



Doctoral Thesis

Ganymed A Platform for Database Replication

Author(s):

Plattner, Christian A.

Publication Date:

2006

Permanent Link:

<https://doi.org/10.3929/ethz-a-005361356> →

Rights / License:

[In Copyright - Non-Commercial Use Permitted](#) →

This page was generated automatically upon download from the [ETH Zurich Research Collection](#). For more information please consult the [Terms of use](#).

Diss. ETH No. 16945

Ganymed: A Platform for Database Replication

A dissertation submitted to the
SWISS FEDERAL INSTITUTE OF TECHNOLOGY ZURICH

for the degree of
Doctor of Technical Sciences

presented by
CHRISTIAN ANDREAS PLATTNER
Dipl. Informatik-Ingenieur ETH
born January 21, 1976
citizen of Bretzwil/BL, Switzerland

accepted on the recommendation of
Prof. Dr. Gustavo Alonso, examiner
Prof. Dr. M. Tamer Özsu, co-examiner
Prof. Dr. Willy Zwaenepoel, co-examiner

2006

Abstract

This thesis deals with the construction of Ganymed, a platform for cluster based database replication. Database replication is typically used to either improve fault tolerance or to improve the performance of computational operations on the replicated data (or both). Ganymed, however, goes beyond those goals. So called satellite databases are not only used to achieve the goals of traditional replication, rather they can be used to extend existing databases in various ways.

In the thesis we first discuss RSI-PC, a new, efficient algorithm for database replication. The algorithm schedules transactions between a master database and a set of attached satellite databases. The idea is to focus on transactional web applications: in such environments, the commonly encountered scenario is a workload that consists merely of complex read-only transactions that are accompanied by few short and simple update transactions. Hence, while ensuring consistency and correctness at all times, the main focus of RSI-PC is to use read-only satellite databases to maximize the speed up for the execution of read-only transactions. Due to the fact that consistency and correctness are never abandoned, the resulting system acts like a single image system and therefore no changes are needed on the client side.

Based on the core system the thesis then shows how it can be extended to cope with various other issues: the flexibility and independency of a concrete backend database engine provider is shown by constructing heterogeneous setups where the master database and the satellites are different engines from different vendors. The ability to extend the functionality of the master database is shown by using Ganymed as way to implement new functionality on the satellites. By using Ganymed, clients are not aware that the added features do not belong to the master but rather are being offered by the satellites. An example of such a feature is time-travel, where clients can inspect older states of the database. By using the Ganymed approach, space requirements or query execution times on the master are not affected in any way, at the same time satellites can be used to offer arbitrary time-travel functionality.

The thesis furthermore shows that Ganymed is not limited to static setups. By using dynamic replication the system can vary the amount of attached satellites to a master as demand requires. Furthermore, the system can be used to implement multi-tenant support, i.e., many different customer databases can be handled in parallel. For each tenant the system then dynamically manages a tailored set of satellites. Last but not least it is important to note that the system and the proposed ideas in the thesis have been fully implemented.

Kurzfassung

Diese Dissertation beschäftigt sich mit der Architektur von Ganymed, einer Plattform für clusterbasierte Datenbank Replikation. Datenbank Replikation wird typischerweise eingesetzt um die Ausfallsicherheit und die Zugriffsgeschwindigkeit auf Daten zu verbessern. Ganymed geht jedoch weiter als traditionelle Replikation, indem es erlaubt, Datenbanken in verschiedenen Arten und Weisen zu erweitern.

In der Dissertation wird zuerst RSI-PC vorgestellt, ein neuer, effizienter Algorithmus für Datenbank Replikation. Der Algorithmus disponiert Transaktionen über einen Master Datenbank Server und eine Menge von sogenannten Satelliten Servern. Die Hauptidee ist die Fokussierung auf transaktionsbasierten Web-Applikationen. In solchen Umgebungen ist das Hauptszenario eine Last welche grösstenteils aus komplexen Nur-Lese Transaktionen besteht, welche von wenigen, simplen Schreib Transaktionen begleitet werden. Das Augenmerk von RSI-PC liegt deshalb auf dem Maximieren der Performance von Nur-Lese Transaktionen mittels Satelliten Datenbanken. Gleichwohl wird Korrektheit und Konsistenz jederzeit garantiert: Ganymed, welches RSI-PC implementiert, verhält sich wie eine normale Ein-Instanz Datenbank und infolgedessen sind keinerlei Änderungen an der Klientensoftware nötig.

Basierend auf dieser Grundarchitektur demonstriert die Dissertation dann in der Folge weitere Anwendungsmöglichkeiten von Ganymed. Die Flexibilität und Unabhängigkeit des Systems wird demonstriert durch die Konstruktion eines Setups in welchem der Master Server und die Satelliten Datenbank Software von verschiedenen Anbietern verwendet. Die Fähigkeit den Master Server, ohne diesen zu Verändern, mit neuer Funktionalität auszustatten ist eine weitere Anwendungsmöglichkeit. Durch den Einsatz von Ganymed sind sich Klienten nicht bewusst, dass sämtliche erweiterte Funktionalität transparent durch die Satelliten zur Verfügung gestellt wird. Ein Beispiel hierfür ist das sogenannte Time-Travel, bei welchem frühere Datenzustände von Klienten abgefragt werden können. Durch den Einsatz von Ganymed steigen die Platzanforderungen oder Antwortzeiten von Anfragen auf dem Master Server in keinerlei Weise, da sämtliche noch so komplexen Time-Travel Anfragen auf den Satelliten durchgeführt werden.

Im weiteren wird gezeigt, dass Ganymed nicht auf statische Konfigurationen beschränkt ist. Durch den Einsatz von dynamischer Replikation kann das System jederzeit auf geänderte Lastbedingungen reagieren und die Anzahl der benötigten Satelliten anpassen. Zudem können auch Multi-Klienten Systeme konstruiert werden, in welchen eine grosse Anzahl von Datenbanken von verschiedenen Klienten parallel betrieben werden. Für jeden Klienten wird dann spezifisch die Konfiguration der Satelliten angepasst. Zu guter Letzt muss betont werden, dass sämtliche Ideen dieser Dissertation komplett implementiert wurden.