtime.

Moving on to a multi-instrument example, in the query below the \[ \text{Database(...).Items(...).Where(...).Values(yield)} \] sections return multiple historic yield time series for those bonds within each ratings class. The ‘ColCalc’ operator is then used to “Average” these time series to produce a single average yield series through time. Finally, the results for one ratings class are subtracted from the other to produce the historic average yield spread, which is displayed in Figure 5.

\[ \text{ColCalc(Database(“BondDB”).Items(“Corp Bond”)} \]
\[ .\text{WHERE(Rating = “AA-“).VALUES(yield), “Average“)} \]
\[ - \text{ColCalc (Database(“BondDB”).Items(“Corp Bond”)} \]
\[ .\text{WHERE(Rating = “AAA”).VALUES(yield), “Average“)} \]

While TimeScape QL+ is an easy and powerful language for data analysis, it should be emphasised that a client’s own analytics can be integrated to be part of the language, enabling consistency in both data presentation and calculation.
Support for Derived Data

In addition to being able to normalise different sources of data, TimeScape provides an analytics and models SDK that enables a client’s preference regarding statistical analytics, yield curve and pricing calculators to be integrated into the system in a consistent manner. This is illustrated in Figure 6 below.

![Figure 6 – Integrating Proprietary Analytics for Consultant Calculations](image)

Besides the potential for integrating pricing models, TimeScape also allows the integration of spreadsheet-like calculations, minimising the operational risk of spreadsheet usage and enabling the rapid transfer of spreadsheet prototypes to mainstream calculation.

While consistent data is a foundation for greater efficiency in managing data, consistent calculations are equally essential since these produce the numbers that matter to the clients, auditors and regulators.
Summary

This paper has set out a case for a business-focused, distributed approach to the management and integration of data and analytics across the broad range of systems typically found at large financial institutions. Xenomorph’s analytics and data management system, TimeScape, is ideally suited to such an approach and exhibits the following characteristics:

- Consistent Data Access
- Easy Integration Framework
- Support for All Data Types
- Extendable Data Representation
- Fine-Grained Control
- Easy Data Access
- Easy ID Code Translation
- Easy Data Analysis
- Support for Derived Data

For further details of the topics discussed in this paper please look at the case studies and white papers found at www.xenomorph.com and for any questions please contact info@xenomorph.com.
Appendix 1 – TimeScape Business Flow

Sources, Systems and Sites

<table>
<thead>
<tr>
<th>Datafeeds</th>
<th>Files</th>
<th>Databases</th>
<th>Spreadsheets</th>
<th>Analytics</th>
<th>Models</th>
</tr>
</thead>
<tbody>
<tr>
<td>Raw Data of any Type, Complexity, Frequency or Volume</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Customise

<table>
<thead>
<tr>
<th>Database integration and normalisation</th>
<th>Datafeed integration normalisation</th>
<th>Statistical analytics integration</th>
<th>Pricing model integration</th>
<th>Spreadsheet calculation integration</th>
<th>Financial Object Integration</th>
<th>Data Model/ Instrument Design</th>
</tr>
</thead>
</table>

Control

<table>
<thead>
<tr>
<th>User Access Permissioning</th>
<th>Fine-Grain Access Permissioning</th>
<th>Database Template Management</th>
<th>4-Eyes Approval</th>
<th>Acquisition Configuration</th>
<th>Audit Trail</th>
</tr>
</thead>
</table>

Acquire

<table>
<thead>
<tr>
<th>Streaming/Snap Real-Time Datafeeds</th>
<th>Interactive Market/Static Datafeeds</th>
<th>FTP/File Import Management</th>
<th>XML Import Management</th>
<th>Excel Imports</th>
<th>Customised Importers/APIs</th>
</tr>
</thead>
</table>

Create

<table>
<thead>
<tr>
<th>Instrument Creation and Management</th>
<th>Spread Curve Creation and Management</th>
<th>Indices/ Baskets Creation and Management</th>
<th>Volatility Surface Creation and Management</th>
<th>Derived Data Generation</th>
</tr>
</thead>
</table>

Validate

<table>
<thead>
<tr>
<th>Validation</th>
<th>Cleansing</th>
<th>Exception Management</th>
<th>Data Enrichment</th>
<th>Waterfall/ Preference Data Rules</th>
</tr>
</thead>
</table>

Store

<table>
<thead>
<tr>
<th>Real-time, Static, Historic, Derived</th>
<th>Multi-ID, Multi-Source</th>
<th>High Volume</th>
<th>Simple to Complex Data Types</th>
<th>High Performance Vector Storage</th>
<th>Standards-Based DBMS</th>
</tr>
</thead>
</table>

Analyse

|-------------------------------------------------|----------------|-----------------------|---------------------|--------------------|---------------------|-------------------|

Distribute

<table>
<thead>
<tr>
<th>File Export</th>
<th>Broadcast</th>
<th>API Interfacing</th>
<th>Excel</th>
<th>Event Management</th>
<th>Notification</th>
</tr>
</thead>
</table>

Multiple Destinations

Front Office  Risk  P & L  Back Office

Consistent, High Quality, Auditable, Secure Information
Appendix 2 – TimeScape Logical Architecture

Xenomorph®

TimeScape®

APIs
.NET, COM, C/C++, SOAP, Java, Microsoft Excel

Applications
TimeScape Workbench
Microsoft Excel
Web Client

TASK SERVICES
• Validation
• Pricing
• Bulk Data Capture
• Tick Data Capture
• Data Import/Export
• Custom Processes

SERVICES
• Event
• Broadcast
• Tick Capture
• Import Server

DATABASES AND DATA SOURCES
FAME
Reuters
Bloomberg

SQL Server
Microsoft

XDB

CONNECTIVITY SERVICES

DATA SERVICES

QUERY SERVICES

PRICING SERVICES

ANALYTICS SERVICES

PRICING SERVICES

ANALYTICS SERVICES

DATA SERVICES

CONNECTIVITY SERVICES

QUERY SERVICES

PRICING SERVICES

ANALYTICS SERVICES
About Xenomorph

Xenomorph delivers Analytics and Data Management (ADM) solutions to the financial markets. Our TimeScape technology leverages our clients’ proprietary expertise, enabling them to analyse and manage more data with greater control and transparency.

Our focus is to make our clients more successful by closing the productivity gaps between high performance database technology, data management and end-user data analysis. Through unified and transparent access to data and data analysis, our clients achieve even higher levels of financial innovation, business process efficiency and regulatory compliance.

Trading, research, risk, product control, IT and back-office staff use Xenomorph’s TimeScape data management platform at investment banks, hedge funds and asset management institutions across the world’s main financial centres.