



An Architectural Overview of the Ipedo XML Intelligence Platform

**Integrating Information from Diverse Sources
to Enable Real-Time Decisions**

April 2004

EXECUTIVE SUMMARY	1
MAKING ENTERPRISE DATA ACCESSIBLE.....	2
OPERATIONAL BUSINESS DECISIONS.....	2
INFORMATION INTEGRATION CHALLENGES.....	2
INTELLIGENT SYSTEM DESIGN CRITERIA	3
OPEN STANDARDS FACILITATE DEVELOPMENT	3
IPEDO XML INTELLIGENCE PLATFORM.....	5
FUNCTIONAL DESCRIPTION	5
ARCHITECTURAL OVERVIEW.....	6
INTEGRATION MANAGER	6
<i>Content Converters</i>	7
<i>XML Views</i>	7
<i>XML Feeds</i>	9
<i>Synchronization Triggers</i>	9
<i>Inbound Pipelines</i>	10
<i>Rules Checking</i>	10
QUERY MANAGER.....	10
<i>XQuery Engine</i>	11
<i>Federation Manager</i>	12
<i>Results Cache</i>	12
<i>Schema Manager</i>	12
<i>XQuery Module</i>	13
XML DATABASE	13
<i>Indexing Engine</i>	13
<i>Virtual Documents</i>	14
<i>Transaction Processing</i>	15
<i>Version Control</i>	15
ASSEMBLY MANAGER.....	16
<i>Outbound Pipelines</i>	16
<i>JSP Tag Library</i>	17
<i>Accelerator Framework</i>	17
ACCELERATORS	17
IMPLEMENTATION.....	19
DEVELOPMENT.....	19
ADMINISTRATION, MAINTENANCE, AND SECURITY	19
IPEDO BACKGROUND	20

Executive Summary

Most enterprises are rich with data that continually increases in volume and complexity. Each department or business unit captures or generates the information it needs and stores it for subsequent analysis. What happens when an executive has to make decisions using data that spans these information silos? And combine historical data with real-time feeds? And integrate information from customers, suppliers, and business partners? Companies expend significant resources combining data from different areas using processes that are often difficult, cumbersome, or require manual steps with high risk of introducing delays or errors.

These information silos originate when different divisions define their own ways of organizing, updating, and extracting data, or when companies merge. The systems that support each environment may be quite different as a result of timing, acquisition, or independent purchasing decisions. Integrating data from a wide variety of sources and analyzing it effectively is critical to making real-time decisions.

Fortunately, several methods exist for integrating data from diverse sources. But, combining data and displaying it in a single location is not enough. Applications need to extract intelligence from the combined data to enable real-time decision making. Ipedo's approach treats all corporate information as a single, virtual data source, and allows business users to perform on-demand federated queries across all their data. Federated queries allow business users to generate the information they need as they need it, without relying on stale information or static report formats. This drives operational efficiency in rapidly changing environments, which characterize most industries.

XML is emerging as a standard for data exchange, systems integration, and interoperability. Companies use it to either replace or complement EAI (enterprise application integration), EII (enterprise information integration), or ETL (extract, transform, load) initiatives. XML enables detailed query and analysis across diverse information sources such as relational databases, data warehouses, data from enterprise applications such as ERP (enterprise resource planning), CRM (customer relationship management), or HR (human resources), document archive systems, and departmental databases. Since XML documents describe their contents, it is the perfect format for combining data from diverse sources into a single, searchable framework. This framework provides the foundation for real-time data integration that can drive operational business intelligence.

This paper is intended for developers, architects, and technical management in enterprises that need to manage financial or other complex, rapidly changing information. The paper will cover the challenges of making enterprise data accessible to people as well as applications, the importance of open standards such as XML for information integration and designing a scalable IT infrastructure, and the role of federated queries in providing real-time business intelligence. It will provide an architectural overview of Ipedo XML Intelligence Platform and describe why Ipedo is the best choice for creating intelligent applications that integrate information from diverse sources to enable real-time decisions.

Making Enterprise Data Accessible

Operational Business Decisions

Every decision maker in an organization has unique needs. One may want to view information for a specific customer, another for a certain region, and someone else across the supply chain. The vantage point may be as broad as an entire company over several years or as narrow as a specific project in the last quarter.

True operational business intelligence results from the ability to deliver timely, relevant, and accurate information to the point of decision, as opposed to using historical data to validate decisions that have already been made. Applications exist that offer powerful tools to analyze historical data, but they lack capabilities to support real-time business decisions. In some cases, users may want to update information based on their new insights and push their changes back to the original sources, which they may not even know. This creates a requirement that the end-application have a tight bi-directional integration with data sources.

Products that allow intelligent information integration, using technologies such as XML, help businesses leverage the full potential of their existing database management systems, CRM and ERP applications, and content management systems. Ipedo's approach is to integrate information into a flexible middle-tier hub, where information can be centrally managed and customized for each user's or application's unique needs.

Information Integration Challenges

Integrating data from different sources becomes increasingly difficult as the number of formats and the volume of data increases. Yet most large enterprises need to use multiple disparate sources of data to make operational business decisions. Their information resides in multiple systems, in multiple formats, refreshing at different rates. This generates a non-trivial information integration challenge.

Having access to data from different sources is certainly necessary, but by no means sufficient. True information fusion requires processing and analysis of data from different sources into a single, seamless platform that contains some annotation about its source, last update, and what it represents.

Innovative companies are moving to XML to address these concerns. Their goal is usually to build in the ability to use data across multiple applications, and to assure ongoing access to their stored information, regardless of what applications or platforms they use.

XML provides more flexibility and utility than alternative approaches. Enterprise application integration (EAI) products allow all systems to talk to one another, but cannot deliver timely, relevant information to end-users. Enterprise information integration (EII) products consolidate information, but business users often want to create new intelligence from the consolidated data, which these products generally do not allow.

Performing federated queries across multiple data sources presents several major barriers. Systems have different ways to access data that originates in different sources. Relational databases, transaction data, and collections of documents each have unique organization schemes and varying degrees of being searchable.

Valuable Information Originates in a Variety of Data Sources

- **Relational Database Management Systems**
 - Oracle
 - DB2
 - SQL Server
 - MySQL
- **Enterprise Applications**
 - Enterprise Resource Planning (ERP)
 - Customer Relationship Management (CRM)
 - Content Management Systems (CMS)
- **Message Queues**
- **Files on servers**
 - Excel worksheets
 - Word documents
 - PDF files
 - XML documents
 - Flat files
- **Web Services**

Each data source may contain information from different time periods. Aligning the time periods and synchronizing the information presents integration challenges. Another issue regarding integrated data is determining how long to keep information that you want to query. A flexible framework will provide options for data persistence so you can determine how long to cache information, queries, and results for subsequent analysis. It will also keep the data in a common format such as XML so other approved applications or users can access the same consolidated data source to perform their own analyses.

Intelligent System Design Criteria

When developing the architecture of a real-time system that delivers actionable information from disparate data sources, system architects must build in flexibility and scalability in addition to usual design parameters such as security, reliability, stability, and easy maintenance.

XML is peerless as an information model for data and application integration. Its flexibility as a data modeling language makes it an ideal choice for describing message formats and data sources. Its self-describing nature makes it an ideal format for content as well as data. It allows people or other systems to enrich the information by adding structure and additional metadata. Using XML, the walls between data and content come down, creating a seamless information model across any number of sources.

XML offers universal access to information. It can understand relational data, structured and unstructured documents, Web pages, Web Services, native XML, as well as data streams such as message queues. XML is especially well suited for business documents, which often contain arbitrarily complex hierarchical structures or schemas.

Design Checklist

Include these functions when designing your real-time operational business decision-making system.

- Universal access to information
- Bidirectional read/write ability to standard databases
- Perform federated queries across all data sources
- Database capabilities for caching data (operational data store/persistent cache for queries, results, or data)
- Support for XML schemas
- Convert between different schemas
- Convert non-XML data into XML
- Support for XML standards: XQuery, XPath, XSL
- Support for XML industry-specific schemas: XBRL, FpML, FIXML, MISMO
- Robust and intuitive development tools to speed development time and release your solution more quickly

Businesses are migrating towards XML because it supports several key needs for information storage, organization, search, and retrieval. XML facilitates integration and analysis across multiple data structure sources. It has provisions for original source retention, versioning, or archiving. XML tools exist to handle the inevitable changes to document schema that occur over time.

Open Standards Facilitate Development

Using open standards provides several advantages over proprietary architectures and data formats, especially when it comes to data integration, business process integration, and system interoperability.

In general, an open data standard offers improved access to information that both humans and applications can read. Using XML adds context to the content, so any piece of information has a description that annotates it. XML can describe information from a variety of sources, such as relational databases, structured, semi-structured, and unstructured documents, and object repositories. Once in XML, any application or Web Service with the right permissions can read, understand, and process the data.

Open standards also facilitate interoperability between different platforms and applications. Rather than rely on vendors to reveal their proprietary data formats, open systems freely share information using a common public standard. Web Services via SOAP are becoming one key method for applications to communicate and exchange information with one another. If all applications understand the data, it is also easier to connect business processes together, distribute processing, and optimize overall system performance.

With the inevitable change that accompanies IT, open standards are easier to evolve. When the relevant industry-supported standards body releases a new version of the standard, organizations can migrate their data at their own pace or rely on software vendors to perform the format change as part of an application development project.

Key W3C XML standards include: XML Schema, XPath, XQuery, XSL, and SOAP. XML Schema define the structure, content and semantics of XML documents. XPath is an expression language that allows applications to locate and process values in XML data structures. XQuery is a query language that uses the structure of XML to extract information intelligently from XML documents. XSL (Extensible Stylesheet Language) is a language for transforming, formatting, and displaying XML documents. SOAP (Simple Object Access Protocol) is a communications protocol used for defining web services.

Ipedo XIP, through its XML foundation, provides companies the ability to understand and manipulate their content and data in ways not previously possible. All XML data becomes rich, granular, and searchable at the XML node level. Applications must understand the nuance of XML's structure, granularity and embedded meta-data to perform optimally. These granular access capabilities can also assure that only the most relevant information will be assembled and delivered to applications, thus avoiding the "information overload" associated with other solutions.

Ipedo XML Intelligence Platform

Functional Description

The Ipedo XML Intelligence Platform (XIP) offers a new approach to data management and content delivery. It gives organizations centralized control over their information and the ability to extract intelligence from that information. XIP enables users to instantly access and view information from multiple sources in real-time.

Ipedo XIP is a powerful software infrastructure platform that automates aggregating and presenting information from disparate sources. These sources may be relational databases, legacy applications that lack the latest data integration capabilities such as XML, or collections of structured or unstructured documents. Ipedo XIP combines this information, transforms structure and/or content according to user-defined rules, dynamically assembles it, and allows applications to instantly provide users with the information that they want, the way they want it. The core functions include:

- Integrating disparate information in many formats from database records, packaged or legacy applications, word processing documents, presentations and Web pages
- Managing and enriching information centrally, consolidating information in a single view and providing details about how disparate information is related
- Delivering customized, intelligent views of information to people and applications by using XML to identify and correlate information
- Leveraging existing infrastructure by using data from existing database management systems, CRM and ERP applications, and content management systems

XIP At-A-Glance

A unique combination of integration and management

- Simplifies real-time access to information
- Reduces implementation complexity
- Validates and screens incoming data from different sources using business rules
- Allows custom post-processing on information
- Dramatically reduces the cost of change and evolution of information requirements

A powerful XML foundation

- Addresses all content and data types
- Allows flexibility for change
- Uses a template-driven approach
- Includes intuitive management tools

A robust platform that complements existing reporting systems

- Integrates data from different sources without hardcoded connections
- Extracts only relevant information, when needed
- Bridges across many content and data sources

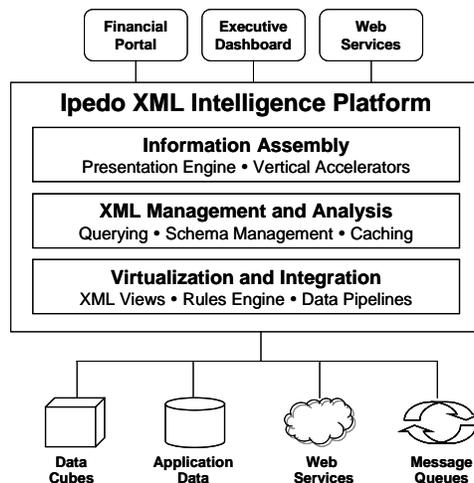


Figure 1 - Ipedo XIP performs a unique combination of integration and management functions, focusing on assembling and delivering relevant and timely information to people or applications.

Architectural Overview

Ipedo XML Intelligence Platform acts as an intelligent information assembly server that connects existing information systems with new applications to deliver powerful, real-time data analysis capabilities seamlessly across all the data sources. Ipedo XIP provides a system for capturing knowledge and intelligence and automating the assembly and flow of information to applications and services exposed to end-users. Developers can deploy this functionality using either the intuitive GUIs that Ipedo provides or with industry-standard J2EE development tools.

Ipedo XML Intelligence Platform consists of four tightly integrated modules plus industry-specific Accelerators. The Integration Manager, Query Manager, XML Database, and Assembly Manager work together to deliver a complete data integration and analysis solution. The Integration Manager federates information from various sources into XML. The Query Manager performs a variety of powerful search capabilities across information from multiple sources. The XML database can be used for caching results, creating an operational data store, or for storing information in XML format. The Assembly Manager speeds delivery of information in recipient-customized format and provides reporting and analysis functionality for all the information available through the Ipedo XIP platform.

In addition to these modules, Ipedo offers Accelerators that speed development of vertical applications. Accelerators include a set of customizable components that add business rules, queries, report templates, and schema libraries. The Accelerators work seamlessly with each module to deliver a rapid solution to a specific market.

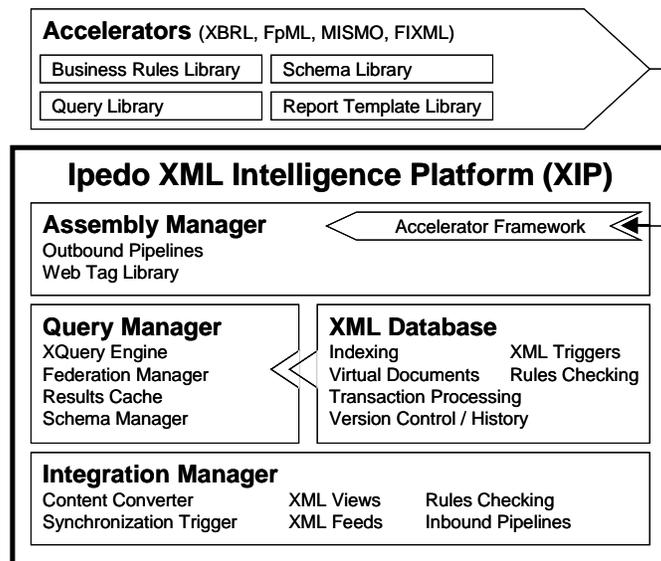


Figure 2 - Ipedo XML Intelligence Platform

Integration Manager

The Integration Manager connects the Ipedo XML Intelligence Platform to remote data sources. It can perform this function using either a “pull” or “push” mechanism (via **XML Views** or **XML Feeds**, respectively). Ipedo XIP integrates with existing relational databases such as Oracle, DB2, SQL Server, and MySQL, packaged applications such as SAP and Siebel, and document management systems including Documentum and Interwoven, to bring both data and content together, seamlessly, into a common information model.

The Integration Manager consolidates and standardizes data, which permits users to perform federated queries and generate

Ipedo Integration Manger

Merges information of all kinds transparently and intelligently

- Integrates information from multiple sources in a single view
- Ensures consistency across disparate systems and documents by expressing all information as XML
- Allows users access to up-to-the-minute information by providing on-demand views of different information sources

common views across all their information assets. This broad information gathering ability allows users to bridge gaps between different databases and create new insights.

Content Converters

Content Converters translate proprietary content formats, including Microsoft Word, Microsoft Excel, Adobe PDF, and Corel WordPerfect, into XML. Once in XML, Ipedo XIP allows applications to manipulate these documents just as granularly as any other XML information available to the platform. For example, Ipedo XIP's powerful XML Query engine can seamlessly combine content from Word and PDF documents with Oracle and other data via XML Views. This allows applications to access information trapped in proprietary, difficult-to-manage formats as fluidly and flexibly as data from traditional database systems. All XML information becomes available for integration, querying, or analysis, regardless of its origin.

XML Views

XML Views create virtual, real-time snapshots of information from external XML and non-XML sources. Potential data sources include relational databases, Web Services, any HTTP-accessible data source, and any XML documents stored in both local and remote Ipedo XIP instances. XML Views present a standard, common, fully searchable XML format to simplify integration tasks in applications.

An XML View defines a data source of interest and the connection criteria needed to access that data source. The requesting application does not extract the information it needs from the underlying information source until a query is performed, assuring that XML Views provide true real-time access to information.

For example, a developer can define an XML View over a relational database by specifying connection information for the database (XML Views currently support JDBC-based connections). As an option, they can limit the working set of the XML View by using SQL queries or table listings. For a Web Service, the View definition will contain the accessor methods used to pull data via the SOAP interface. Views utilize intelligent query re-writing and execution to generate the optimal approach for pulling information from the external source.

Developers, without programming, create view definitions through a simple and intuitive graphical user interface. Since XML Views consolidate information from different data sources, running queries using XML Views provides an easy method for performing federated queries. The user is always assured of having the latest information available, as the query does not execute until the information is requested by the application. The following example illustrates the power of XML Views.

Example

Using XML Views to display real-time information from multiple information sources

Scenario:

A trader is deciding whether to complete a marginally profitable transaction with a long-term customer. Fortunately, the bank has a portal that enables trading and risk management personnel to make informed business decisions based on current data. They have a seamless view of real-time information, including positions data (from a relational database), customer profiles (from a CRM system), current market prices and interest rates (from a Web Service), and financial forecasts (from an XML document cached locally in XIP).

Conventional Solution:

With traditional integration approaches, an application needs to access each individual information source, extract the required information, and assemble the "portal view" information object manually. This requires a significant amount of error-prone custom coding.

Innovative Solution:

Ipedo XIP offers a more elegant and intuitive solution to this problem using XML Views. The developer builds the view using a screen similar to Figure 3. XIP automatically creates the XQuery statement to retrieve the information from its original sources.

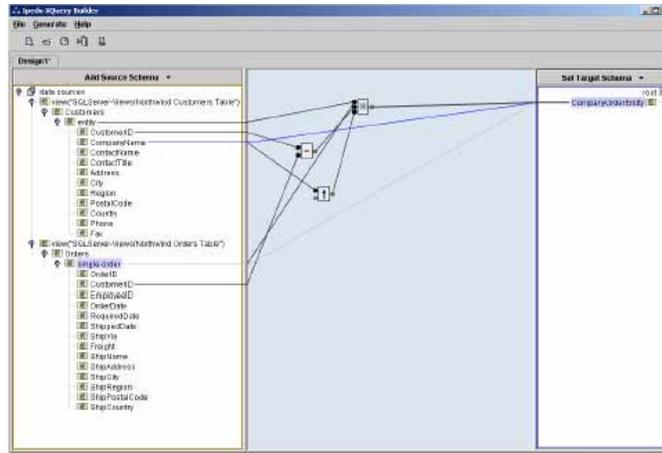


Figure 3 - XML Views provide an intuitive way to query information from diverse sources.

“**PositionsView**” performs a join across the customer, position, pricing, and settlement tables in a relational database.

“**AccountView**” examines a data warehouse, which loads nightly and includes mark-to-market data.

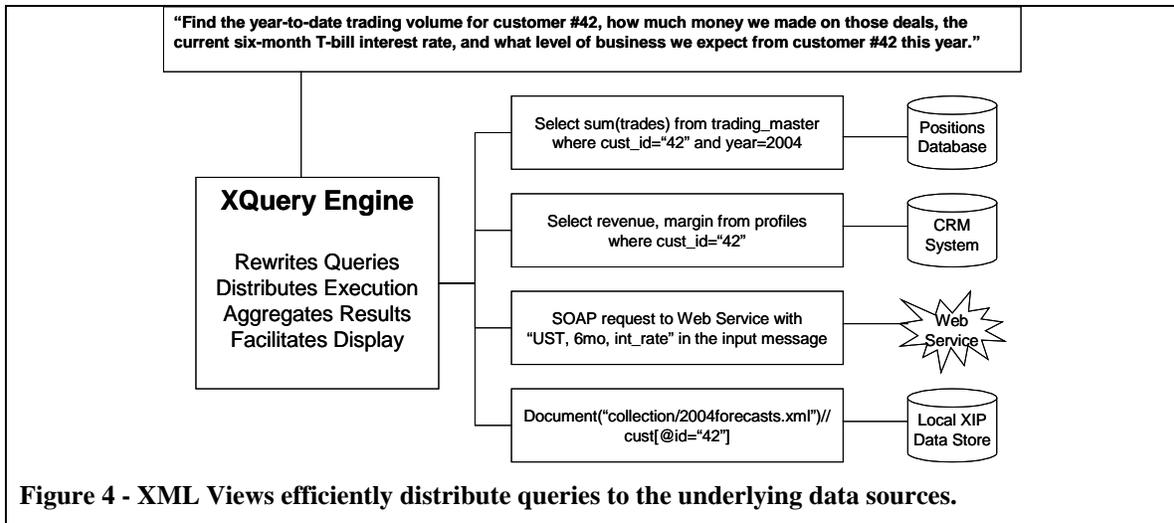
“**MarketView**” uses a Web Service to retrieve current market prices and interest rates.

“**ForecastView**” overlays the preceding Views and combines them with sales forecasts extracted from a standalone XML document stored in the local Ipedo XML data cache.

By combining this information in real-time, a trader can run what-if scenarios to calculate profitability based on different pricing and maturity dates, taking into account trading at other desks, current prices and interest rates, forecasts, and the group’s profitability targets. Since all the data is stored in XML, other traders can access the same data sources and perform different analyses. Executives or global risk managers can drill down from corporate data to individual departments or customers seamlessly.

This solution is fully extensible: to include another information source, just add another view. It is also easy to modify, either through the GUI or by changing the application.

The XQuery engine efficiently distributes queries over Views to the underlying information sources in that source’s native query language. In this example, the query is rewritten as a SQL query over the relational database, a SOAP message request to the Web Service and a local XQuery query over the forecasts documents. Figure 4 illustrates this concept graphically.



XML Feeds

XML Feeds load data into the XML Intelligence Platform from external processes. In contrast to XML Views, which represent a “pull-based” process, XML Feeds represent a “push-based” approach. This lets developers use a data loading model that best fits their application.

As with XML Views, Ipedo XIP can convert data from non-XML sources into XML documents on the fly. In addition, XML Feeds include filters that perform common tasks such as validating data against schemas, storing documents, executing XQueries or transforming data using XSLT stylesheets. In addition, companies can apply business rules or filters to scan incoming data. Users can define their own filters using either built-in tools through a GUI or by writing their own scripts and integrating them into XIP.

Synchronization Triggers

Synchronization Triggers provide a mechanism for synchronizing XML information managed by local Ipedo XIP data stores with a master copy of that information residing in an external trusted source, whether or not that external source is XML. Existing applications on the external source and new XML-based applications running on Ipedo XIP use this integration model when both need to update live information regularly. This allows new applications to take advantage of the flexibility of the XML information model without modifying existing applications or information sources. Triggers watch a specific element in an XML document and activate when that field is modified or deleted.

Synchronization Triggers use an activation mechanism built into the XML Database to synchronize updates to information with external sources. Synchronization Triggers consist of two parts: the inbound trigger, which runs in Ipedo XIP, and the outbound trigger, which runs on the external source system. Ipedo provides pre-built outbound triggers for any relational database that supports triggers, such as Oracle, IBM DB2 and Microsoft SQLServer. The inbound and outbound triggers communicate with each other using an efficient XML protocol developed by Ipedo.

XML messages adhering to the protocol can travel between the triggers using either HTTP or JMS queues. Ipedo recommends JMS for better scalability and robustness. When an update occurs on Ipedo XIP-managed XML data linked to a trigger, it sends an XML message from the Ipedo inbound trigger to the external source’s outbound trigger. The outbound trigger decomposes the message and performs the requested update. When an update occurs on the external source, it fires a trigger to send a message to the Ipedo XIP inbound trigger to perform the necessary update. Any XML document or XML document fragment managed by Ipedo XIP can link using triggers. These capabilities allow Ipedo XIP to run transparently alongside existing information sources of record.

Inbound Pipelines

Inbound Pipelines contain a sequence of steps that operate on a set of data sources. Users can create and save different pipelines for different applications and automatically deploy them, based on the content of the incoming XML documents. By integrating business logic with a set of saved pipelines, they can create a powerful method for screening data and taking different actions such as sending to different destinations or performing different content or style transformations.

Pipelines contain multiple stages, where each stage has a transformation operation with an input source and an output source. Since a conversion step can be one of the process steps, Pipelines can handle any of the data sources that feed into the Ipedo XML Intelligence Platform. Rules Checking can also become part of the Pipeline, further automating the process of consolidating data and verifying its validity as part of an incoming feed.

Pipelines use a stepwise model for processing XML content. They break complicated eXtensible Stylesheet Language (XSL) transformations into manageable pipeline steps. XML Pipelines also allow complex, multi-step processing via XSL and XQuery combinations resulting in a highly flexible approach to integrating, validating, and interrogating data from diverse sources.

Rules Checking

Ipedo XIP provides comprehensive business Rules Checking capabilities to screen or validate messages according to user-defined criteria. Rules Checking examines the content of incoming data in real-time. It performs semantic checks based on the data content, and may also confirm syntax or structure.

XIP's powerful rules checking can examine XML documents at any level of granularity and perform appropriate actions based on the defined rules. Rules Checking looks at the information in specific XML tags and compares the values to pre-defined constants, ranges, or lists. It can also compare data to information in other XML documents or the results of a Web service. Customers can use Rule Checking to authenticate data integrity, verify compliance with government or industry regulations, or confirm internal quality assurance guidelines.

For example, FpML trades can be reviewed as they occur for accuracy, completeness, and regulatory compliance. Approved trades can route to the proper trading desk risk officers, and different trade tracking systems, each in a different format and only containing the relevant content for that destination. Rejected trades can go to different destinations, depending on the corrective action required.

Developers can specify business rules either declaratively using Ipedo XIP's built-in management tools, programmatically in Java, or by using a platform independent mechanism such as Web Services. Ipedo XIP also lets companies incorporate existing business logic through an outside rules engine.

Query Manager

The Query Manager resides at the core of the XML Intelligence Platform. It provides centralized query capabilities across diverse information sources. Query Manager helps enterprises organize, enrich, and mine information collected using the Integration Manager and assemble the most relevant responses to application queries.

Ipedo Query Manager automatically organizes and validates information, assuring orderly and accurate access. Sophisticated indexing and query functions add structure and context to collections of information. This enhances the intelligence of the information and

Ipedo Query Manger

Provides unified management of diverse information sources through a unique fusion of XML standards and database technology

- Centralize information from any source and display it in a single view
- Eliminate information overload by assembling only the most relevant information from multiple sources in response to a query
- Simplify information management to accommodate changing business rules

enables more detailed searches. In addition, an integrated style engine allows users to create customized content views.

Query Manager components include a sophisticated XQuery engine, robust centralized XML schema management, intelligent indexing, and other features to support XML applications.

XQuery Engine

The XQuery Engine forms the foundation of the Query Manager. It consists of an analysis engine, optimizer, and execution engine, all tightly integrated with an associated XML database. The pieces work together to allow companies to mine their information for the most relevant answers.

The XQuery Engine conforms to the latest W3C XQuery 1.0 draft (Nov 2003). It includes extension functions to perform granular updates and full-text searches. This support for granularity in both updates and full-text searches gives users much more control over their information versus traditional natural language processing approaches. The XQuery Engine can also perform query plan caching to greatly improve the performance of often-used queries that contain the same base query logic but may contain different result set constraints.

The query optimizer is the most important and sophisticated component of the query engine. It performs a number of optimizations like predicate reordering (changing the execution pattern of result set constraints to better optimize the returned information set), choosing the most appropriate indexes for the query, and even reordering the query statement itself to create the most direct query plan possible. The XQuery engine can also give developers a human-readable log of the query plan that the optimizer generates so that developers can optimize their queries by hand if needed.

Ipedo includes an XQuery Builder application that provides an intuitive user interface to graphically build queries. It provides views to all the information sources, along with decision criteria to build the search conditions. The GUI shows the query language as it builds it, giving developers the opportunity to make changes if they choose.

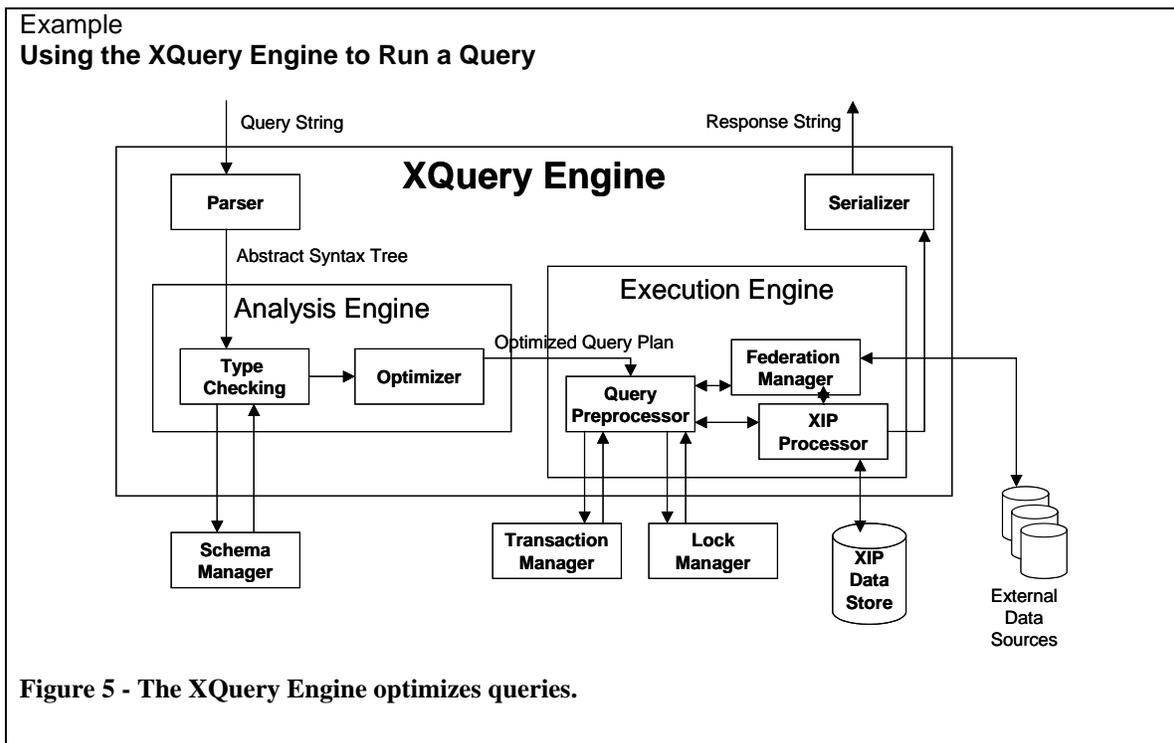


Figure 5 - The XQuery Engine optimizes queries.

Figure 5 and the following process steps illustrate how XQuery Engine executes a query.

1. An application submits a query string to the query engine.
2. The parser generates an abstract syntax tree.
3. The analysis engine performs static type checking to detect any type errors.
4. If the query involves documents that have an XML schema, the analysis engine uses the schema manager to obtain type definitions.
5. The optimizer generates an optimized query plan based on the information generated in steps 1 to 4. It makes each query more efficient by leveraging all available indexes.
6. The execution engine continues processing the optimized query.
7. The transaction manager establishes the transaction boundary.
8. If the query involves an update operation, the lock manager implements the necessary locks to ensure concurrency control.
9. If the XQuery statement involves operations that use XML Views over external information sources, the Federation Manager helps with the query execution.
10. The XIP processor executes queries over documents managed locally by the XML Database and aggregates these results with those from external data sources.
11. The serializer prepares the query response for delivery to the requesting application.

Federation Manager

The Federation Manager aggregates information and performs lookups between disparate data sources using XQuery. It maintains an XML data dictionary of data locations and formats across multiple relational and non-relational sources. This allows it to aggregate content and streamline its delivery in a near real-time environment, giving users access to fresh information on either a time-based or event-driven basis.

Federation Manager provides users a broad range of query functions that allow SQL-like queries across all the data simultaneously. It generates efficient SQL code that pushes the evaluation of search and sorting criteria to each underlying relational database or queryable data source. With a single query, applications can gather information from across the enterprise and deliver it to a single destination for viewing or subsequent analysis. The ability to seamlessly combine and manipulate data from different sources gives decision makers unprecedented power.

Results Cache

The Query Manager uses the Results Cache to keep frequently used information in memory for quick, efficient access. The Results Cache relies on advanced memory management technology to optimize the size and content of its cache and provide the requesting application with its information quickly. Developers can move any collection of XML nodes into or out of memory dynamically to tune overall system performance.

Schema Manager

The Query Manager includes robust capabilities for managing and evolving XML Schemas. Schemas represent a set of business rules that ensure consistency across information models and facilitate inter-company information exchange.

Users can attach a valid XML Schema to any XML document or collection managed by the Query Manager. The database also supports the older Document Type Definition (DTD) form for defining schemas. XIP automatically converts DTDs into XML Schemas when they are loaded into the database. When a schema is attached to a collection, XIP validates all documents in that collection against that schema. Once a schema is attached, the database will perform incremental validation against any update

that occurs against the intended document or collection. This ensures data integrity even as the information changes.

The Schema Manager also supports a technique known as schema evolution. As business rules change and XML Schemas need updating to support those changes, the Schema Manager allows you to easily migrate your documents and collections from the old schema into the new schema. The Schema Manager also tracks the history of schema changes so that you can roll back to previous instances of your schemas, if necessary. These powerful capabilities allow you to evolve your information model over time with minimal effort. This gives Ipedo XIP-based applications flexibility to adapt to application changes in an efficient manner not possible with traditional database technologies.

XQuery Module

XQuery Module follows the W3C XQuery specification to extend the functionality of the server. It allows access to external applications and custom functions written to meet application-specific needs. Developers can create user-defined functions using either XQuery or Java, or plug in external function libraries, such as a digital signature or digital encryption library. Once plugged in, these functions are not distinguishable to the user from other W3C XQuery standard functions.

XML Database

The Ipedo XML Database provides unified management of diverse information sources through a unique fusion of XML standards and database technology. Designed to work with existing systems, such as databases, content repositories and applications, the Ipedo XML Database automatically organizes and validates information, giving users access to information without regard to its original source.

Ipedo designed the XML Database to handle XML data in its native format. While relational and object oriented databases decompose XML into relational tables or objects, native XML databases store XML in the original hierarchical form. The result is more flexible and better performing XML data management and querying. Applications can use as much of the XML Database functionality as they need, from caching queries, results, or data, or using it as a complete ACID-compliant database management system.

Ipedo XML Database

Provides a centralized repository for a collection of fully searchable XML documents

- Perform fully granular searches on any field of any XML document
- Stay current with the latest XQuery standards
- Ensure information consistency

Indexing Engine

The XML Database supports indexing of any XML node or collection of nodes to vastly improve search performance. Ipedo uses a patented technique of in-memory indexing (called Active Edge™) to further improve performance. Active Edge moves often-used indexes and data into main memory for faster processing and improved search results. Developers can specify what indexes they want to accelerate.

Database indexes fully understand the underlying XML Schema. This allows XIP to treat dates as dates, numbers as numbers, and strings as strings, providing the most efficient indexing possible. XIP supports six types of indexes: string, float, int, date, path and full-text. Full-text indexing tools include full or partial document indexing, keyword and phrase searches, Boolean operators, fielded search, parametric search, word stemming and breaking, stop word lists and term weighting, proximity search, and wildcard search. XIP's indexing on strings supports internationalization, for searching documents with non-English character sets. Indexing based on XPath allows developers to create indexes on any XML element, attribute, or path, with or without schema.

Unlike many traditional databases, the XML Database's indexes are adaptive. They automatically update themselves as the underlying information changes. There is never a need to explicitly rebuild an index. This

greatly simplifies administration and maintenance and assures consistent query performance even as the underlying information changes.

Virtual Documents

Virtual Documents provide a simple, link-based way to integrate and aggregate information to form application-specific XML documents. They can deliver a variety of extremely fast access methods for data organization purposes. For example, consider a collection of documents related to a commercial real-estate transaction, such as the loan application, credit reports, bank account information, and appraisals. Virtual Documents can instantly show a snapshot of any information across all these data sources on a single screen. Users can define Virtual Document templates for different purposes to deliver information based on a response to a query.

Virtual Documents are treated by the Query Manager as if they were physical, locally stored documents. The content is materialized on the fly when an operation is performed on the document. This fully separates the physical information storage model from specific application needs, allowing applications to form their own documents that adhere to their needed information model.

A Virtual Document is composed using “virtual document links.” Ipedo developed this URL-based link syntax, making it as easy to use as standard HTTP-based hyperlinks. Virtual document links can point to many different sources of information accessible to the Ipedo XIP including:

- Any XML documents accessible over the Internet via standard protocols like HTTP and FTP
- Any XML document or document fragment in local or remote Ipedo XIP native stores
- Any XML information accessible through Integration Manager’s XML Views

Because they are only composed when requested by an application, Virtual Documents always contain the latest version of information. Changes made to any underlying linked information are transparently reflected in the Virtual Document. This allows for great flexibility in developing application-specific XML documents, even if the structure or information model of those documents must change frequently over time.

```
<WhitePaper>
  <vdoc:Link id="copyright" url="http://www.ipedo.com/pubs/copyright.xml"
  evaluate="STRICT" xmlns:vdoc="http://www.ipedo.com/vdoc" /> 1
  <Title>
    <H1>Ipedo XML Intelligence Platform</H1>
    <H2>Architecture Overview</H2>
  </Title>
  <Section>
    <Header>Executive Summary</Header>
    <Content>
      <Para>Most enterprises are rich with data...
    </Content>
  </Section>
  <Section>
    <Header>Architectural Overview</Header>
    <Content>
      <vdoc:Link id="arch overview"
      url="ipedo://pubs/arch.xml/arch#/overview/"> 2
    </Content>
  </Section>
  ...
  ...
</WhitePaper>
```

1 Link to a boilerplate copyright notice
2 Link to an architectural overview in another document, arch.xml, under the xpath /arch/overview

Figure 6 - Virtual documents consist of data extracted from different sources using embedded links.

The document in Figure 6 contains two links, one to a copyright notice, and another to an architectural overview in a separate XML document. When this document is viewed, queried, transformed or otherwise processed, the links are resolved and the content referenced by the links is placed in-line. Any updates to the linked content are also reflected in the virtual document instantaneously.

Ipedo XIP also supports parameterized Virtual Documents, which essentially creates document templates. Parameters can be used to constrain the resolution of a link to a specific sequence of XML document fragments based on a key, for example. Figure 7 illustrates the use of parameterized Virtual Documents.

```

<CompanyInfo xmlns:vdoc="http://www.ipedo.com/vdoc">
  <BusinessProfile>
    <CompanyDescription>
      <vdoc:Link id="bioprof"
        url="ipedo://companies/biz.xml#/BusinessSummaries/
          Company[@name=$company]/text()"
        Evaluate="STRICT">
        <Param name="company" default="IBM"/>
      </vdoc:Link>
    </CompanyDescription>
    <CompanyFinances>
      <vdoc:Link id="finprof"
        url="ipedo://companies/fs.xml#/FinancialSummaries/
          Company[@name=$company]/text()"
        Evaluate="LAX">
        <Param name="company" default="IBM"/>
      </vdoc:Link>
    </CompanyFinances>
  </BusinessProfile>
</CompanyInfo>

```

1 Link to business profile of company identified by parameter "company"

2 Link to financial profile of company identified by parameter "company"

Figure 7 - Parameterized Virtual Documents allow dynamic criteria to determine the information aggregated when the document is requested.

The links in this document accept a parameter named "company" that constrains the XPath query used to resolve the link. Only XML document fragments that contain "IBM" will be inserted into the document when the link is resolved. The document template in this example can generate the profiles for different companies by passing a different value for the company attribute each time. This illustrates the power and flexibility of Virtual Documents for creating application-specific documents. When the document is requested, dynamic criteria determine the information aggregated into the final product.

Transaction Processing

The XML Database provides a full set of transaction handling functions, including commit, rollback, and log files. Ipedo defines transactions as a group of database operations combined into one logical unit of work that is either wholly committed or rolled back. ACID-compliance (atomic, consistent, isolated, and durable) assures consistent performance for the most demanding transaction processing applications. In addition to basic transaction processing, Ipedo XIP provides enterprise-class capabilities including backup/restore, journaling, and bulk loading.

Version Control

Ipedo XIP contains a complete version control system for XML documents, which you can optionally deploy on your documents. If you choose to implement version control, users or applications can check in and check out an XML document, retrieve previous versions of an XML document by version, date and label, label documents within a collection, and show the revision history of an XML document.

Each document under version control has its own revision history. The revision history tells who made changes to a document and when they made them. You can view the whole revision history or select by user or date.

Version Control allows users to modify source information from queries. They can annotate, modify and refine information through a powerful granular update engine. The built-in versioning system tracks modifications to information, which provides a change history that many regulatory applications require.

Assembly Manager

The Assembly Manager makes information contained in the Ipedo XML Intelligence Platform available through a flexible and easy-to-use framework. The framework uses a set of multiple XQuery and XSL stylesheets to deliver dynamic information to applications, regardless of the intended channel or delivery format. Assembly Manager grants application developers quick access to XML data and content in their Web applications and portals, offering customized views of XML-based information. It reduces application development costs by utilizing standards-based technology to access and manipulate content sources.

Ipedo Assembly Manager

Provides a flexible and easy-to-use framework that uses XQuery and XSL stylesheets to deliver dynamic information

- Develop industry-specific applications quickly
- Reduce application development costs by utilizing standards-based technology
- Make applications more adaptable by separating content from presentation

Ipedo Assembly Manager provides a Web application development framework that leverages the advantages of XML technology while simultaneously exploiting existing Web application and business logic infrastructures such as J2EE, .NET and Web Services. Assembly Manager also provides a set of markup tag libraries that makes it easy for page designers who do not know Java or other high-level languages to make use of the “pipeline” model and other Ipedo product capabilities. It allows them to update applications faster by fully separating content from presentation to allow the application to evolve independently from changing content sources.

The remainder of this section will discuss Assembly Manager’s components and how they facilitate Web application development and encourage content reuse.

Outbound Pipelines

A Pipeline allows for assembly line-like processing of XML content. Pipelines simplify complex XML processing by allowing users to segment processing into manageable reusable steps. Developers give this sequence of steps a global unique name and stored it as a Pipeline definition. They can then reference the Pipeline easily by name. The Assembly Manager hides the more complex processing that occurs during Pipeline execution.

A Pipeline contains three types of components: an XML source, a series of transforms, and a destination. A source component is any producer of XML, which includes whole documents stored in the XML Database, XQuery queries (including queries over Views and Virtual Documents) and other Pipelines that output XML. In a Pipeline, the source XML runs through a series of reusable transform components, each of which performs some action based on the information it receives. The results of the transformations become the XML output and go to the destination. Information destinations include portals, Web services, desktop applications such as Excel or Crystal Report, and handheld devices.

Transform components rely on some combination of filtering, rules checking, and style transformations. Assembly Manager includes a GUI to simplify building complex transformations. Users can easily link components together and implement business logic that screens based on XML content. They can also parameterize Pipelines so that the components use the parameter values to adjust their behavior. Parameterized Pipelines allow for more dynamic assembly of data based upon application-specific information. Developers can also use custom Java filters rather than the GUI, if they prefer.

The Transformation Engine leverages the XSLT standard to easily manipulate XML into other forms including HTML, WML, PDF, plain text, and other dialects of XML. XSLT stylesheets contain rules organized into templates that define how XML should be processed and transformed into the intended destination data model. These rules contain XML queries and in-line processing instructions such as rules or filters that walk through an XML document or XML query result in order, transforming it into another format. The Transformation Engine leverages database-managed granular indexes to greatly improve the performance of the XML queries contained in the templates. The Engine also caches often-used templates into memory to further improve the performance of often-used transformation instructions. The Transformation Engine allows the Pipeline function to provide true multi-channel assembly and delivery of information for applications.

JSP Tag Library

Assembly Manager's tags provide a standard tag library that developers can use for quickly building Java Server Pages (JSP) and Active Server Pages (ASP) applications. These tags make inclusion of XML content into existing Web pages as simple as adding tags. This can greatly shorten the development cycle of Ipedo XIP-based applications and allow developers with even nominal skill sets to write and deploy applications quickly.

These tags encapsulate Ipedo XIP functionality and make it very easy and quick to develop and deploy JSP-based applications. Simple tags are provided to query, update, transform, and execute Pipelines within Web pages and allow developers to easily navigate and publish XML query result sets into their pages. These tags are also supported by popular JSP authoring tools including Macromedia Dreamweaver and Adobe GoLive, allowing non-programmers to make use of Ipedo XIP functionality in their applications.

Accelerator Framework

The Accelerator Framework enables the industry-specific Accelerators to connect with the appropriate components in the Ipedo XIP platform. Each Accelerator includes a set of libraries that contain business rules, schemas, queries, and report templates for a particular industry. The Accelerator Framework integrates these libraries with their more general counterparts in the different XIP modules. Developers can then customize the individual components within each library to meet their individual company's needs.

The Accelerator Framework provides a methodology for building XML information feeds specific to applications or industries. It greatly simplifies the process of integrating, managing and applying business rules on XML information for supporting specific industry standards such as XBRL, FpML, FIXML, or MISMO. The Accelerator Framework is also extensible for other XML standards, so you can integrate your own schema into the XIP Platform.

Accelerators

Ipedo's unique XML-based Accelerators allow organizations to quickly build upon their existing IT systems and implement XML initiatives for a specific industry. An Accelerator contains a set of customizable template libraries for a certain application. Each Accelerator includes four key modules: Business Rules Library, Schema Library, Query Library, and Report Template Library. The Accelerator connects to the Ipedo XIP through the Accelerator Framework in the Assembly Manager.

The Business Rules Library allows a developer to easily apply customizable business rules on XML documents in Ipedo XIP for the purpose of regulatory compliance or business logic validation. These business rules are expressed in a flexible format using XQuery, XSL, or custom Java code. Business rules in existing applications can be leveraged through a

Ipedo Accelerators

Allow organizations to implement XML initiatives for a specific industry or application quickly and easily

- Streamline data exchange with trading partners
- Cut cost of complying with regulatory agencies
- Reduce close time for each reporting cycle
- Improve data integrity and consistency
- Automate manual business processes

number of interfaces such as Web services, Enterprise-JavaBeans (particularly for J2EE applications), JDBC View (for accessing stored procedures in relational systems) or Java API. A developer can pick and choose the rules to be applied at various stages in the lifecycle of an XML document.

The Schema Library contains a set of pre-defined industry-specific schemas. Ipedo tracks the current standards so companies can always follow the latest approved schema for their industry. Ipedo XIP automatically manages the schemas to allow validation and other out-of-the-box XML processing, such as searching XML documents using a query-by-example paradigm. If a standards body updates an industry schema, XIP can migrate documents to the new schema, and keep a revisions history, if desired.

The Query Library has a set of standard queries relevant to that specific application. Developers can customize these queries using XIP's standard graphical tools to add extra functionality their company requires. Using pre-built queries from the Query Library greatly reduces the time to build applications by encapsulating important business logic into the XQuery language. You can display the information in these queries using the report templates.

The Report Template Library comes with a set of customizable Web-based reports and other common reporting tools, such as Excel and Crystal Reports. The Connector library also contains a set of pre-defined Web services (using SOAP or HTTP-URL) that connect to existing applications and expose results of Ipedo XIP's XML processing to other IT systems. In addition, the Accelerator package provides sample Web-based reports that you can use to create portals using fresh information from real-time queries.

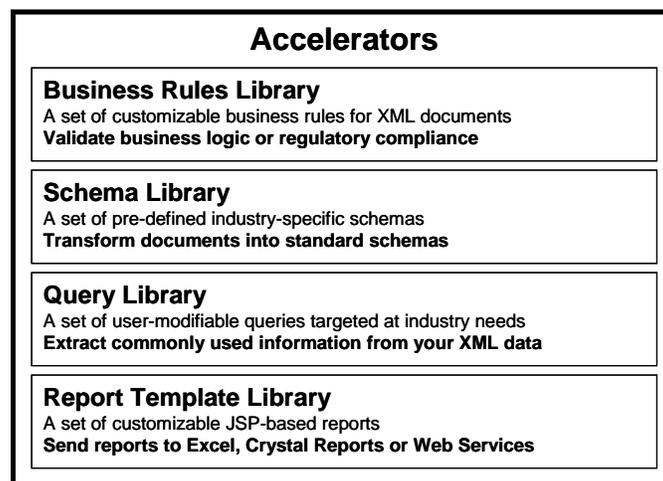


Figure 8 - Accelerators speed application development for specialized vertical markets.

Ipedo offers Accelerators for the following industries.

- Financial Reporting (XBRL - eXtensible Business Reporting Language)
- OTC Derivatives (FpML - Financial Products Markup Language)
- Securities Transactions (FIXML - Financial Information eXchange) (available 2004 Q4)
- Mortgage Banking (MISMO - Mortgage Industry Standards Maintenance Organization) (available 2004 Q4)

Implementation

Development

The Ipedo XML Intelligence Platform frees architects and developers from the burdens of XML processing and lets them concentrate on solving real business problems associated with information integration, enrichment, assembly, and delivery. Ipedo XIP enhances productivity by simplifying development and customization of solutions.

Ipedo XIP provides developers J2EE, .NET, and Web Services (SOAP) application programming interfaces (APIs) for all of its capabilities. It also has WebDAV support and XQuery extensions for update and database resource management. In addition, XIP includes a graphical administration console that manages all aspects of the platform, including linking data sources, building queries, and presenting results.

Ipedo supports deployment with the latest application servers and development tools including BEA WebLogic Server and Workshop, IBM WebSphere, and other J2EE-compliant platforms. Ipedo XIP ships with the Tomcat server, so you can run it immediately. Ipedo XIP runs on Linux, Solaris, AIX, HP-UX, and Windows. Enterprises can deploy XIP's unique functionality using either client-server or embedded deployment models. The combination of API access and GUI-based tools speeds and simplifies application development.

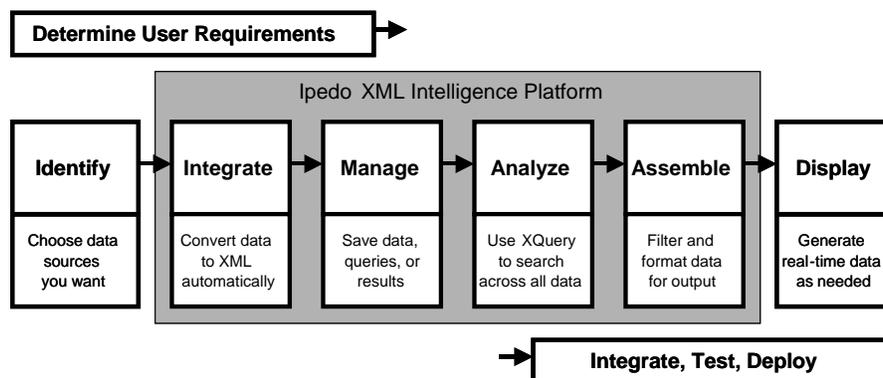


Figure 9 - Ipedo XIP simplifies capturing data and distributed information, creating knowledge and automating the assembly and flow of intelligence to applications and services exposed to end-users.

Administration, Maintenance, and Security

Ipedo XIP simplifies administration tasks with a secure, comprehensive, and easy-to-use graphical administration console that manages all aspects of the system. You can configure Synchronizer Triggers, create Views, manage XML document collections, manage access control, and configure server settings from this central console. This console also provides testing tools for XPath, XQuery, and XSLT transforms that allow you to test your XML logic operations outside of your application code.

For those platforms that do not have a graphical windowing environment or need unassisted administration capabilities, Ipedo

Administrator Friendly

- All administration functions available through GUI or scriptable command-line tool
- Remote monitoring through standard SNMP-based monitoring tools
- Easy installation and configuration
- Import and export – to replicate Ipedo set up and resources (e.g. users, ACL, views, collections, etc.)
- Replication – hardware-based and native support
- Security – full JAAS compliant ACL model with support for external LDAP or any Access Control Manager
- Supports single sign-on to leverage existing authentication methods
- Fully configurable for backup, clustering and failover, system monitoring

XIP provides a text-based, scriptable administration console that provides all of the same capabilities of the graphical console. Both consoles run on any operating system supported by the Ipedo XML Intelligence Platform.

Ipedo XIP allows system administrators to monitor their systems through standard enterprise management tools such as HP OpenView, Computer Associates Unicenter, and IBM Tivoli. XIP supports any remote monitoring tool that follows the JMX or SNMP protocols. This makes it easy to deploy XIP into your existing IT infrastructure.

Journaling, Backup, and Recovery

Ipedo XIP provides a robust journaling mechanism that tracks every transaction performed in the system and allows for full recovery in the event of system failure. The XML Database also supports on-line backup, allowing backups of managed information to be performed safely even while the system is running. Incremental backup keeps your data safe while reducing system load by only backing up data that has changed since the last backup. Full backup is also available. These capabilities ensure full recovery in the event of any major system failure or error and are fully configurable using the administration consoles.

High Availability Deployments

Ipedo XIP supports popular clustering mechanisms including Microsoft Windows, Sun, and IBM Clusters for deploying the platform in any mission-critical environment. This ensures that the platform will be constantly available in the most demanding environments. Apart from fail-over mechanisms, the platform also allows administrators to deploy multiple read-only replicas of XML data to provide options for high scalability.

Security

The XIP platform supports resource-based access control designed around the standard Java Security model. This resource-based model allows system administrators to define access control lists for any Ipedo XIP-managed resource including XML documents, collections, views, adapters, pipelines, and profiles. They can organize users into groups to create more global security roles. This allows Ipedo XML Intelligence Platform administrators to restrict access to information on a more global basis in addition to defining security on an application-by-application basis. Ipedo XIP also supports single sign-on to simplify access to your existing systems.

Ipedo Background

Ipedo is a pioneer in using XML to solve challenging data integration problems and deliver real-time business intelligence. Ipedo was the first company to commercialize an XQuery compliant product, an XML Federated query product, and a Java-based XML database engine. They hold several patents involving innovative uses for XML technology. Ipedo's products are used by leading financial, government, high technology, manufacturing, and pharmaceutical organizations.

For more information about integrating information from diverse sources to enable real-time decisions, please contact Ipedo.

Ipedo, Inc.
1001 Marshall Street,
Redwood City, CA 94063
Tel: 650.306.4000
Fax: 650.306.4001
<http://www.ipedo.com>

© 2004 Ipedo, Inc. All Rights Reserved. Ipedo, Ipedo XML Intelligence Platform, Ipedo Integration Manager, Ipedo Query Manager, Ipedo XML Database, Ipedo Assembly Manager and Active Edge are trademarks of Ipedo, Inc. All other products and services mentioned are trademarks of their respective companies. 4/04