

Demonstration: Active Asynchronous Transaction Management in High-Autonomy Federated Environment Using Data Agents: Global Change Master Directory v8.0

Omran Bukhres, Srinivasan Sikkupparbathyam,
Kishan Nagendra, Zina Ben Miled, Marcelo Areal
Indiana University Purdue University
Indianapolis, IN 46202-5132

Lola Olsen, Chris Gokey,
David Kendig, Rosy Cordova,
Gene Major, Janine Savage
NASA/Goddard Space Flight Center,
Greenbelt MD 20771

Abstract

The Global Change Master Directory (GCMD) is an earth science information repository that specifically tracks research data on global climatic change. Building a directory of Earth science metadata that allows the exchange of metadata content among partner organizations is challenging due to the complex issues involved in supporting heterogeneous metadata schema, database schema, database implementation and platforms. This demonstration presents the design of the MD8 (Master Directory v8.0), which allows automated exchange of metadata content among earth science collaborators through a proposed asynchronous distributed transaction protocol. Specifically, the demonstration will focus on the Local Data Agent (LDA) that captures local database updates and broadcasts them to other cooperating nodes asynchronously using an Announcer.

This demonstration focuses on the implementation of an asynchronous transaction protocol [1, 6, 8] which satisfies the requirement of replicating DIF entries through the internet across continents in the International Directory Network (IDN). We have designed a Primary Site Distribution protocol which is asynchronous but active i.e. participating sites in the network announce the arrival of new data to other sites in the network. This protocol also addresses the issues involved in dual ownership of entries which is a challenge that faces several emerging distributed transaction management systems [3, 4, 8].

The objective of the new system (MD8) is to promote the current architecture of the GCMD from a client-server architecture to a federated architecture that is totally automated while providing sites participating in the IDN network with the autonomy needed to maintain their local systems. A detailed overview of the design of the MD8 is included in [9].

1. Introduction

The Global Change Master Directory (GCMD) is a repository that contains information on the changing environment collected by various agencies including the United States government agency Global Change Data Center (GCDC) at NASA. Other agencies that actively collect similar type of information include the Canadian, Australian, Japanese and Dutch government agencies. The GCMD has been in existence for the past 11 years. It is a database that is growing in importance due to the availability of recent research on global climate change [2]. Furthermore, the GCMD is one of the few organized efforts to create a system that supports uniform storage, access and retrieval of change research metadata. At the core of the GCMD is a *DIF (Directory Interchange Format)*. DIF is a standard used to store and transfer data within the various sites in the IDN network. It consists of a collection of fields that describes the GCMD data.

2. MD8 Implementation

The asynchronous transaction model [6] is the model of choice for the MD8 implementation since the data in the various IDN sites need not be *synchronized* immediately. The *temporarily out-of-sync* condition of the data is well tolerated. The proposed design also stipulates that only one site (owner of DIF entry or the GCMD) can initiate and complete an insert, update or delete of a given *DIF* entry. This satisfies the *unidirectional update propagation* property of asynchronous transactions. However, a locking mechanism is needed for the case of *not-in-sync* updates that happen simultaneously at the GCMD site as well as the originating site. The asynchronous transaction model was selected because it also improves response time and prevents global deadlock. Furthermore, because the underlying architecture is a federated architecture, complex handshaking procedures associated with the two-phase protocol implementations are avoided.

When a new entry is added to site A_1 , the proposed design propagates the entry to sites $A_2 \dots A_n$. Site A_1 uses the Announcer module to invoke methods of the LDA (Local Data Agent) in every other site. Thus, to propagate an insert or an update, a specific method called PutActionItem is invoked. When a propagated item has been received, an acknowledgement is sent back to the site that originated the request. Locks are required for updates because of the dual ownership of DIF entries between the GCMD site and the originator. In order to support this feature, the proposed design includes methods for acquiring and releasing locks. Deleting an entry can be done both by the originating site of the entry as well as the GCMD site. Relevant information about the server and the other sites including their I/P locations are stored in a Java-RDBMS known as InstantDB on each IDN site. The number of IDN sites to which a given site and server have access is dynamic and can increase over time. The actual list of sites and their I/P addresses can be entered by the GCMD server administrator at the time of system installation or can be added later.

Each LDA provides a uniform external interface and a customized internal representation based on the target database type (e.g. Oracle, Sybase, etc.) The implementation of the LDA makes the heterogeneity of the underlying site at the level of the operating systems, database systems, schemas and storage formats transparent to the user. The Remote Announcer and the Receiver are the two components within the LDA that perform the asynchronous distributed transaction. MD8 implements asynchronous transactions by storing a schedule of inserts, updates and deletes and by broadcasting them whenever a remote connection is available to the respective sites. This MD8 requirement is similar to the Lazy Group Replication [5]. However, instead of having the entire group own all the objects being updated, there are only two owners for each object: the originating site and the GCMD master site. Additionally, read serializability of propagated entries is not considered in order to avoid network traffic. The IDN network is different from a banking or a finance application in various aspects including the fact that the IDN tolerates temporary inconsistency. The consistency and correctness of updates in the proposed system is protected through locks. The current implementation uses persistent locks.

3. Related Work

The concept of Master copy replication scheme [5] has been proposed to overcome the unstable behavior of indiscriminate replication. The authors of [5] propose tentative update transactions that are later applied to a master copy. This approach is related to the approach used in the proposed system. The difference lies in the fact that

the MD8 is a truly federated system, and does not have a master copy.

The implementation of transaction models based on asynchronous communication using a two-phase commit protocol discussed in [6] indicates that the two-phase transaction model did not provide the required response time for the Service Level Agreement for Boeing. In the MD8 environment, asynchronous transactions can greatly enhance response times while allowing temporary inconsistent states between source and target databases.

4. Conclusion and Future Work

The major advantage afforded by the implementation of the MD8 for the IDN network is the presentation of consistent and complete results for queries by accessing a unique (read any) site in the network. This delivers the speed of query execution of a centralized system with the comprehensiveness of the results of a distributed query execution system [7]. The MD8 also promotes the autonomy of the participating sites in accordance to the requirement of a truly federated system.

5. References

- [1] Database Transaction Models for Advanced Applications, A. Elmagarmid, Morgan Kaufman, 1990.
- [2] GCMD System Documentation Software version 7.0 <http://gcmd.gsfc.nasa.gov/>
- [3] Performance of Update Algorithms for Replicated Data in a Distributed Database, Hector Garcia Molina et al Stanford U., CA, June 1979.
- [4] How to Assign Votes in a Distributed System, Hector Garcia Molina, D.Barbara, ACM Journal, 32(4). October 1985.
- [5] The Dangers of Replication and a Solution, Jim Gray et al, SIGMOD 1996, Montreal, Canada.
- [6] The Need for Distributed Asynchronous Transactions, Lyman Do, Prabhu Ram, Pamela Drew, Boeing Phantom Works, Mathematics and Computing Technologies SIGMOD 1999.
- [7] Databases on the Web: Technologies for Federation Architectures and Case Studies by Ralf Kramer, Forschungszentrum Informatik, SIGMOD 1997.
- [8] Revisiting Commit Processing in Distributed Database Systems, by Ramesh gupta, Jayant Haritsa and Krithi Ramamritham, SIGMOD 1997.
- [9] Global Change Master Directory: Active Asynchronous Transaction Management in Federated Environment Using Data Agents, Zina Ben Miled, Omran Bukhres, Srinivasan Sikkupparbathyam et al., ACM Symposium on Applied Computing, 2001.