

Progress® DataXtend™ CE for Java

Full Lifecycle Data Services for High Performance Enterprise Applications

Highlights

Progress DataXtend CE for Java provides cost-effective, scalable, and high performance object-oriented access to relational data for custom applications. Whether you are developing with the popular BEA® WebLogic® or IBM® WebSphere® application servers or are exposing persistent data to a pure Java application, DataXtend CE for Java meets data access requirements at each stage of the project lifecycle, ensuring a fast, free flow of data.

Features at a Glance

- Eclipse-based object-relational mapping tools and model-driven, interactive code generation to accelerate development.
- Built-in intelligent cache that understands the object model and schema for high performance.
- Continuous Cache Coordination proactively “pushes” database changes to distributed cache.
- Transparent database optimizations for all major databases for native performance and vendor independence.
- Cache clustering for scalability and high availability.
- Cross-platform support of all major deployment platforms.

Think Data First!

Plan for data access early in a custom development project, or accept the risk that data bottlenecks will plague application performance and limit scalability.

Introduction

Technology trends such as grid computing, service oriented architectures (SOAs), and data integration drastically increase the demands on data access infrastructure. Progress® DataXtend™ CE for Java provides cost-effective, scalable, and high performance object-oriented access to relational data for custom applications. With dozens of large scale deployments at global Fortune 500 companies, Progress Real Time Division products are proven in the most demanding enterprise systems. Efficient data access is a significant concern throughout the project lifecycle:

- **Development Phase:** An R.B. Webber study concluded that coding and configuring object relational (O-R) data access typically accounts for 30 to 40% of total project effort.
- **Tuning Phase:** Performance and data integrity issues often do not surface until testing. A recent survey of J2EE users found that 57% of performance and availability problems came from inefficiencies in the data access layer.
- **Deployment Phase:** The same survey noted that during initial deployment, only 42% of applications perform as planned. According to a 2004 Forrester Research survey, 66% of performance problems are first identified by irate users calling the IT help desk.

Whether you are developing with the popular BEA® WebLogic® or IBM® WebSphere® application servers or are exposing persistent data to a pure Java application, DataXtend CE for Java meets data access requirements at each stage of the project lifecycle, ensuring a fast, free flow of data. DataXtend CE’s tightly integrated features shorten development schedules, simplify performance tuning, and ensure enterprise wide deployment scalability and high availability. When you use DataXtend CE’s development tools, you build runtime performance and scalability into your application. You gain the flexibility to seamlessly access sophisticated tuning and deployment features without rearchitecting or additional coding. Beyond the initial cost savings, DataXtend CE’s data services platform reduces the risk of project failure due to schedule overruns, poor performance, or application failure.

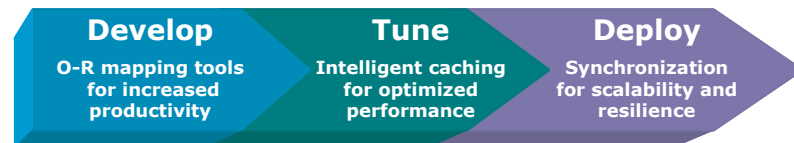


Figure 1. DataXtend CE optimizes data access across the project lifecycle.

The DataXtend CE Advantage

Most data access products address the requirements of just one phase of the project lifecycle, such as O/R mapping for development, caching for tuning, or cache clustering for deployment. Because these standalone products each address only a portion of the data access problem, they require additional integration with other products, an effort that is intrusive and that cannot deliver the efficiency, data integrity, and operational resilience of a comprehensive data access platform. In contrast,

DataXtend CE for Java offers a flexible, full-lifecycle, and cross-platform approach that meets requirements from development through deployment:

- **Development features:** Speed design and implementation with tools for rich O/R mapping and model-driven code generation. Database schema browsing and import automates specification of the application data model.
- **Tuning features:** Ensure high performance with intelligent caching for the generated mapping classes to dramatically reduce the load on centralized data sources. Configurable cache management policies optimize performance with no additional coding.
- **Deployment features:** Provide cost-effective scalability and resilience for deployment within the data center and beyond. Clustering, cache synchronization, and high availability features keep your application up and running even under peak loads. When user demand increases, these features support incremental infrastructure expansion without code changes.

DataXtend CE's flexible and extensible data access platform is optimized for demanding applications exploiting new architectural patterns, especially those with complex data models, high request rates, or both. This comprehensive data solution eliminates data bottlenecks that limit performance and scalability or even contribute to project failure.

Development Features Increase Productivity and Reduce Risks

During the development phase, meeting functional specifications and delivering the application as quickly as possible take top priority. Developers have many choices for data access – such as JDBC,™ EJB, JDO, or proprietary tools – each with benefits and drawbacks:

- Some tools provide no abstraction of the data layer, forcing developers to understand the database and schema and to expose relational complexity in the application.
- Code-driven approaches provide little automation, requiring significant development efforts.
- EJB deployment descriptors and JDO metadata files require developers to program data access logic in XML, a language not well-suited for this purpose.
- The overhead imposed by entity bean remote interfaces is widely recognized as a limit to performance and scalability.

- Many O/R strategies lack flexibility for mapping complex objects to existing data tables and can force time-consuming workarounds.
- Most O/R mapping tools are language specific and do not provide consistent object model semantics across multiple platforms, requiring duplicated efforts to build and maintain mapping layers across the enterprise.

Each strategy has different capabilities and performance characteristics, making it difficult to identify a winning combination. In contrast, the ideal solution for data access is a single product that relieves developers from writing low-level infrastructure and supports complex object models, while building in performance and scalability.

DataXtend CE meets these key requirements in three ways:

1. By simplifying data model design and O/R mapping with automated tools.
2. By generating a high quality, flexible data access layer that reduces testing and debugging efforts.

3. By integrating the generated code with intelligent caching to ensure performance and data integrity without additional programming.

Progress Real Time Division's patented O/R mapping and caching technology reflects the sophistication and reliability of hundreds of man-years of research and development – delivering battle-tested code for the data access layer. The reduction of development, testing, and maintenance delivers quantifiable cost savings as documented by our customers.

Flexible Design Tools

DataXtend CE O/R mapping tools support a variety of development processes. For developers who prefer integrated development environments (IDEs), the DataXtend CE Plug-Ins for Eclipse add O/R mapping and code generation to this open source IDE. The DataXtend CE product includes a complete Eclipse workbench, for which a variety of value-added plugins that simplify and automate common development tasks are available.

Documented Development Savings

DataXtend CE customers have compared the efforts involved in hand-coding data access with automatic data layer generation:

- In one case, every hand-coded persistent class contained about 1,000 lines of code. Not only did use of DataXtend CE-generated code result in a five times increase in productivity, it also reduced bugs by 60%.
- In another case, defects were reduced by more than 90%. The total savings for an application with several million lines of code was estimated at nearly two million dollars when factoring in reduced development, testing, and maintenance efforts.

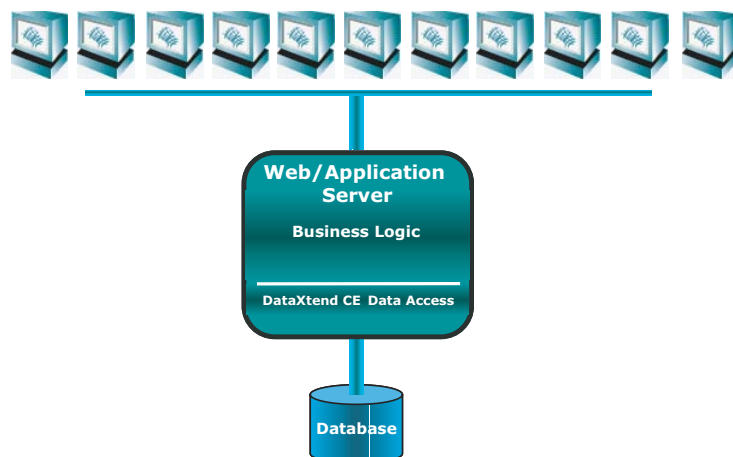


Figure 2. A DataXtend CE data access layer provides O/R mapping, caching, and cache synchronization. A Java application or J2EE application server hosts the data access layer and business logic.

For those who prefer UML modeling, DataXtend CE Add-Ins for Rose provides O/R mapping and code generation for IBM's Rational Rose®. For those who choose not to use Rose or Eclipse, the DataXtend CE Object Builder offers an alternative simple interface. All of these tools define a project that captures O/R mapping and code generation options. You can easily move DataXtend CE projects from one tool to the other.

Many O-R mapping tools map existing classes to new tables, but provide only limited support for existing tables. DataXtend CE tools allow designers to start from either the application data model or from the database schema. The Plug-Ins' Database Explorer reverse engineers existing database schemas and allows you to drag-and-drop elements into a DataXtend CE project, accelerating development of the persistent data access layer.

Benefits of a Data Services Layer

The persistent data access code generated by the DataXtend CE tools abstracts the data source, allowing you to switch the database without changing application code or recompiling. By supporting all major relational databases, use of DataXtend CE prevents database vendor lock-in and simplifies migration. Yet, the underlying data access code takes advantage of each database's unique performance features.

DataXtend CE tools can generate Java, C#, and C++ code, providing a consistent, enterprise-wide data layer across services in an SOA. DataXtend CE can integrate with WebLogic, WebSphere, and Microsoft® .NET application servers. No other O/R mapping tool offers such complete language and platform support.

Tuning Features Ensure Data Access Performance

Simply getting an application to run is only half of the battle. Meeting response times and transaction rate requirements during load testing is equally important for project success. Simplistic or inefficient O/R mapping and a stateless architecture cannot meet the needs of applications with high request and/or transaction rates.

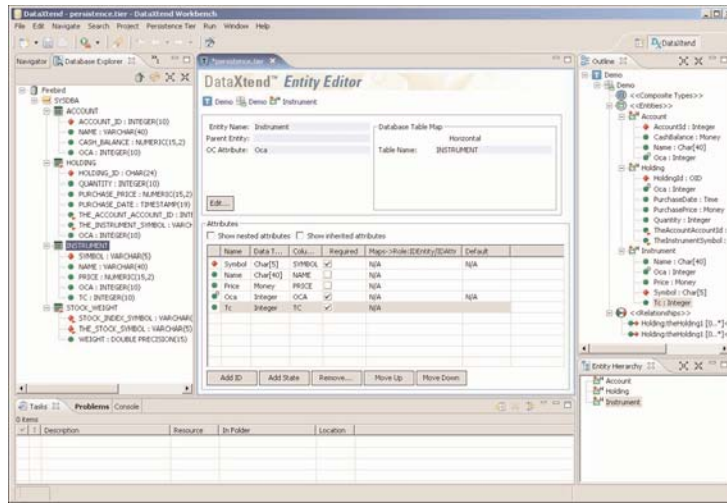


Figure 3. DataXtend CE Plug-Ins for Eclipse simplify development of the data layer and increase productivity. The Database Explorer view (long pane on left) provides drag-and-drop convenience for applications that will use existing tables.

At this stage of the project lifecycle, decision-makers tend to throw bigger hardware at the performance problem or add caching software. However, both of these strategies are costly. While hardware has become more affordable, hidden costs include personnel for maintenance and the development time required to distribute the application – not to mention additional database licenses and operational expenses. In some cases, disproportionate additions of computing power are required to achieve adequate performance.

As a software solution, Java server-side caching is well-accepted for increasing performance and scalability. Homegrown or bolt-on caches can increase performance but also require extra application coding. The additional programming, testing, and maintenance add up quickly. Caching that relies on Java serialization can also lose performance benefits due to the overhead of object serialization.

J2EE application servers include limited general purpose caching that works best for simple data that does not change often. In contrast, DataXtend CE for Java's caching was designed for applications with complex object models and dynamically changing data. Architects and developers who take advantage of DataXtend CE's development features can rest assured that performance and scalability are already built-in to their application. By allowing multiple clients safe, concurrent access to cached data, DataXtend CE relieves the load on the database and eliminates bottlenecks caused by redundant database queries.

DataXtend CE customers have dramatically reduced the frequency of database access in their applications. Depending on the balance of read versus write operations, response times improve anywhere from 50-1000%. Customers typically see several-fold improvement in the number of concurrent users that can be served using the same hardware.

Built-In Intelligent Caching

There are several approaches to caching objects in an application. Some caches use data structures that mirror database rows, columns, and tables, thus exposing schema details to the application code. Other caches treat objects as opaque data, which makes it impossible for the cache to effectively understand and manage relationships between objects. Both of these approaches generally work best for applications dealing with simple data that does not change frequently. For more complex situations, the burden falls on developers to programmatically integrate the application, the database, and the cache.

In contrast, DataXtend CE's intelligent caching offers several key features to support applications with complex object models and/or high request rates:

- **Code-free cache management:** DataXtend CE's object-relational mapping classes are integrated with the cache – objects are automatically instantiated in the cache when the application accesses them. Developers do not

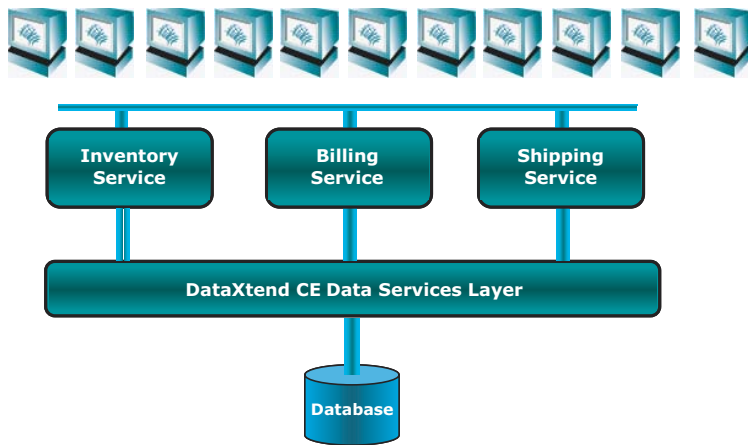


Figure 4. A DataXtend CE enterprise data services layer provides high performance and scalability in an SOA.

have to manually track which objects must be cached, when they should be cached, or when they should be cleared from the cache.

- **Data abstraction:** The intelligent cache knows both the object model and database schema. This ensures that DataXtend CE materializes exactly one cache object for each row that it loads from the database. Developers do not need intimate knowledge of the database schema nor do they need to actively manage data integrity. They can take advantage of object-oriented benefits such as inheritance and encapsulation.
- **Safe, concurrent access:** A multi-threaded application can simultaneously read and/or update cached objects. DataXtend CE ensures data integrity and avoids supplying stale versions of the object to the application's threads. Applications can serve more client requests with fewer database connections.
- **Dynamic relationship caching:** Intelligent caching remembers not just the values of object attributes, but also the relationship between objects. Unlike other caching products, DataXtend CE can dynamically instantiate relationships, or it can pre-populate the cache with frequently-used relationships. By caching relationships, DataXtend CE can dramatically reduce the number of expensive join queries performed by the database, resulting in a corresponding improvement in performance.

While most applications perform well with the intelligent cache's default settings, every application has its own data access patterns. DataXtend CE's advanced cache management service and database tuning features (such as cache indexing, lazy loading, and database write policies) are configurable and do not require code changes or recompilation.

Continuous Cache Coordination

Object-oriented applications built on the advanced caching capabilities of DataXtend CE operate in complex environments where other enterprise applications that make use of traditional SQL access methods may be modifying the data as well. It is critical that updates made by these other enterprise applications are reflected in the object cache in a timely and consistent way.

Typical O/R mapping and caching infrastructures follow a traditional "request-response" model for accessing data. In this model changes made by other enterprise applications would not be seen until an explicit request is made to reload the cached data. This can result in accessing stale or even inconsistent data.

To address the caching requirements of these complex environments, DataXtend CE features Continuous Cache Coordination to proactively "push" changes made to the database out to the distributed cache. This ensures that the enterprise data caching infrastructure maintains fresh and

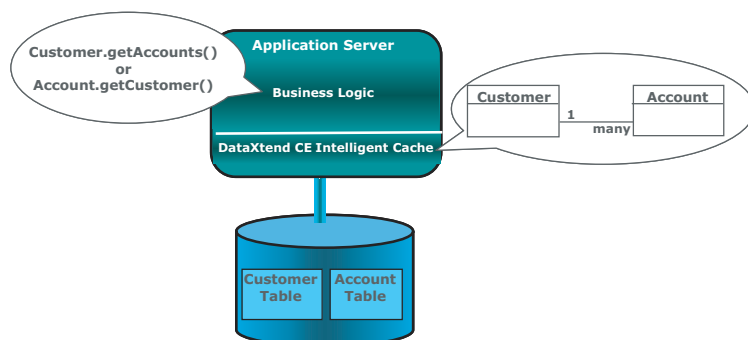


Figure 5. The intelligent cache understands the database schema and the application data model, simplifying data access while ensuring data integrity.

consistent data. The advanced data caching technology of Continuous Cache Coordination is only available with DataXtend CE.

Deployment Features Deliver Data Access Scalability and High Availability

Application success depends on the scalability and resilience of the production system. Ironically, business success can trigger application failure when unexpected demand overwhelms the system. Sluggish response times can doom an otherwise elegant application, as will incorrect data. Similarly, without fault tolerance, the data access layer is much more brittle, allowing even minor network or database outages to threaten application availability.

For the most demanding, data-intensive applications, DataXtend CE's deployment features ensure that the performance and scalability achieved during preproduction continues through rollout – within the data center and beyond. Applications based on DataXtend CE's data services platform can achieve cost-effective scalability and reliability with cache clustering. DataXtend CE for Java takes full advantage of the stateless clustering features provided by WebLogic, WebSphere or other application servers and complements them with stateful cache synchronization, guaranteeing fresh data for all users and supporting load balancing and failover implementations.

Scalable, Resilient Cache Synchronization

Cache synchronization uses messaging to ensure that the distributed caches remain up-to-date. DataXtend CE cache synchronization offers a choice of delivery modes: express and guaranteed.

Express synchronization sends the synchronization message immediately after a transaction completes when speed is the highest priority. Guaranteed synchronization integrates the sending of the synchronization message with the transaction and also writes the message to a persistent store, protecting against data loss even in the face of multiple system failures.

Because DataXtend CE's synchronization requires only basic configuration, servers can be brought online at any time to handle load increases or spikes, future proofing your system. DataXtend CE also provides automatic reconnection to the database, insulating users from a common cause of application failures and increasing uptime.

Cache synchronization also provides key benefits for advanced deployments involving service-oriented architectures or remote data centers:

- No IT department can effectively build custom SOAs without first implementing a shared data layer that guarantees data consistency across all services. Yet, most IT groups today use ad-hoc data access solutions such as JDBC that are unable to support enterprise-wide data consistency. Progress Real Time Division provides data access infrastructure software specifically designed to provide consistent performance and highly available data across distributed computing environments.
- The advanced synchronization capabilities within DataXtend CE allow cost-effective deployment to multiple geographic locations. A "virtual data center" architecture requires only one central physical data store and uses replicated caching to enable applications to be deployed to remote locations. By eliminating the need to replicate expensive databases and the personnel and hardware they require, virtual data centers dramatically reduce infrastructure and maintenance costs, while ensuring data consistency.

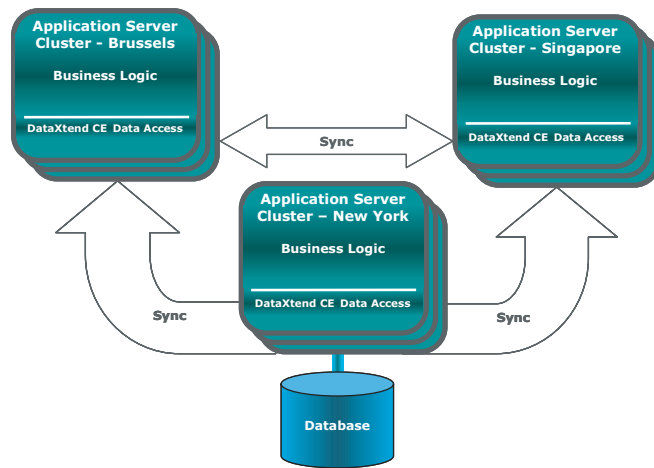


Figure 6. Virtual data centers use cache synchronization to ensure data consistency and improve local response times.

Services to Ensure Success

Our world-class Customer Care organization offers tiered support and consulting options that cover you from application design through deployment – even in situations when you might encounter problems from interactions with other vendors’ products. Progress Real Time Division’s customer-focused offerings include:

- Dedicated Technical Account Managers to assist every customer through successful deployment.
- Intensive online and classroom training that brings developers up to speed quickly and ensures that they take full advantage of DataXtend CE’s capabilities.
- Expert services to provide advanced troubleshooting in situations where other vendors’ tools may be affecting your application performance.
- An experienced Professional Services team offers architectural assessment, design and implementation assistance for even faster deployment.

The Leader in Data Access Infrastructure

The Progress Real Time Division is the only open and cross platform technology provider of data access infrastructure. Major partners like Adobe, Sun, Cisco and Sybase, have validated Progress Real Time Division’s leadership by licensing our state-of-the-art technology, and our award winning software has been recognized in Intelligent Enterprise’s “IT Dozen,” Java Report’s “Great Java Products,” “Software 500” and the Java Developer’s Journal “Editor’s Choice Awards”.

More than 100 Global 2000 customers rely on Progress Real Time Division products to provide data access for the world’s most demanding applications including real-time trading systems, transaction-intensive logistics and package delivery systems and complex network management systems. To experience DataXtend CE for Java’s developer productivity, performance, scalability and availability for yourself, visit www.progress.com/realtime and download a free 30-day evaluation.

Progress DataXtend CE for Java Specifications

Compatibilities

IBM® WebSphere®
 BEA® WebLogic®
 Custom Java® applications

Database Support

IBM® DB2
 Informix®
 MySQL®
 Oracle®
 SQL Server™
 SYBASE

Progress DataXtend CE for Java Specifications *(continued)*

Development Platforms

Windows® XP

Windows® 2003

Suggested Development Configuration:

Processor: PIII or greater

RAM: 256 MB or greater

Disk space: 165 MB

Deployment Platforms

AIX®

Windows® XP, 2003

HP-UX®

Linux

MacOS®

Solaris®

Progress DataXtend CE for Java Features

New in Version 3.1

Continuous Cache Coordination	Pushes changes made by traditional SQL access-based applications from the database to the distributed cache, maintaining consistency and high performance
SonicMQ support	Adds this popular messaging platform, providing more options for deployment
Bundled JDBC drivers	Simplifies your development needs for accessing existing database schemas, no time wasted in obtaining and installing the correct JDBC driver
Large cache support	Caches larger than 4GB are now possible with DataXtend CE's support for 64-bit memory addressing

Development Features for Productivity: Object-Relational Mapping

Tools include Plug-Ins for Eclipse, Add-Ins for Rose, and DataXtend CE Builder	Support a variety of development processes to increase productivity, giving you the choice to work with an IDE, a modeling tool, or a simple standalone interface to define your DataXtend CE project.
Database Explorer	Included with the Plug-Ins for Eclipse, supports connecting to a database, browsing schemas, and importing schema elements into a DataXtend CE project to further increase developer productivity
Bundled JDBC drivers	Simplifies your development needs for accessing existing database schemas, no time wasted in obtaining and installing the correct JDBC driver
Primitive data types	Maps object attributes to database column types, eliminating the need to write conversion routines
Composite data types	Combines multiple columns into a single object attribute structure, simplifying access to logically related information
Multi-attribute primary keys	Uniquely identifies objects with a combination of attributes, preserving object identity without changing the database schema
Deletion propagation constraints	Models the existence properties of relationships, preserving data integrity
Character set support	Transparently translates character strings between application and database, simplifying development of international interfaces
Database-independent APIs	Abstracts underlying database from application logic, allowing transparent switching of underlying database without re-coding
Database-specific optimizations	Automatically exploits database-specific features when possible, increasing overall performance

Development Features for Productivity: Object-Relational Mapping (continued)

Stored procedures	Enables compliance with administrator-imposed data access rules, avoiding intrusive disruption
Optimistic concurrency	Automatically tracks object modifications, protecting data integrity while supporting
Object locking	Manages concurrent access to shared data with row-level locks or version attributes, enabling fast concurrent access to data without compromising integrity
Inheritance	Supports union and horizontal mapping, optimizing application performance for different database schemas
Cardinality constraints	Captures the semantic properties of related objects, protecting database integrity concurrent reads for application-level scalability
Transactional cache	Isolates object modifications within a transaction until commit, preserving ACID properties while supporting concurrent access for performance and scalability
Object pre- and post- processing hooks	Supports the addition of custom logic, simplifying integration with other applications and providing additional flexibility
Interactive command-line tool	Supports direct calls to DataXtend CE APIs, providing a better evaluation experience and simplifying testing efforts

Tuning Features for Performance: Intelligent Cache

Object-model aware	Understands both the application object model and the database schema, preserving object uniqueness
Relationship aware	Caches relationships, dramatically shortening response times
Continuous Cache Coordination	Pushes changes made by traditional SQL access-based applications from the database to the distributed cache, maintaining consistency and high performance
Multi-class fetch	Pre-loads cache with objects and relationships, improving efficiency, increasing cache hit rates, and decreasing redundant database queries
Indexing	Builds an in-memory index for non-key attributes, speeding retrieval of cached objects
Eager and lazy object loading	Supports retrieval from large result sets as needed, reducing network round trips
Eager and lazy relationship loading	Allows related objects to be loaded on initial access or as needed, providing flexibility for improved performance
Large cache support	Caches larger than 4GB are now possible with DataXtend CE's support for 64-bit memory addressing
Clearing policy	Provides explicit lifecycle management, clearing unneeded objects to improve cache efficiency while ensuring that frequently-requested objects remain to increase cache hit rates
Transient objects	Exposes all cache features, providing consistent infrastructure for application data that will not be written to persistent storage
Monitoring APIs	Supports creation of routines to monitor runtime execution, simplifying tuning

Tuning Features for Performance: Database Optimizations

Connection pooling	Automatically manages shared connections, reducing the load on the database
Array reads	Exploits database-specific optimizations, decreasing network traffic and increasing performance
Hints	Allows application to specify query hints, optimizing performance for complex queries
Write policies	Sends changes to database immediately or only at commit time, providing the best performance for each circumstance

Tuning Features for Performance: Database Optimizations (continued)

Lazy attribute loading	Loads large attributes like BLOB and CLOB as needed, reducing network traffic and memory consumption
Batch updates	Sends multiple updates in one database request, reducing network round trips and increasing efficiency
Sparse updates	Sends only modified attributes to the database, improving performance for objects with many attributes
Partitioning	Allows object data to be distributed across multiple databases, supporting easy integration with existing data

Deployment Features for Scalability and Resilience: Cache Synchronization for Clustered Servers

Auto-configuring synchronization	For TCP messaging, allows clustered servers to find each other on a local network, simplifying deployment
Native application server support	Integrates seamlessly with J2EE application servers' clustering features, leveraging your investment
State-based updates	Transmits object state instead of simply invalidating cached objects, supporting caching of dynamic data and increasing scalability
Express messaging	Ensures cache consistency without writing to disk, maximizing performance (Supports Sonic MQ, Tibco Rendezvous, WebLogic JMS, WebSphere JMS, or selfconfiguring TCP messaging)
Guaranteed messaging	Temporarily writes synchronization messages to persistent store, ensuring cache integrity across application or network failures (Supports Oracle AQ and MQ Series messaging)
Open API	Allows replacement of synchronization transport, simplifying integration with corporate messaging standards
Cross-platform support	Enables synchronization across applications written in different languages and deployed on different platforms, providing the only enterprise solution for heterogeneous environments

Deployment Features for High Availability

Failover	Complements application server's failover capability with synchronization, ensuring that cached data is consistent and increasing uptime
Load balancing	Keeps the cached data in clustered applications synchronized, allowing requests to be correctly handled by any server
Automatic connection recovery	Releases and re-establishes dropped database connections within a pool, insulating application code from errors
Manual database reconnect	Detects failures, closes and reopens all connections in the pool, accelerating recovery from intermittent database outages

PROGRESS
SOFTWARE

www.progress.com

Specifications subject to change without notice.

© 2005 Progress Software Corporation.

All Rights Reserved.

EXJ/BC/0705-

Code 2483



0000105333

Worldwide and North American Headquarters

Progress Real Time Division, 14 Oak Park, Bedford, MA 01730 USA Tel: +1 781 280 4000

UK and Northern Ireland

Progress Real Time Division, 210 Bath Road, Slough, Berkshire, SL1 3XE England Tel: +44 1753 216 300

Central Europe

Progress Real Time Division, Konrad-Adenauer-Str. 13, 50996 Köln, Germany Tel: +49 6171 981 127

France

Progress Real Time Division, 3 Place de Saverne, Les Renardières B, 92901 Paris la Défense Tel: +33 1 41 16 16 56

www.progress.com/realtime

© 2005 Progress Software Corporation. Progress and DataTend are trademarks or registered trademarks of Progress Software Corporation in the U.S. and other countries. Java and all Java-based marks are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries. Any other trademarks or service marks contained herein are the property of their respective owners.