

# Formicary Financial solution

## The answer to today's data exchange challenge

Formicary Financial provides a framework for exchanging data between internal and external applications, in particular between exchanges, information vendors, counterparties and internal trading systems.

The framework enables disparate systems to connect and communicate using a suite of interoperable components consisting of plug-in modules and connectors that can be used alone or in combination to route and transform the data flow.

## Overview

The Formicary Financial solution speeds the flow of data between different systems and decouples dependencies using a hub and spoke messaging architecture, resulting in increased reliability and improved efficiency.

Financial Institutions maintain control over the propagation of data feeds between all applications, allowing simple configuration and reuse.

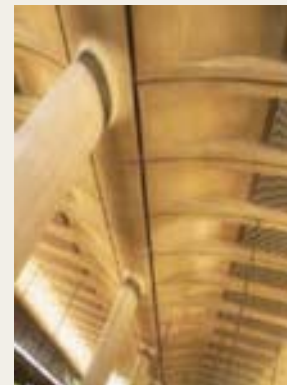
Additionally, all data that flows across the system is automatically recorded, ensuring all trade information is available for future reference, analysis and audit.

Formicary Financial in turn provides further support to traders and other personnel who rely on the most accurate and up to date information available from multiple data sources to maximise their productivity as all cumbersome activities are now reduced to a few clicks of the mouse.

## Benefits

Developed on a component-based architecture, the solution allows applications to be added or modified as required without having to re-configure all other applications. This saves valuable time and resources as changes occur in either trading or regulation procedures.

In addition Formicary Financial offers the facility to enable any corrupted or error data to be corrected and returned to the flow ensuring end users receive only consistent and quality data to maintain effective performance.

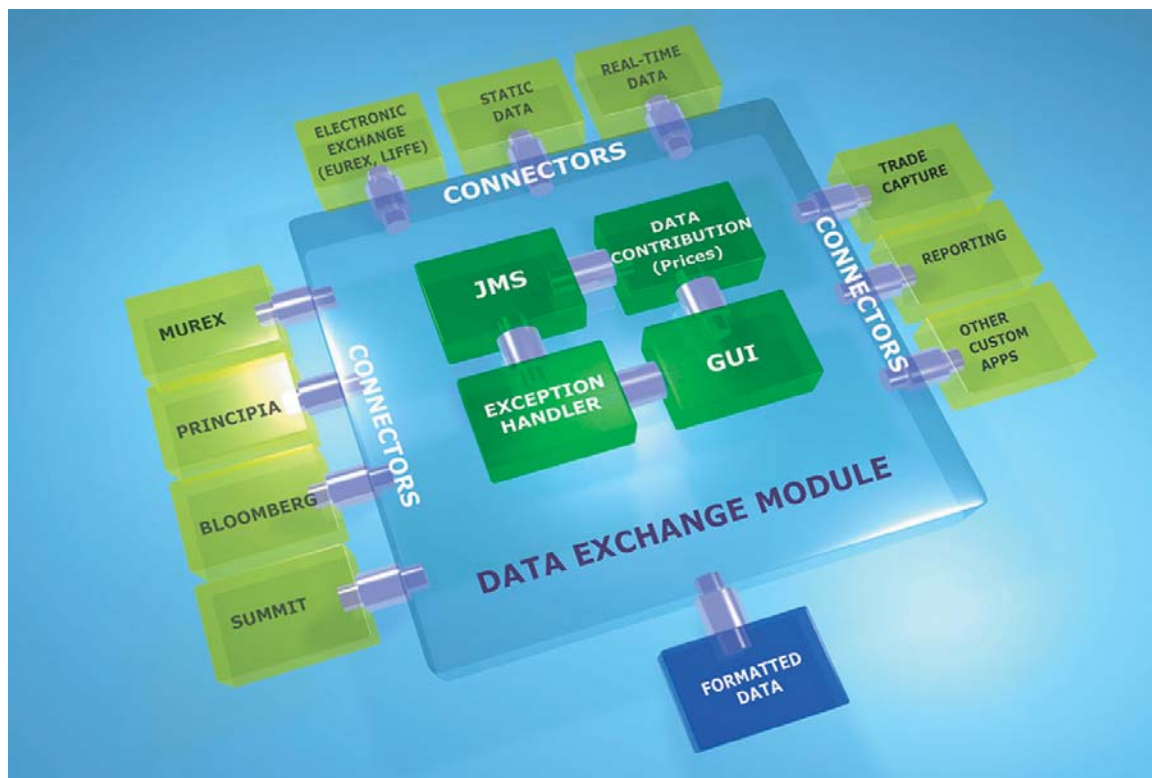


The robust, scalable and highly extensible framework is the ideal vehicle for businesses to achieve the goals of speed to market, effectiveness and return on investment.



# How it works

The simple methodology behind the Formicary Financial framework is that all applications, regardless of language, protocol or system specific communication preferences, are treated as components which can be easily plugged into a central hub. Within the hub, the data can flow freely between all components and is readily available for use by any interested parties.



At the heart of the Formicary Financial solution lays the Data Exchange Module. This is the core component that utilises Java Messaging Service (JMS) interface for effective communications. It consists of a GUI builder, an Exception handler and a Data contribution component for publishing prices. It is the hub that 'Connectors' can be plugged into, enabling data/messages to flow via the Data Exchange Module, connecting multiple disparate systems.

Data sources can consist of feeds from exchanges or trading systems such as Bloomberg, Murex, Summit, Principia and EUREX. Each of these systems has its own custom-built connector, which converts raw or application specific data into a common, standardised format such as FpML or XML. This data can be easily accessed through the Data Exchange Module and utilised by all other connected applications.

Error data is automatically routed to the Exception Handler where it can be centrally monitored and corrected using the GUI provided.

Formicary Financial can be scaled to accommodate a large number of concurrent systems and applications, including web trading applications, reporting tools and many other operationally vital applications. The solution also provides two ready-made modules – The Trade Capture Module and The Reporting Module – these deliver further benefits of consolidating information from multiple sources and systems.

Other customised applications or new developments can be quickly and easily deployed within the Formicary Financial framework, taking full advantage of the Data Exchange Module, simply by plugging in additional custom built connectors.

