WebSphere software

IBM

Distributed Caching for Performance and Scalability: Save Time and Money with the WebSphere DataPower XC10 Appliance

Redguides
for Business Leaders

Carla Sadtler

- Maximize the performance and reliability of applications using expandable grids
- Protect data in the grid using built-in security features
- Avoid downtime and manage data grids with ease

Redbooks

Executive overview

Businesses today face many challenges in meeting their growing IT requirements. Investments made years ago in networks and infrastructure are aging. As the business grows, the challenge becomes how to use the existing infrastructure at its highest capacity to handle the growth in business.

Scalability is the ability of a system to handle increasing load in a graceful manner, implying that a system can be readily extended as your business grows. The ability to scale directly impacts the business applications and the business unit that owns the applications. You want to be able to scale your product with predictable costs and without major disruption when the business grows.

The traditional method of scaling systems is to improve the memory and processing power of existing systems and to add systems to spread the workload. In most cases, the web systems visible to the users and the middleware systems that host the business applications can be effectively scaled with additional servers to provide additional memory and processing cycles to service requests. However, as these systems are scaled to handle additional load, the load is also increased on the data servers that contain business and application data, often creating a bottleneck for throughput.

Data servers present their own challenges when scaling to handle the demand for large amounts of data. While the same approach taken to scale web and middleware systems is a viable approach, challenges arise in keeping the data servers synchronized for data integrity and crash recovery. The communication between servers that is required for synchronization can grow exponentially as the number of data servers increases, ultimately limiting their scalability.

A typical approach to resolving performance and, therefore, scalability problems that are inherent with data servers is to implement some form of caching of the data. Simply defined, the *cache* is a copy of frequently accessed data that is held in memory to reduce the access time to the data. Caching data has the effect of placing the data closer to the application, which improves performance and throughput. It also reduces the number of requests to the database and, thus, reduces that resource's potential as a bottleneck in the application. A cache, then, can extend the storage capability and can act as a *shock absorber* to the database.

The IBM® WebSphere® DataPower® XC10 Appliance provides a solution to the scalability challenge for data. The WebSphere DataPower XC10 Appliance caches data from the data servers for high-speed access by applications. The cache acts as a cushion for the data servers by storing frequently used data, thus reducing the amount of read and write accesses to the data servers. This caching tier is situated between the application server tier (which hosts the business applications) and the data tier (Figure 1). The cache is easily expandable and can store and manage large amounts of data securely with integrity and with consistent performance.

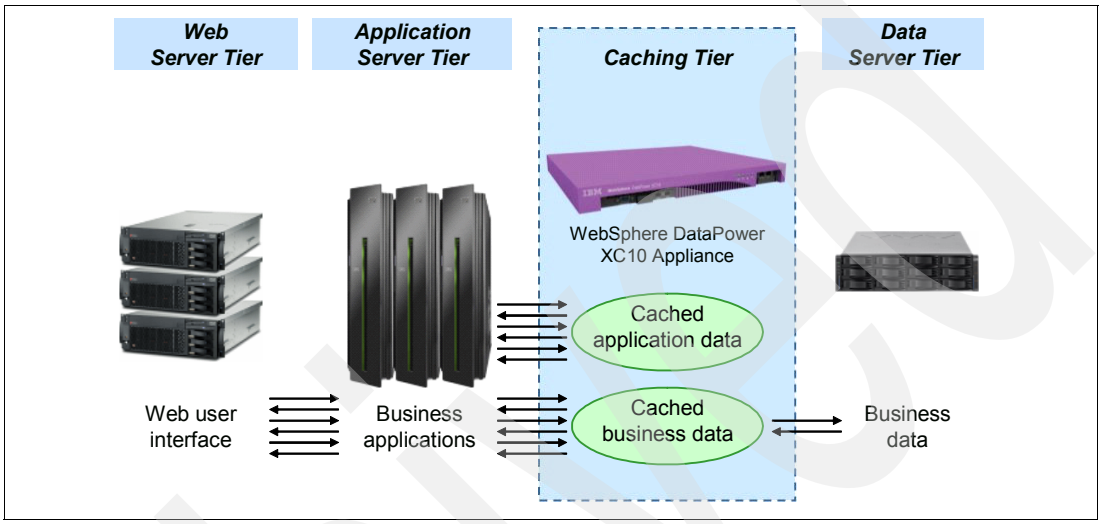

Figure 1 Application infrastructure topology

This IBM Redguide™ publication provides an overview of the WebSphere DataPower XC10. It explains how you can save time and money using distributed caching with the WebSphere DataPower XC10 Appliance. Although business executives and solution architects will find this information helpful, the guide is also useful to anyone who is interested in learning more about the XC10.

WebSphere DataPower XC10 Appliance: The scalability solution

The WebSphere DataPower XC10 Appliance (Figure 2) is an easy-to-use appliance that provides elastic caching functions. It is designed for simple, rapid *drop-in* deployment. With the XC10, you can scale business-critical applications cost effectively with consistent, high performance.

Figure 2 IBM WebSphere DataPower XC10

The XC10 features an expandable data grid, simple management features, and security for your data. With multiple XC10s, you can build in fault tolerance for your data, ensuring high availability.

Caching data close to applications in an expandable data grid

A data grid stores data in a *grid style* manner (Figure 3) using a large amount of loosely coupled cooperative caches to store data. The WebSphere DataPower XC10 Appliance is capable of holding one or more data grids. Each grid is associated with a specific application or set of applications. The WebSphere DataPower XC10 Appliance contains a 160 GB of storage for grid capacity. If this capacity is not enough, you can expand the grid by adding one or more appliances to the configuration, which creates a collective of appliances that hosts data. This capability of grouping appliances allows you to increase cache capacity and data throughput quickly and easily as needs grow.

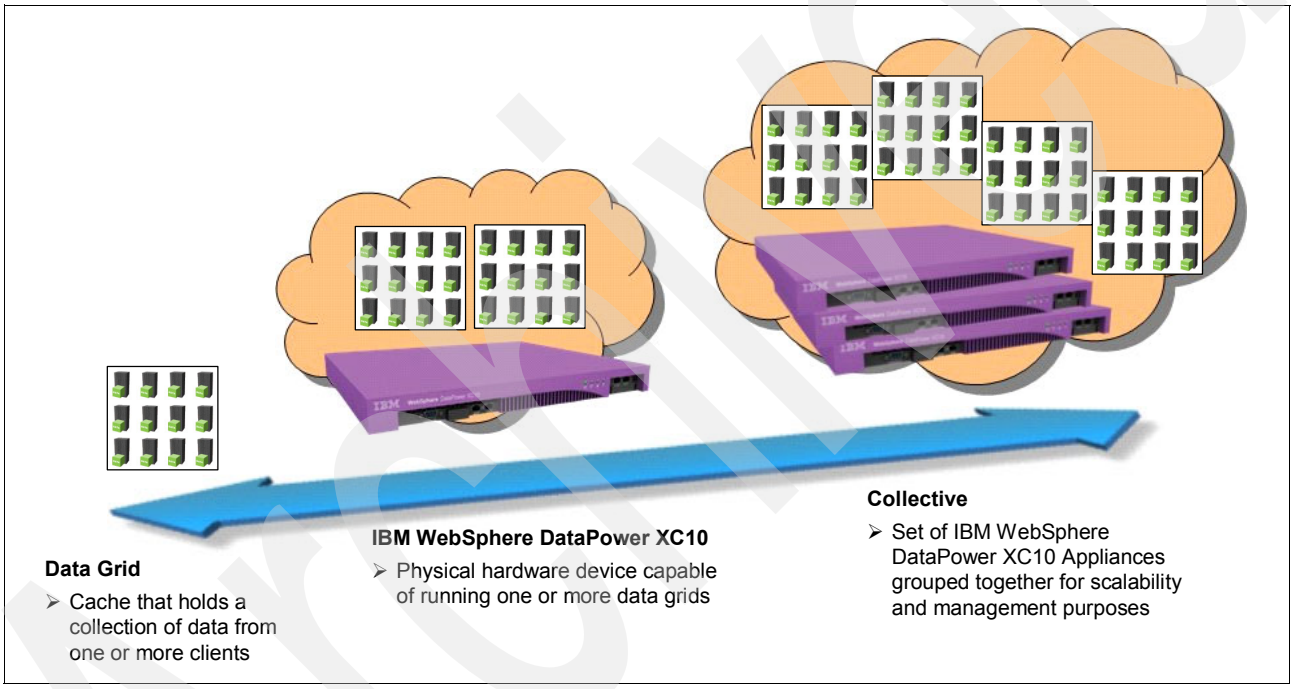

Figure 3 Data grids

The WebSphere DataPower XC10 Appliance maximizes the performance and reliability of applications using data grids designed for specific solutions. The types of grids include:

- *Session* data grids are used by WebSphere Application Server to manage data that is used during the course of a web session.
- *Dynamic* cache grids are used by the dynamic cache service in WebSphere Application Server to store frequently accessed web pages.
- *Simple* data grids hold data that is accessed directly by applications.

Using drop-in scenarios with no required code changes

The WebSphere DataPower XC10 Appliance complements the WebSphere Application Server family of products by including two drop-in scenarios to provide a distributed caching platform for WebSphere business applications. These drop-in scenarios are designed to run with any WebSphere product that is based on WebSphere Application Server V6.1 and V7.0.

In this section, we describe these two scenarios:

- Caching user session data
- Caching dynamic web pages

Caching user session data

Many web applications use the HTTP session state management features provided by Java™ Enterprise Edition (JEE) application servers. Session state management allows the application to maintain user information and the state of the session for the period of a user's interaction with the application. The traditional example for this scenario is a shopping cart, where information about intended purchases is stored until the shopping session is completed.

To support high-availability and failover, the state information must be available to all application servers, which application servers typically achieve by storing the state information in a shared database or in the application server's memory. In the latter case, session data is sent to each application server to keep the data in sync.

When session data is stored in a shared database, scalability is limited by the database server. The disk I/O operations are an expensive performance cost to the application. When the transaction rate exceeds the capacity of the database server, the database server must be scaled up.

When session data is kept in memory and replicated between application servers, the limits to scalability vary depending on the replication scheme. Commonly, a simple scheme is used in which each application server holds a copy of all user session data. In this scheme, the total amount of state information cannot exceed the available memory of any single application server. Memory-to-memory replication schemes often trade consistency for performance, meaning that in cases of application server failure or user sessions being routed to multiple application servers, the user experience might be inconsistent and confusing.

With the WebSphere DataPower XC10 Appliance, session data is kept in the grid, providing high-speed access to the data from all application servers. If one server goes down, the session can be continued on another server with the session data kept intact.

Configuration is simple with support for the WebSphere DataPower XC10 Appliance built into the application installation process. You can specify an existing grid or create a new grid directly from the WebSphere Application Server console.

As a practical example of this type of scenario, consider an insurance company that provides an application on its internet site that allows customers to get a price quote on car and home insurance. As potential customers using the site navigate through a series of pages, they enter information about the assets that they want to ensure, and they make selections regarding coverage. This information is collected and stored by the web application as session data to be used to create the price quote. At the end of the process, the user reviews the information and submits it for processing.

In this scenario, user performance is critical, because users tend to become frustrated with slow movement between pages. While the default in-memory session management that was used by the application serving environment worked, the replication that was required to keep

the data synchronized for availability purposes put a load on the application servers and caused performance delays for the user. In addition, the accumulation of session data due to the length of time that it took users to complete the application became a burden on the memory of the application servers.

By moving this session data to a collective of XC10 appliances, several critical improvements occurred. The response times to the user became more consistent and faster. Memory was freed for executing the application rather than storing data. Synchronization of data between the application servers was no longer necessary, which reduced the load on the application servers. Reliability was also improved. The collective contained replicas of the data so that if an application server went down, the data was not lost with it. The remaining servers could continue the session without losing the data.

Caching dynamic web pages

Many web-based applications use dynamic page generation techniques for pages that rarely change, such as product details or information about corporate policies. Application servers generally provide at least an in-memory cache to store the generated output the first time that the page is rendered in order to save the processing work and back-end system load for subsequent requests.

The WebSphere Application Server dynamic cache service stores the output of dynamic web pages for future use in memory, using replication to share the data among application servers. When the in-memory capacity is reached, the data is off-loaded to shared disk and disk access time becomes an important factor in application performance. Faster disk technologies, such as Storage Area Networks (SANs), are often used to improve performance.

The XC10 provides a drop-in replacement for the dynamic cache service's default cache provider. The XC10 stores cached data in a grid, allowing the size of the cache to expand to the sum of all available memory in XC10s in the grid. Cached data is readily available, making the application's performance more reliable and eliminating load spikes to back-end systems as applications are scaled. The XC10 dynamic cache service provider can compress the contents of the cache to get the most from the memory available.

As an example of this type of scenario, consider an airline that posts information about current and future flights on its website. The site is used by customers who are looking for available flight information, by passengers checking to see whether flights are delayed or canceled, and by others checking for flight arrival information. Users can view a list of flights by date, select a specific flight for more information, or select multiple flights for comparison.

The flight information is stored in a database and accessed by the web application. While the information about flights for the current day tends to change frequently, the data for future flights is fairly static. Dynamic caching is used to store the data that is frequently accessed but infrequently changed, thus reducing the time and resources that are required to build the web pages.

The high volume of requests, however, puts a significant load on the application servers, increasing response times and making the performance erratic. By introducing a collective of XC10 appliances, the company offloaded the memory and computing resources that are required to maintain the cache from the application servers, increasing both throughput and performance.

Using a simple cache for application data

Applications can use a data grid in the XC10 as a simple cache. The application checks the cache every time that data is needed. If the value is not found (cache miss), then the data is retrieved from the backend database and inserted into the cache. The next time that data is called for, it is retrieved from the cache.

Simple data grids can be used with WebSphere Application Server or with a stand-alone Java application. In both cases, the WebSphere eXtreme Scale client is installed on the JVM. The application accesses the data grid using the ObjectMap API. The WebSphere eXtreme Scale client is available for download through the IBM Support Portal at:

`http://www.ibm.com/support/entry/portal`

While this type of use requires some alterations to your application code to take advantage of the data grid, it is an excellent way to provide high-speed caching.

As an example of this type of scenario, consider an animal rescue organization that has a website where potential adopters can view information about the animals that are available for adoption. The information about each animal is stored in a database and is updated on a regular basis with the adoption status, new pictures, and age and health information. These updates are not frequent, but users viewing the site need to know whether an animal is already adopted.

Accessing the database with every request has proven to be expensive in terms of response time. When the volume of requests is high, the database becomes a bottleneck, reducing performance significantly. Use of the site (and adoptions) dropped as these problems grew.

The application was modified to use the WebSphere DataPower XC10 Appliance as a simple cache. When a user requests information about an animal, the data is retrieved from the database and is cached for future retrievals. Caching the data significantly improved performance and provided the opportunity for future growth.

Securing your data

Much of the security functionality offered by the WebSphere DataPower XC10 Appliance to protect the data in the grid is built into the construction of the appliance. Additional security settings are included to provide security options for accessing the user interface.

The WebSphere DataPower XC10 Appliance includes the following security features:

- The appliance is contained in a tamper-resistant case. In addition, an intrusion detection switch in the chassis is monitored continuously. If the switch is triggered, the appliance does not start and must be returned to IBM before it can be started again. Additional elements, such as the tamper-resistant screws on the case, are also included to discourage opening the case. The design of the appliance ensures that you can access the customer-replaceable items from the rear of the appliance without opening the case.
- There is no access to the operating system through a command shell in the operating system of the appliance. By design, no command interpreters are included on the appliance to reduce security vulnerabilities. The appliance includes only one operating system user ID. You cannot log on to the appliance externally with a user ID, because there is no shell available.

- No user-provided logic can be run on the appliance. The appliance does not provide any ability for a user to upload an executable script or code. The only exception to this statement is a system firmware update, in which you can run a script to install updated firmware on the appliance. These system updates are signed by the firmware manufacturer as a precaution. No user-provided untrusted software can run on the appliance.

In addition to the security features for the hardware and operating systems, the WebSphere DataPower XC10 Appliance provides a high level of data security, allowing you to control access to the grids through authentication and authorization mechanisms. User and group information that is used to authenticate with the appliance can be defined in the appliance or can be taken from a Lightweight Directory Access Protocol (LDAP) directory. Users and groups are authorized for actions, such as creating data grids or administering and monitoring the appliance, directly through the appliance user interface.

Avoiding downtime (fault tolerance)

When you have grouped multiple appliances into a collective, the collective lowers the risk of data loss through the automatic replication of data. If an appliance housing the primary data grid fails, a replica on another appliance in the collective is promoted automatically to be the primary. Multiple replicas are supported and are transparent to the application that uses the WebSphere DataPower XC10 Appliance to store data. This fault tolerance setup provides continuous availability in the event of the loss of one of the appliances. The grid becomes self-healing, in that failures are detected and managed automatically.

Managing grids and collectives with ease

The WebSphere DataPower XC10 Appliance provides a web-based console that allows you to create, manage, and monitor data grids and collectives. You can also use the web interface to manage users and user groups for your appliance. The console provides simple, intuitive steps to create everything that you need to get started and provides guidance to help you complete each task. Figure 4 illustrates this console.

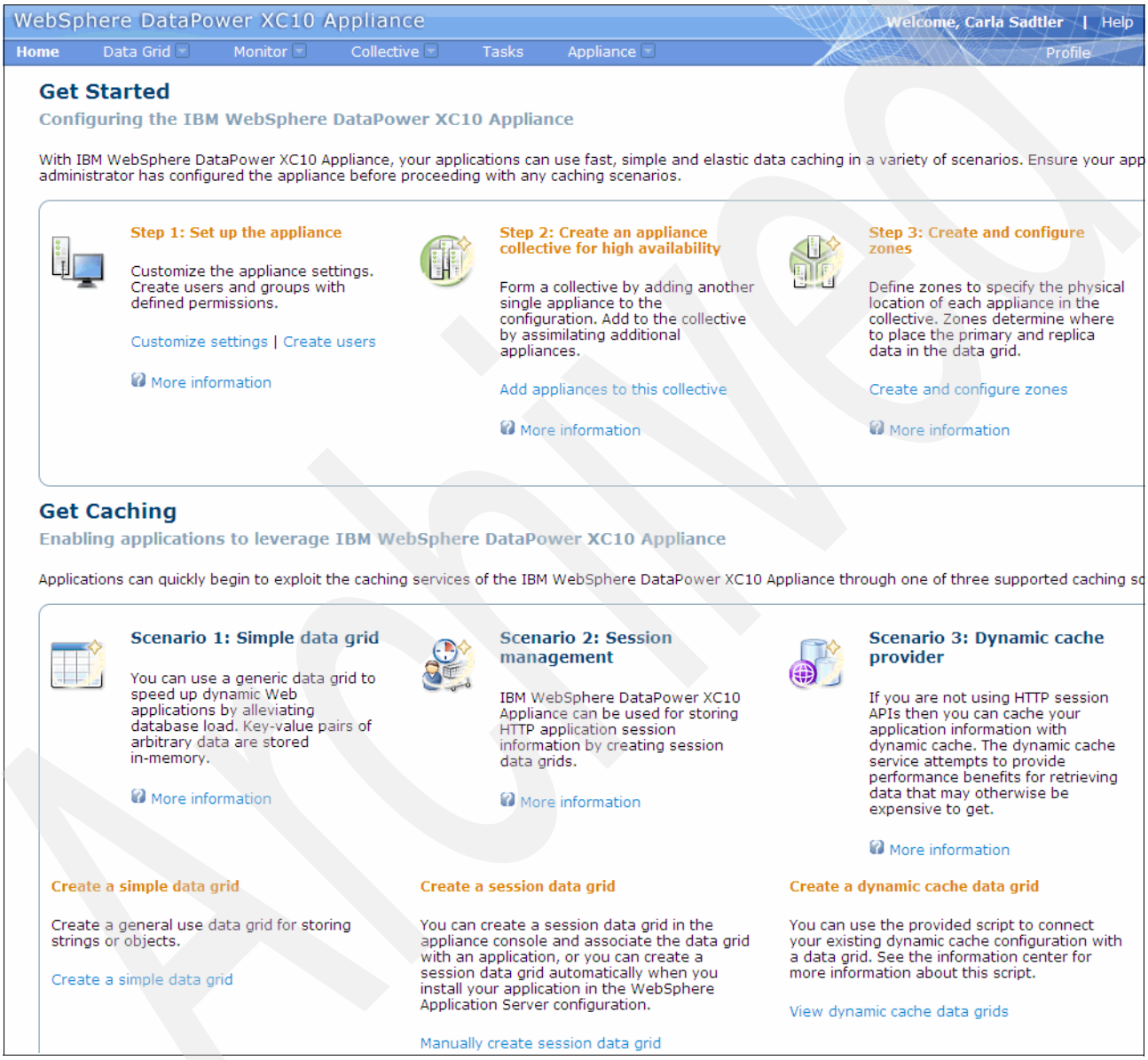

Figure 4 Welcome window in the appliance web console

Creating grids

Creating a grid is as simple as clicking a button and providing a name for the grid. Modifiable settings for the grid are minimal, including only security and replica settings. Figure 5 shows the window to set dynamic cache properties.

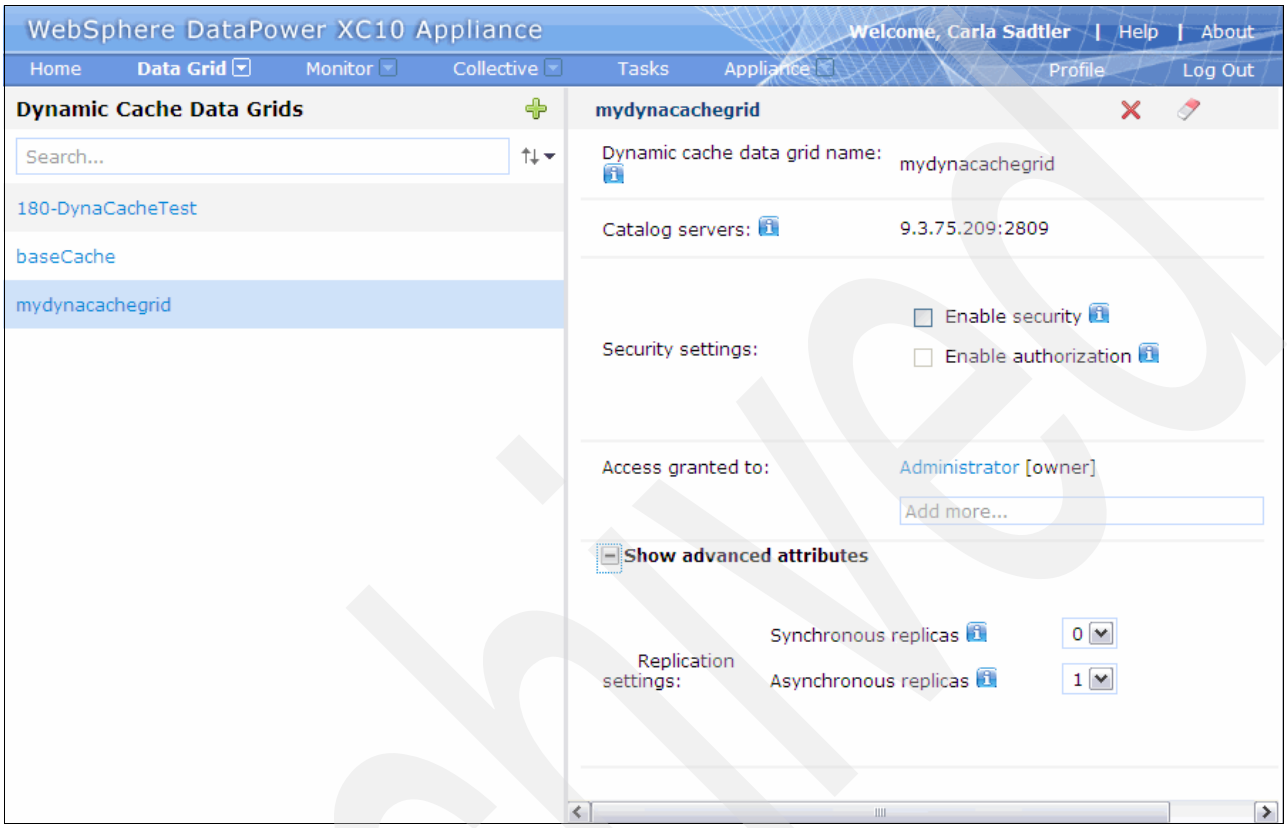

Figure 5 Dynamic cache properties

Monitoring grids

The console also provides real-time monitoring of the performance and health of the grids (Figure 6). Both summary and detailed reports are available. You can see the number of entries in the grid, average throughput and transaction times, used capacity, and other valuable information that can help you monitor the grids.

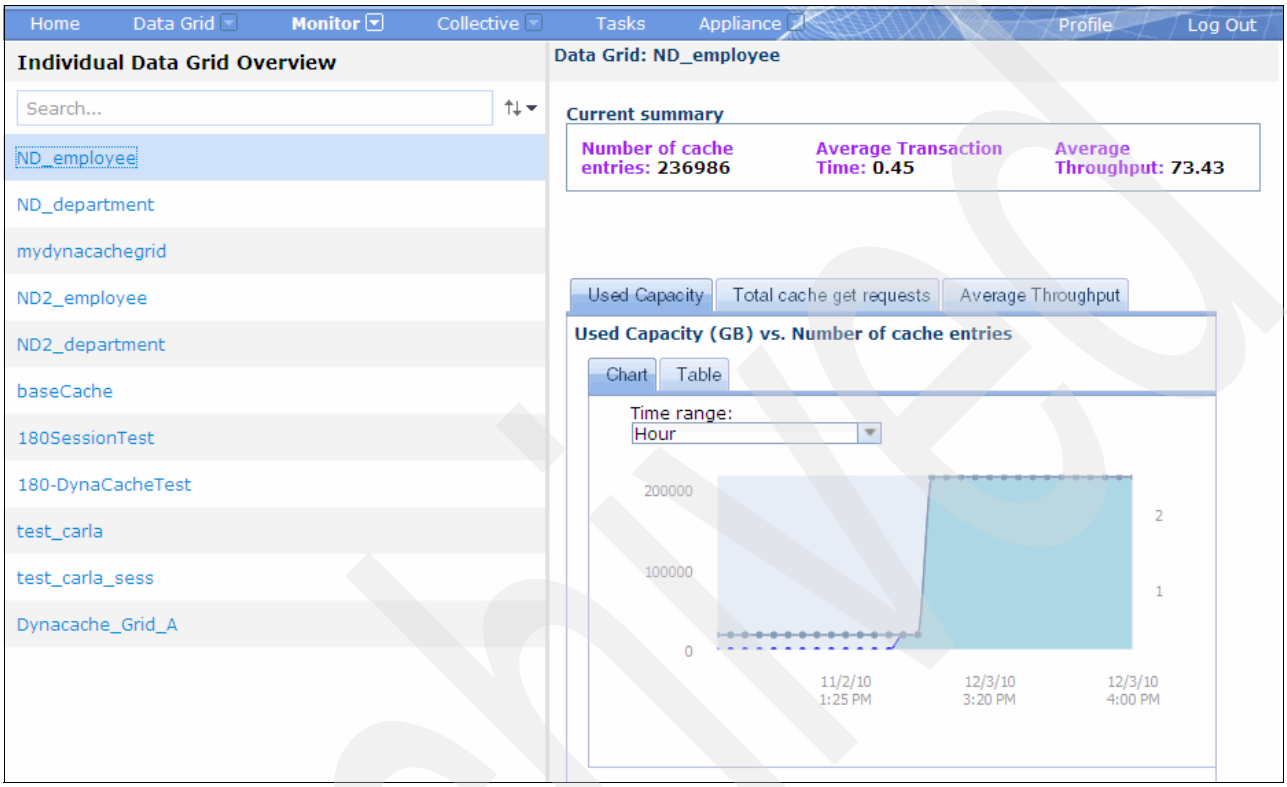

Figure 6 Grid monitor

Integrating with WebSphere Application Server

The WebSphere eXtreme Scale client code, installed into the WebSphere Application Server environment, seamlessly integrates the WebSphere DataPower XC10 Appliance with the application server environment. The option to use the XC10 as the dynamic caching provider is available from the administrative console, and options are built into the application installation process to allow you to select a data grid in the XC10 for session persistence.

Comparing the WebSphere DataPower XC10 Appliance to the WebSphere eXtreme Scale software offering

WebSphere eXtreme Scale software has long been recognized as the scalable caching mechanism of choice in WebSphere Application Server environments. Like the WebSphere DataPower XC10 Appliance, WebSphere eXtreme Scale supports a dynamic grid infrastructure with support for substantial scale outs. Both offer high performance for extreme transaction processing.

What are some of the most prominent differences? Where the WebSphere DataPower XC10 Appliance stores cached data in the appliance, WebSphere eXtreme Scale builds grids using

Java Virtual Machines (JVMs). In addition, whereas the WebSphere DataPower XC10 Appliance is designed for the three caching solutions (dynamic cache, session management, and simple side cache), WebSphere eXtreme Scale supports these scenarios and offers additional flexibility in the type of caching that it supports and how the cache is loaded.

Summary

The WebSphere DataPower XC10 Appliance provides a simple, yet powerful, way to improve the performance and availability of session-intensive or data-intensive applications. Mission-critical applications benefit from the performance and consistent response times that can be achieved by caching data in the WebSphere DataPower XC10 Appliance. The appliance is designed for the following uses:

- Simple data cache for applications
- Session persistence for web applications
- As a dynamic caching provider

As your data and transaction volumes grow, the caching capability can be expanded seamlessly by combining multiple appliances in a collective. The collective expands the workload capacity, provides high availability, and lowers the risk of data loss through automatic replication of data.

The installation and configuration of the appliance is quick, reducing the time to deployment. It provides simplified management through its web console and integration with WebSphere Application Server administrative tools.

Other resources for more information

The following websites offer additional information about the WebSphere DataPower XC10 Appliance:

- IBM WebSphere DataPower XC10 Appliance product page

  http://www.ibm.com/software/webservers/appserv/xc10/
- The Global WebSphere Community Presents: Save Time, Money, and Rackspace with IBM's Revolutionary WebSphere DataPower XC10 Appliance

  http://event.on24.com/r.htm?e=220930&s=1&k=DA2CFEE7A5FFEB1BF2926818432B2941
- IBM WebSphere DataPower XC10 Appliance developerWorks® wiki

  https://www.ibm.com/developerworks/wikis/display/extremescale/IBM+WebSphere+DataPower+XC10+Appliance
- IBM WebSphere DataPower XC10 Appliance Information Center

  http://publib.boulder.ibm.com/infocenter/wdpxc/v1r0/index.jsp
- IBM WebSphere DataPower XC10 Appliance forum

  https://www.ibm.com/developerworks/forums/forum.jspa?forumID=2247&start=0

The team who wrote this guide

This guide was produced by a specialist from the International Technical Support Organization (ITSO).

Carla Sadtler is a Consulting IT Specialist at the ITSO, Raleigh Center. She writes extensively about WebSphere products and solutions. Before joining the ITSO in 1985, Carla worked in the Raleigh branch office as a Program Support Representative, supporting MVS™ customers. She holds a degree in mathematics from the University of North Carolina at Greensboro.

Thanks to the following people for their contributions to this project:

Brian Martin
IBM U.S.

Lan Vuong
IBM U.S.

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:
IBM Director of Licensing, IBM Corporation, North Castle Drive, Armonk, NY 10504-1785 U.S.A.

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

This document, REDP-4678-00, was created or updated on January 24, 2011.

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. These and other IBM trademarked terms are marked on their first occurrence in this information with the appropriate symbol (® or ™), indicating US registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the web at
http://www.ibm.com/legal/copytrade.shtml

The following terms are trademarks of the International Business Machines Corporation in the United States, other countries, or both:

DataPower®	MVS™	WebSphere®
developerWorks®	Redguide™	
IBM®	Redbooks (logo) ®	

The following terms are trademarks of other companies:

Java, and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Other company, product, or service names may be trademarks or service marks of others.