COMPOSITE
— SOFTWARE —

Composite Information Server

> *"At Pfizer, we have all the data integration tools that you can find on the market. But when senior execs come to me daily with key project/resource questions whose answers will determine the courses of action we'll take in running our business, my team uses the rapid deployment methods that are built within Composite. This reduces each project from 4-6 weeks to 2-3 days."*
>
> Director of Business Operations, Dr. Michael Linhares

Enterprises and government agencies constantly strive to find new and better ways to increase productivity, grow revenue, cut costs, and reduce risk. In today's information-centric environment, effective data integration is a critical success factor in accomplishing these timeless business goals.

Composite Software's data virtualization approach to data integration enables you to more efficiently and effectively leverage valuable data that is scattered throughout your organization. Composite integrates data from multiple, disparate sources - anywhere across the extended enterprise - in a unified, logically virtualized manner for consumption by nearly any front-end business solution, including portals, reports, applications, search, and more.

The award-winning Composite Information Server enables IT to:

- Respond faster to business requests by delivering the data new and evolving applications require.
- Reduce costs by leveraging your existing data in new and powerful ways, without added replication expenses; reuse data services across multiple projects to compound your savings.
- Minimize project risks by enabling quick iterations and easy validation of user requirements throughout the development life cycle.

PRODUCT

The Composite Information Server is a data virtualization server that connects to existing data non-invasively, federates disparate data, abstracts and simplifies complex data, and delivers the information as data services. The server includes a graphical development environment that enables rapid design and development of database-centric objects such as relational views and service-oriented objects such as XML Web services. The Composite Information Server also includes a complete set of management capabilities.

ADVANTAGES

The Composite Information Server provides many advantages, including:

- **Ease of Use** – Graphical development and management environments are easy to learn and intuitive to use.

- **Performance** – Dozens of query optimization techniques and algorithms are applied automatically.

- **Flexibility** – Deployable for a wide range of use cases and solutions that require data integration, data access, data abstraction, and more.

PERFORMANCE DIFFERENCE

Based on over 300 man years of query optimization R&D, the Composite Information Server offers the highest performance in the industry. Composite pushes the state-of-the-art in query optimization algorithms and techniques. Our query engine offers both cost-based and rules-based optimizations as well as leverages numerous advanced approaches including multiple join algorithms, predicate push down, parallelism, flexible caching, result streaming and more.

In addition, Composite PerformancePlus Adapters™ provide performance optimization capabilities far beyond those offered by competing products. These adapters intelligently evaluate and leverage underlying data source capabilities to ensure optimal federated query performance.

Composite Information Server and Product Options Architecture

FEATURES

DEVELOPMENT ENVIRONMENT

- **Modeling** – Design views and services using a high-productivity development environment

  - Introspection – Automatically probe physical data sources and select desired resources.   Optionally introspect data sources interactively.
  - Data Discovery - Reveal data relationships across disparate entities using formal keys and fuzzy matching with the Composite Discovery Option.
  - Bottom-up Modeling – Design views by combining data from disparate systems without worrying about underlying source access and format complexity.
  - Custom Logic – Create views by using scripting languages for added convenience and flexibility.
  - Contract First – Define WSDL first, then develop the Java wrapper.
  - Contract Last – Define Java wrapper first, then develop WSDL.
  - Views Dependency Graph – Graphically display what data sources views are dependent on and vice versa.
  - Query Plan – View all the steps and details around the execution of a view and/or procedure.
  - SQL Script – Implement stored procedures with Composite's own scripting language.
  - SQL 99 – Support of the latest SQL standards.
  - Physical Table Creation - Create and drop physical tables within a single data source.
  - Analytic functions – Supports a full set of analytic features (i.e., CORR, COUNT, NTILE, STDDEV, VARIANCE).

- **Transformation** – Transform complex data structures

  - XPath Transformation – Establish arbitrary complex mapping of XML Schema elements into XML output.
  - XML Shaping – Transform data from tabular to hierarchical and vice versa. Access XML data with SQL.
  - XQuery – Create complex XML structures using a graphical XQuery editor.
  - Contract First Development – Build services using pre-existing WSDLs and schemas.

- **Metadata Repository** – Store and manage metadata

  - Complete Repository – Manage resources, such as data sources, views, and procedure throughout their lifecycle.
  - Public Metadata API – Available as a collection of Web services and built-in procedures.
  - Schema Change Notification – Receive notice when data source schemas change.
  - Metadata Exposure – Access metadata of physical data source.
  - Open API – Web services-based for easy access and sharing.

- **Version Control** - Manage development and changes in a controlled manner.

  - Version-based Import and Export - Save and restore resources from directories and files.
  - Resource Locking - Protection against inadvertent modifications and overwrites.
  - Change History - Track changes made by users with annotations.
  - Source Control and Transport – Facility for managing data artifact sources and transport from development environments through to production environments.

- **Governance** – Provide complete visibility, traceability and control.

  - Data Lineage – Traceability from data source to data consumer. API available for third-party systems to access logging data.
  - Logging – Full activity tracking or only selected components of the server.
  - Standards Enforcement – Apply industry data standards in views/services development.
  - Policy-based Security – Authentication, authorization, encryption.

- **Data Quality** – Ensure correct and complete data.

  - Standardization and Conformation – Create views that meet agreed standards.
  - Enrichment and Augmentation – Extend views with data from other sources.
  - Validation – Verify data value and user requirements.
  - Masking – Present data within contextual formatting.

RUN-TIME SERVER

- **Query Engine** – Execute and optimize queries on a single data source or across multiple and disparate data sources.

  - Federation Engine – Join and aggregate data that is vertically and horizontally partitioned.
  - Cost-base Optimizer – Utilize statistics to create an optimal query plan that minimizes unnecessary data flow across the network.
  - Rule-base Optimizer – Allow users to specify exactly how they want to execute a particular query.
  - Hybrid Memory/disk Usage – Balance usage of memory versus disk space for optimal performance.
  - Transformation – Shape data using XQuery, XSLT, Java and SQL functions.
  - SQL Script – Implement stored procedures with Composite's own scripting language.
  - SQL 99 – Support of the latest SQL standards.
  - Alerts – Resource, event and user-defined triggers. Published API to handle custom java alerts.
  - Scheduling – Execute queries based on set times.

- **Performance Optimization Algorithms and Techniques** – Optimize query performance for large and complex data sets.

  - Complete Set of Join Algorithms – Select and employ the most efficient join strategy for a given situation (e.g., Hash Join, Sort-Merge Join, Distributed Semi-join, Data-ship Join, Nested-Loop Join) to ensure the most efficient data processing.
  - Single-Source Join Grouping – Execute data-reducing joins in the data source rather than bringing the data across the network.
  - Predicate Push-Down – Push WHERE-clause predicates all the way down into the underlying data source to reduce data in the source.
  - Serialize or Parallelize Join Operators – Determine the proper join order and join algorithms based on estimated cardinality and join results derived from data distribution histograms.
  - Projection Pruning – Eliminate all unnecessary columns from fetch nodes in a query tree.
  - Constraint Propagation – Distribute filters to multiple branches of the query plan, allowing data reduction by a single filter to potentially occur in multiple data sources.
  - Scan Multiplexing – Re-use datasets that appear in multiple places in a single query plan.
  - Empty Scan Detection – Detect logical conditions that would produce empty data sets, and then eliminate those parts of the query plan prior to execution.
  - Redundant Operator Cropping – Eliminate redundant or extraneous operators within a complex multi-operator query.
  - Blocking Operator Pre-fetching – Proactively execute parts of the query plan that must complete before other parts of the query plan can continue, thereby increasing overall responsiveness of the query.
  - Result Streaming – Stream data to consuming applications as results are obtained and processed from the underlying sources.

- **Caching** – Materialize data into a designated storage location.

  - Event-driven Refresh – Update cache based on defined business rules.
  - Scheduled Refresh – Update cache based on set times.
  - Incremental Refresh – Update partial cache based on triggered changes.
  - Manual Refresh – Update cache on demand as needed.
  - Native Data Source Load – Use target repository native load functions to load and refresh the cache.
  - Parallel Load - Use multiple threads to load the cache in parallel.

RUN-TIME SERVER (CONT.)

- **Data Access** – Connect and expose data from diverse sources.
  - Connection Pool Sharing – Share access to data source to avoid bottleneck.
  - Databases – Connect to standard databases using ODBC/JDBC.
  - Web services – Consume SOAP over HTTP and JMS services. Supports XML over HTTP. A message pipeline allows interjection of custom logic during the request/response of the Web service.
  - Packaged Applications – Connects to SAP, Oracle E-Business Suite, Salesforce.com and other applications via their approved APIs using Composite PerformancePlus Application Adapters.
  - Multi-dimensional Data Sources – Access multidimensional data sources such as SAP BW and Oracle Essbase using Composite PerformancePlus Data Adapters.
  - Mainframes – Access CICS, VSAM, IMS, etc.
  - Big Data – Access Hadoop through Hive and MPP-based analytic appliances such as IBM Netezza and HP Vertica.
  - Native XML Support – Supports XML internally for fast parsing and joins.
  - Java API – Access non-relational sources using custom procedures.
  - Adapter SDK – Framework and tools for developing custom connectivity to data sources.
- **Data Delivery** – Deliver data to consuming applications.
  - Database Objects – Publish data models in the form of views and procedures for consumption via ODBC/JDBC/ADO.NET.
  - Web services – Publish data services in the form of WSDLs for consumption via SOAP or SOAP over JMS. A message pipeline allows interjection of custom logic during the request/response of the Web service.
  - REST – Publish data services in the REST format. Supports REST CRUD functionality.
  - OData – Publish data services in the OData format.

- **Security** – Support multiple forms of security to maximize data protection.
  - Single Sign-on – Sign on once, and access all integrated applications and data sources.
  - Row-level Authentication – Allows data access authentication to the row level.
  - SSL over HTTP with support for mutual authentication – Mutual authenticate for published services, Web services data source and Oracle databases. Certificate based authentication and WSS authentication support.
  - Pass-through – Use existing user ID and password and pass-through to Composite for authentication.
  - LDAP – Leverage security profiles from LDAP to authenticate user access to protected data sources.
  - Pluggable Authentication Module (PAM) – Leverage third-party systems for authentication.
  - Access Management – Use Composite as the system of record for security roles and profiles.

MANAGEMENT

- **Multiple-access Management Console** – Manage using Studio or a Web browser.
- **Real-time System Indicators** – Monitor key system metrics and tune for optimal performance.
  - Monitor memory usage of the Composite Information Server.
  - View the query plan for currently running and past requests.
  - Check the status on all underlying data sources & cached resources.
- **SNMP Support** – Allow for monitoring by third-party systems.

MANAGEMENT OPTIONS

- **Clustering** – Flexibly manage multiple clustered servers using Composite Active Cluster Option.
- **Monitoring** – Monitor multi-server cluster environment using Composite Monitor Option.

SPECIFICATIONS

ODBC/JDBC ADAPTERS
- ADO.NET
- iODBC 3.521 for Linux, AIX, HP-UX, Solaris
- JDBC 2.0, 3.0, 4.0
- Windows

PERFORMANCEPLUS CORE ADAPTERS
- Cloudera CDH4
- Cloudera Impala 0.6
- Greenplum 3.3, 4.1
- Hadoop/Hive 0.8.1, 0.9.0
- IBM DB2 LUW 8, 9, 9.5
- Informix 9.4
- MS Access
- MS Excel
- MS SQL Server 2000, 2005, 2008, 2012
- MySQL 4.1, 5.0, 5.1, 5.5
- Netezza NPS 5.0, 6.0
- Oracle 9i, 10g, 11g, 11g Type 2, RAC
- PostgreSQL 9.0, 9.1
- Sybase 12.5, 15, 15.5
- Sybase IQ 15
- Teradata 12, 13, 13.10
- Vertica 5.0

PERFORMANCEPLUS DATA ADAPTERS
- SAP BW 3.5, BI 7.0
- Web Content

PERFORMANCEPLUS APPLICATION ADAPTERS
- SAP R/3 4.7, mySAP 5.0, 6.0
- Siebel 7.7, 7.8, 8.0
- Oracle E-Business Suite 11.5.8, 11.5.10
- Salesforce.com

MAINFRAME ADAPTERS
- IBM DB2 z/OS 8

ENTERPRISE SERVICE BUSES
- Sonic 7.5
- TIBCO Active Matrix

DELIVERY INTERFACES
- ADO.NET
- ODBC 3.521
- Hadoop
- JDBC 2 SE 1.5.0, 1.6.0, 2.0, 3.0
- SOAP 1.1
- SOAP/JMS - TIBCO EMS, Sonic MQ
- REST

WEB SERVICES PROTOCOLS
- .NET 1.1, 2.0, 3.0 (client side)
- OData
- REST/JSON
- SOAP 1.1, 1.2
- WSDL 1.1
- WSI 1.0, 1.1
- XPath 1.0, 2.0
- XQuery 1.0
- XSLT 1.1, 2.0
- XML Schema 1.0, SAX and DOM
- XML (Flat files or over HTTP)

DIRECTORY SERVICES
- Windows Server 2003 Active Directory
- iPlanet 4.1.6 sp1
- Composite Information Server
- Novell eDirectory

OTHER STANDARDS
- SQL 92, 99
- Unicode support
- JDK 1.6, J2EE 1.3, JNDI

CACHE REPOSITORIES
- Greenplum 4.1
- IBM DB2 LUW 9.5
- Microsoft SQL Server 2000, 2005, 2008, 2012
- MySQL 5.0, 5.1, 5.5
- Netezza 5.0, 6.0
- Oracle 9i, 10g, 11g
- Sybase ASE 12.5, 15, 15.5
- Teradata 12, 13, 13.10
- Vertica 5.0

SECURITY
- TEAV
- Base64
- Kerberos
- NTLM
- SSL
- WS-Security

PLATFORMS
- Client for Composite Studio
  - Microsoft Windows 2000, 2003, XP, Vista, 2008, Win 7
- Server
  - IBM AIX 5.3x
  - HP-UX B 11.11x
  - Red Hat Enterprise Linux AS 3 32bit, 4x, 5x, 6x
  - Sun Solaris 9x, 10x
  - SUSE Linux Enterprise Server 9.3 32bit, 10x, 11x
  - Microsoft Windows 2000, 2003, 2008, 2012, Vista Business Edition, XP, Win 7
- JVMs
  - 32-bit, 64-bit