Project no. 004758

GORDA

Open Replication Of Databases

Specific Targeted Research Project

Software and Services

Final Activity Report

Period covered: from October 2004 to March 2008
Date of preparation: June 2008

Start date of project: 1 October 2004 Duration: 42 Months

Project coordinator: Rui Oliveira
Project coordinator organisation: Universidade do Minho

Revision 1.0

Contributors

Alex Pilchin, USI
Alexandre Pinto, U. Lisboa
Alfrânio Correia Junior, U. Minho
Ana Nunes, U. Minho
António Sousa, U. Minho
Antti Keränen, Continuent
Bruno Matos, U. Minho
Christophe Taton, INRIA
Csaba Simon, Continuent
Damian Arregui, Continuent
Dulce Domingos, U. Lisboa
Emmanuel Cecchet, Continuent
Fernando Pedone, USI
Florent Métral, INRIA
Gilles Rayrat, Continuent
Hugo Miranda, U. Lisboa
Sylvain Sicard, INRIA
Jean-Bernard Stefani, INRIA
Jérémy Philippe, INRIA
José Marques, U. Minho
José Mocito, U. Lisboa
José Pereira, U. Minho
Lasaro Camargos, USI
Luís Rodrigues, U. Lisboa
Luís Soares, U. Minho
Marc Herbert, Continuent
Mathieu Peltier, Continuent
Miguel Matos, U. Minho
Nuno Carvalho, U. Lisboa
Nuno Carvalho, U. Minho
Paulo Jesus, U. Minho
Ricardo Vilaça, U. Minho
Robert Hodges, Continuent
Rui Oliveira, U. Minho
Sara Bouchenak, INRIA
Stephane Giron, Continuent
Susana Guedes, U. Lisboa
Sylvain Sicard, INRIA
Vaide Zuikeviciute, USI

Contents

Executive Summary

Replication and Data Management

The Information Society places unprecedented demands on database management systems (DBMSs). DBMSs are at the core of systems supporting a wide range of economic, social, and public administration activities as shaped by the eEurope 2005 initiative. As a result, high availability and prompt disaster recovery for the underlying DBMSs are essential. This makes database replication a key technology for the long-term competitiveness of today's businesses. By keeping continuously updated replicas of data, it becomes possible to withstand server failures and even complete data-center destruction.

Replication is challenging for three main reasons: the integration of legacy systems, the performance in wide-area and large systems and the cost of database management systems supporting replication. Innovation, in replication, is stifled by proprietary or inadequate interfaces for developing replication middleware. Currently, developing new replication protocols requires that a custom interface to the database management system be also implemented.

Replication solutions are therefore either implemented within the database core itself, impairing portability and inter-operability, or based on client access middleware, which makes it difficult to support advanced database features (e.g. stored procedures) and hinders performance. In sharp contrast, the wide availability of standard interfaces for client access middleware has sparked fierce competition and innovation based on inter-operability, bootstrapped by a useful set of tools and implementations. The lack of innovation in database replication is obvious in the protocols commonly in use. Most of them are single-master, impairing scalability, and asynchronous, with a serious impact on both consistency and resilience to failure. Synchronous and multi-master replication protocols are scarce and targeted at niche applications that can afford expensive hardware solutions or have restricted requirements. In contrast, research in database replication has shown that by using group communication it is possible to achieve strong-consistency synchronous and multi-master replication. Preliminary results indicate that such protocols can be scaled to large clusters and wide-area networks.

Project Goals

The goal of the GORDA project is to foster database replication as a means to address the challenges of trust, integration, performance, and cost in current database systems underlying the information society. This is to be achieved by:

- **Architecture and Interfaces** Promoting the interoperability of DBMSs and replication protocols by defining generic architecture and interfaces that can be standardized.
- **Innovative Replication** Providing general-purpose and widely applicable database replication protocols based on group communication for large-scale database systems.
- **Management Tools** Providing uniform techniques and tools for managing secure and heterogeneous replicated database systems. This means supporting, at architectural and protocol levels, complex environments by addressing partial replication, distributed execution and cross-border security issues.
- **Open Standard** Making all interfaces open and actively working to foster adoption of the interfaces and of the resulting generic replication middleware by different database and middleware providers.
- **Migration Path** Providing an interim middleware-based solution that allows immediate integration of current DBMSs, thus supporting the standardization effort.

Impact

Database management systems play a central role in e-services and thus in eGoverment, eLearning and eHealth as shaped by the eEurope 2005 initiative. GORDA results will contribute to the definition of the next generation information systems, exhibiting not only higher dependability and performance but also lower costs.

The project considers the specification and evaluation of novel extensions to database interfaces, such that replication protocols can operate with optimal performance.

By taking the standardization of these interfaces as a fundamental concern, GORDA will ensure that all vendors of database, middleware, and application products can benefit from the project results.

By bringing together partners from the database arena with partners with strong expertise in the middleware approach, the project will develop and validate an approach that combines the best of both worlds. This will represent a significant advantage for the several players in the market: For the end-users this will open the door for the availability of efficient solutions for database replication that are not necessarily tied to a single vendor or to proprietary interfaces; for database vendors, the opportunity to address the concrete requirements of specific markets without loosing compatibility with existing database replication middleware; for middleware vendors, an opportunity to expand the range of applicability of their replication management products, over many different database vendors.

By providing the tools to create a larger interoperability among existing products and the emergence of new families of products with different balances between performance and cost.

Given the current European know-how on the area of database-replication, it is possible for Europe to address this problem without being dependent on products conceived and manufactured abroad. The results of GORDA are likely to foster the emergence of a cluster of very competitive products, that will complement each other, and be able to supply the large demand for new large-scale, dependable and efficient information systems in Europe.

Achieved results

In its first year the project concentrated on the definition of the GORDA Architecture and APIs (GAPI). The GAPI consists of a generic architecture and interfaces enabling to accommodate different database engines and replication strategies, and offering a set of generic and expressive programming interfaces easing the efficient interaction of the intervening system components. A first proposal for the GAPI has been documented in detail and is being disseminated to the appropriate forums for discussion.

Scientific results with the proposal and evaluation of database replication protocols, group communication protocols, and autonomic system management have been published.

In the second year of the project the GORDA architecture and programming interfaces (GAPI) have been consolidated and mapped into two major database management systems. Several database replication protocols have been implemented making use of the GAPI and two novel protocols have been proposed. A major effort has been put in the refinement of a simulation infrastructure for evaluation and benchmarking of both communication and database replication protocols, in the implementation of publicly available prototypes of the replication protocols and in the development of one of the industrial partners main commercial product, Sequoia.

In the last year of the project, substantial effort was put into the dissemination and adoption of the GORDA architecture and programming interfaces (GAPI) . A novel replication protocol, using the GAPI in wide-area network has been proposed. As for group communication, the consortium proposed the GORDA Group Communication Service Specification (GCS), as a draft standard and promoted its dissemination and adoption. A major effort, during this period, has been put in the integration of the work produced by the consortium, which resulted in a demonstration of a running prototype of several GORDA replicated databases.

Consortium

The consortium is composed of the University of Minho, Università della Svizzera Italiana, University of Lisbon, INRIA, Continuent Oy and MySQL AB. The GORDA proposal brings together the industry, namely database and middleware providers, and academic partners with strong expertise in the target field. This ensures that the consortium is able to transfer and apply the innovation produced, step by step, from academia to businesses. The consortium will draw on their previous experience to achieve a generic and efficient design as well as to deliver implementations of the various components of the system.

1 Project Objectives and Major Achievements

The first year of the project focused on the definition of the architecture of an open replicated database system able to accommodate different database engines and replication strategies, together with a set of generic and expressive programming interfaces easing the efficient interaction of the intervening components: the database engine, the replication and communication modules, the administration tools and the clients.

To this end, a detailed investigation on existing academic solutions and proposals for database replication together with the study of the current commercial offerings and user requirements was undertaken. The results of this work are on the basis of the proposed, yet preliminary, GORDA Architecture and APIs definition report. A first proposal for mapping the GORDA database API to the database in-core and middleware levels is also available. Complementing the Architecture and API definition, a set of short-term research challenges addressing the architectural aspects of the system, replication protocols, group communication protocols and autonomic system management have been identified.

To support the performance and dependability evaluation of the communication and replication protocols being studied and implemented in the context of the project, a detailed centralized simulation model that combines simulated components of the environment with early implementations of key components has been developed. Extensive testing of both group communication toolkits and known database replication protocols has been conducted and reported. These results will shortly shape the reference implementations that will be offered by the project.

In the second year of its execution, the project focused on consolidating the definition of the architecture of the GORDA Architecture and API and validating its adequacy by mapping the API into a set of representative database management systems and database replication protocols. These goals have been satisfactorily fulfilled and both a cluster and a wide-area oriented platform prototypes built according to the GAPI have been made available. Complementary, a novel programming interface for group communication has been proposed and mappings to the most commonly available group communication toolkits developed and used by the project prototypes. As whole, this represents a set of self-contained software packages, publicly available, that enable the database community to easily experiment and assess the project ideas and results.

Research on databases replication protocols led to the proposal of two new protocols targeting cluster and intercluster environments as well as several assessment and optimization results. Total order multicast protocols was the subject of several publications specially targeting wide area scenarios. Research on autonomic system management emphasized on self-optimization of clustered databases.

Underlying the development of new replication and group communication algorithms and their implementations, a thorough performance and reliability assessment was conducted. This builds on the development of a realistic simulation platform that can run actual implementations in a simulated environment

During the third year of the project, several publications in international symposiums helped to disseminate these and other project results. In this period, the project focused mainly on the implementation of the prototype of the integrated system, which is, by itself, an indicator of the fulfillment of all of the project goals.

The prototype presents the following scenarios

- An in-core replication scenario using a primary-backup replication protocol on the Apache Derby database;
- An hybrid replication scenario using the DBSM protocol on a PostgreSQL database with a self-adaptive cluster, running a *TPC-C light* benchmark;
- A middleware replication scenario using Sequoia on a Mysql database running a TPC-C like benchmark;
- An hybrid replication scenario using the DBSM protocol on a PostgreSQL database, running a TPC-C like benchmark.

which were built on the GAPI and jGCS APIs. Being able to implement such protocols using the proposed APIs fulfills the goal of proposing an Architecture, and of proposing interfaces, promoting the interoperability of DBMSs and replication protocols.

The management tools provided by the supervision console, implemented in the prototype, allow to manage, configure and monitorize the running replication protocols. The prototype also features a middleware replication protocol based on Sequoia, which provides an interim middleware-based solution that allows immediate integration of current DBMSs.

During this period novel replication protocols have been produced and published thus contributing to the community.

Finally, the talks, posters, presentations, and documentation distributed in scientific conferences of the GORDA proposed standards, contributed significantly to the dissemination of the project's results and adoption either by the research community either by the industry.

2 Workpackage Progress

2.1 Workpackage 1 – State of the Art & Requirements Analysis

2.1.1 Objectives

This workpackage main objective was an extensive survey of existing work on database replication proposals that would allow the clear identification of relevant outstanding challenges for both scientific and commercial domains.

The workpackage started in the beginning of current reporting period.

2.1.2 Progress Towards objectives

This workpackage has been concluded. It started in month 1 and finished in month 6. The objectives of the workpackage were entirely met. The work was carried out, as planned, in three complementary tasks that (T1.1) surveyed existing database offerings and exposed the state-of-the-art in replicated databases, (T1.2) surveyed work (mainly from the scientific community) on synchronous replication protocols and group communication protocols, and (T1.3) heavily based on the industrial partners knowledge and their user base, identified a set of use cases that will shape the solutions proposed by the project. Most of these use cases derive from current customers from Continuent. All partners were involved in the workpackage. UMINHO was the lead contractor of the workpackage and, together with USI and INRIA, performed the survey work on replicated databases and synchronous replication protocols. FFCUL led the study on group communication protocols. This work is documented in deliverable D1.1. Continuent was mainly responsible for the identification of the use cases appearing in D1.2. The academic partners identified the main strategic research directions in the context of the project's context and requirements (D1.3).

The workpackage included the organization of the project's Kick-off Meeting (Milestone 1.1). The meeting was held at Universidade do Minho, Braga during the 7th and 8th of October 2004 with the participation of all partners (a representative from MYSQL was available remotely). From The meeting resulted in the first project's report for internal reference (accessible through the project's web site).

2.1.3 Changes in workprogramme

There were no deviations from the planned workprogramme.

2.1.4 List of Deliverables

No.	Deliverable name	Status	Notes
D1.1	State of the Art Report	Accepted	Revision 1.2
D1.2	User Requirements Report	Accepted	Revision 1.3 Submitted
D1.3	Strategic Research Directions Report	Accepted	Revision 1.0

2.1.5 List of milestones

Please refer to Section 2.9.

2.2 Workpackage 2 – Architecture and APIs

2.2.1 Objectives

This workpackage's main goal is twofold: the definition of an architecture for an open replicated database, encompassing self-contained and interchangeable database engines and replication strategy modules, and a generic programming interface enabling the interaction between the replication modules and the other components, namely: the database engine, the communication module, the administration tools, and the database clients.

2.2.2 Progress Towards objectives

This workpackage started in month 7 of the project and lasts for 12 months. FFCUL was the lead contractor for the workpackage with UMINHO, UNISI, INRIA and Continuent as contributors. FFCUL has conducted the development of the communication interfaces. UMINHO has been responsible for the replication interfaces. INRIA worked on the management interfaces. USI provided the database recovery interfaces. Continuent reviewed the proposed API and validated them against their Sequoia based product line.

In the first year of the project, a preliminary definition of the GORDA Architecture and Programming Interfaces (GAPI) was proposed. This corresponds to the accomplishment of Milestone 2.1: "Preliminary architecture definition", and Milestone 2.2: "Preliminary database and clients APIs", of the workpackage.

The design of the GAPI stands on the following general principles: 1) independence of operation and configuration, 2) variable geometry interfaces, i.e., each implementer is free to provide only a subset of the whole GAPI that is adequate for each situation, and 3) façade interfaces that allow manipulation of the internal state of the DBMS without forward and backward format conversions.

The result of the work carried so far is detailed in the "Preliminary Architecture and APIs definition report" (D2.1). The document describes the generic GORDA Architecture, its components and their relationships, and several concrete refinements for different implementation scenarios. It also describes the GORDA Programming Interfaces exhibited by each component and illustrates the usefulness of the GAPI by showing how it can be used to implement various replication protocols.

Roughly, group-based replication protocols require a set of functionalities that are tightly coupled to a transaction processing model, in which a transaction starts, operations are processed through different stages (e.g., parsing, optimization and execution) and after that the transaction commits or aborts. Based on this idea, an architecture is proposed built upon the fact that the operation's results can be gathered at different stages of transaction's processing model. By doing this, the requirements imposed by the different group-based replication protocols implemented at the in core or middleware levels are satisfied.

For each phase, it is suggested the adoption of an API based on the interceptor-reflector pattern that allows the observation and modification of the database. Specifically, the interface exposes transaction processing concepts such as parse trees, write sets or transactions as first class citizens. Thus, replication protocols can register for the notification of relevant state transitions events and call methods to alter the state.

Regarding the architectural specification for database management, it highlights the two main reconfiguration mechanisms: the QoS Manager and the Failure Manager and their generic structure based on sensors, analysis/decision components and actuators. The specified management Interfaces focus on the Cluster Controller Management Interface (CCMI), the Database Management Interface (DBMI) and the sensor interfaces (for QoS and failure sensors). These interfaces outline the behavior of the connecting points between the management system and the rest of the replication infrastructure.

In the second year of the project, the work has been focused on refining the preliminary definition of the GORDA Architecture and Programming Interface (Deliverable D2.1). The results have been reported in deliverables D2.2: "Architecture definition report" which describes the generic GORDA Architecture, its components and their relationships, and several concrete refinements for different implementation scenarios, and D2.3: "APIs definition report" which describes the GORDA Programming Interfaces exhibited by each component and illustrates the usefulness of the API by showing how it can be used to implement various replication protocols. This corresponds to the accomplishment of the third milestone of this workpackage: "Architecture and APIs definition".

2.2.3 Changes in workprogramme

There were no deviations from the planned workprogramme.

2.2.4 List of Deliverables

No.	Deliverable name	Status	Notes
D2.1	Preliminary architecture and APIs Report	Accepted	revision 1.0
D2.2	Architecture Definition Report	Accepted	Revision 1.1
D2.3	APIs Definition Report	Accepted	Revision 1.0

2.2.5 List of milestones

Please refer to Section 2.9.

2.3 Workpackage 3 – Replication Protocols

2.3.1 Objectives

The overall objective of this workpackage is the development of innovative database replication protocols targeting wide-area networks and large clusters settings, as well as hybrid environments interconnecting database clusters over long-distance links. The focus is on strong consistency multi-master protocols based on group communication primitives. The provision of group communication protocols, optimized towards transaction processing is also a major goal of this workpackage.

2.3.2 Progress Towards the Objectives

The workpackage started in month 3 and lasts 33 months. The work on this workpackage has been performed by UNISI, UMINHO and FFCUL.

The workpackage achievements in the first year of the project include 1) a fully usable simulation infrastructure allowing the evaluation of both communication and replication protocols in diverse network, load, and faults scenarios, 2) the prototyping and detailed evaluation of the main group communication based database replication protocols, 3) the evaluation of the main, publicly available, group communication toolkits, 4) a novel optimistic total order broadcast protocol, 5) a new refinement of the Data Base State Machine protocol. The above results correspond to a satisfactory and consistent progress on tasks T3.1, T3.2 and T3.3.

USI is concentrated in clustering protocols (T3.2) and working on database replication system that captures the behavior of legacy database systems and do not require any modifications to the database engine to provide replication. USI is investigating and refining the Database State Machine replication technique (DBSM). A DBSM weakness lies in its dependency on transaction readsets, needed for certification. Extracting readsets usually implies changing the database internals or parsing SQL statements outside the database, both undesirable solutions due to portability, complexity, and performance reasons. The original DBSM has been extended to avoid the need of readsets during certification. The approach has no communication and consistency penalties: termination still relies on a single atomic broadcast and the execution is still serializable. Preliminary results have been published in [32]. Complementary work has been done toward extending current database consistency criteria to replicated databases [30] and understanding alternative approaches targeting high performance data management [31].

UMINHO developed a centralized simulation kernel that eases mixing simulated and real components by automating the accounting of real time. The work extends the Scalable Simulation Framework (SSF) specification for event-driven simulation. The centralized nature of the system allows for global observation of distributed computations with minimal intrusion as well as for control and manipulation of the experiment, namely, to perform fault injection. The developed workbench enabled the testing and evaluation of real implementations of both the replication and communication protocol prototypes exposed to very different network settings and client loads that, without the flexibility and scalability of the tool would otherwise be very hard to achieve. This work has been published in [33].

As fundamental preparatory work for tasks T3.1 and T3.2, UMINHO has implemented a set of database replication protocols representative of those proposed in the literature and carried a thorough evaluation in cluster and wide-area settings. The results of this work have been published in [34].

FFCUL has surveyed existing group communication toolkits to assess their potential use in GORDA. Relevant criteria were: modularity and extensibility (to incorporate the changes required by GORDA), expressiveness of APIs, advertised performance (extracted from publications and technical reports, by the authors or by third-parties), level of support provided. Three toolkits have been selected for experimental evaluation: Appia, JGroups and Spread. An experimental setup has been installed to experiment these protocols. In particular, FFCUL used existing replication engines that were implemented using group communication based replication protocols, ESCADA (UMINHO) and C-JDBC (EMIC). The application benchmarks used to run the performance tests consist of implementations of the Transaction Processing Performance (TPC): TPC-C and TPC-W. Finally, a scheme to log performance data from the execution of the group communication toolkits has been implemented to allow a fine grain analysis of the impact of group communication system in the integrated database replication experimental setup. A document on the detailed analysis of these experiments could not be anticipated to the current report but will be available soon.

Early results have shown that Appia (FFCUL) appears to be the best choice for the project. Therefore, FFCUL spent already a great deal of effort in implementing multiple optimizations to the Appia system to improve its performance. Among these improvements we list kernel optimizations, new protocols to maintain the membership and improvements on some existing protocols that help providing the group communication.

FFCUL has yet designed and implemented a novel total order broadcast that combines, in a single protocol, different strategies to provide optimistic total order delivery. Its description and evaluation has been submitted for publication [36].

During the second year of the project, the work has been carried on 1) group communication protocols and toolkit, 2) cluster and wide-area database replication protocols, and 3) the evolving simulation infrastructure that allows the evaluation of both communication and replication protocols in diverse network, load, and faults scenarios.

FFCUL worked on the Appia protocol composition framework, a baseline for the group communication protocols that supports the project replication engines. FCCUL on total order protocols and adaptive switching between several implementations of total order. This improves the system performance by using the implementation of optimistic total order that is more appropriate to the network and traffic conditions. The results of this work have been published in [37, 38, 39, 41].

FFCUL and UMINHO proposed a novel group communication API – jGCS [40]. The fifth release of jGCS is available at http://jgcs.sf.net.

With deliverables D3.1 and D3.2, “Wide-area oriented protocols report” and “Cluster oriented protocols report” respectively, the consortium shaped the replication protocols to be implemented as references. A large body of work has been put during the reporting period on the implementation of theses protocols. Deliverable D3.3: “Replication module reference implementation” contains these protocols implemented (as well as other simpler ones) and has been made publicly available.

UNISI proposed the Multiversion Database State Machine (vDBSM). vDBSM is an extension of the DBSM, a kernel-based optimistic replication protocol, placed at the middleware layer and still providing strong consistency, i.e., one-copy serializability.

Middleware-based replication protocols match the semantics of standard database access interface (e.g. ODBC or JDBC), which makes it straightforward to migrate from centralized to replicated environment. However, standard database interfaces do not provide fine grained information from the database engine, such as, accessed data items per transaction or information on when transaction begins its execution, when an insert, update, delete or select takes place and when a transaction finishes.

UMINHO and UNISI proposed WICE [3] a pragmatic database replication protocol based on group communication that targets interconnected clusters.

In contrast with previous proposals, it uses a separate multicast group for each cluster and thus does not impose any additional requirements on group communication, easing implementation and deployment in a real setting. The protocol ensures one-copy equivalence while allowing all sites to execute update transactions.

UMINHO developed the ESCADA Toolkit which provides a set of classes and interfaces to easily build different replication protocols upon the GAPI and a communication layer for the same toolkit to use with different replication protocols. This layer is built on the jGCS augmenting it and provides multiplexing and de-multiplexing of messages, using channels.

UMINHO also started the development of recovery protocols for the ESCADA Toolkit: transfer of the entire database using PostgreSQL tools; transfer the missed updates to joining replicas using middleware logging; transfer the last version of changed objects using middleware logging. Most of these implementations have been readily available through the project's website.

Continuent focused on the development of the core of its database replication offering – Sequoia. Version 3.0 of Sequoia has just been made publicly available through Continuent.org website.

UMINHO continued the development and use of its centralized simulation kernel that eases mixing simulated and real components by automating the accounting of real time. During the current reporting period work has been focused on studying performance and reliability tradeoffs in available protocols.

In the final one and half year of the project, the work was carried out on 1) group communication protocols and toolkit, 2) cluster and wide-area database replication protocols, and 3) the evolving simulation infrastructure that allows for evaluation of both communication and replication protocols in diverse network, load, and fault scenarios.

UMINHO worked on recovery protocols, i.e. protocols that allow a replica to evolve its state in order to become synchronized with the other replicas. This work was conducted on the ESCADA Replication Server, and reported in Deliverable D3.3: "Replication modules reference implementation".

UMINHO also worked on the implementation of replication protocols on the ESCADA Replication Server. These protocols provide primary-backup, state-machine, conservative and certification-based replication, demonstrating the flexibility of the ESCADA Replication Server. This work is reported in Deliverable D3.3: "Replication modules reference implementation".

UMINHO worked on the integration and configuration of the replication framework which consists of the ESCADA Replication Server and the Appia group communication toolkit. These must cooperate in order to boot the replicated database and during its runtime. To solve dependency issues arising from the components, the Spring Framework has been adopted. This work is reported in Deliverable D3.4: ”Modules description and configuration guide”.

UMINHO worked on the description of the configuration steps needed to setup different replication protocols on different databases. This work is reported in Deliverable D3.4: ”Modules description and configuration guide”.

FCUL worked on the implementation and improvement of the Statistically Estimated Total Order (SETO) protocol, to be used in the GORDA integrated prototype. This protocol provides uniform message delivery with optimistic notifications that can be used by the replication protocols to improve overall system performance. Notifications are conveyed by means of jGCS Services defined in WP2. FCUL also finished and tested the implementation of the Total Order Switching protocol, which can be used by GORDA on unstable environments. This work is reported in Deliverable D3.5: ”Group communication protocols report”.

FCUL implemented a Primary View protocol that defines the notion of a primary partition in a group of replicas. This allows a subset of the system to continue to move forward, even if several nodes become unavailable due to a network partition, and keeps the state of the replicas from diverging. This protocol also includes a mechanism to notify a replica it has lost some transactions and needs to be recovered. Finally, FCUL worked in the Appia group communication toolkit to improve its stability and performance. This work is also reported in Deliverable D3.5: ”Group communication protocols report”.

FCUL analysed how to reuse the system and knowledge produced in the GORDA project in emergent systems such as Software Transactional Memory (STM). This kind of systems is starting to be used in production environments and GORDA should accommodate new needs. Namely, we studied the case of the FénixEDU system, a web application server that supports the academic activities of a Portuguese University (IST/UTL). The system is beginning to show severe scalability and performance problems and would benefit from a database replication solution. More than simply applying the GORDA prototype to FénixEDU, the replication protocols produced in this workpackage can be adapted to do replication in transactional memory systems, as a way to exploit the project results. This analysis is reported in the paper [17].

Finally, FCUL worked on a new database replication protocol that offers strong consistency (linearizable semantics) and allows reads and non-conflicting writes to execute in parallel in multiple replicas. This new protocol supports the use of quorums to trade the availability/efficiency of read and write operations, building a bridge between consensus-based and quorum based solutions for database replication. This work was accepted for publication and is reported in the paper [18].

UNISI proposed and implemented conflict-aware load balancing techniques for database replication. While load balancing, in general, is not a replication protocol in itself, the techniques developed are important because they show how the performance of a DBSM-like protocol can be improved by taking concurrency control issues into account when scheduling transactions for execution. The result of this work was published in the proceedings of the 23rd Annual ACM Symposium on Applied Computing (ACM SAC 2008) [11].

UNISI proposed and implemented BaseCON, a novel fault-tolerant replication protocol that takes advantage of workload characterization techniques to increase the parallelism in the system. This work is planned to be submitted to a conference.

UNISI investigated correctness criteria for fault-tolerant, middleware-based database replication. Besides one-copy serializability, two stronger properties were considered: session consistency and strict serializability. In addition to studying these aspects abstractly, their costs in the context of BaseCON were also evaluated.

UNISI has been investigating recovery on middleware-based database replication protocols. This effort has involved researchers from other projects. Some of the results obtained so far will appear in the proceedings of the 7th International Symposium on Parallel and Distributed Computing (ISPDC 2008) [8].

2.3.3 Changes in the Workprogramme

Justified by the need of readjusting part of the three partners human resources, the workpackage started 3 months in advance. That is, instead of month 6, it started on month 3. The work carried during the first three months was not dependent on the preceding workpackage WP2. UMINHO and FFCUL concentrated on the development of the protocols' evaluation testbeds while UNISI worked on the refinement of the Data Base State Machine.

The workpackage was extended by six months. The granted extension allowed the workpackage participants to complete the specified deliverables.

2.3.4 List of Deliverables

No.	Deliverable name	Status	Notes
D3.1	Wide-area Protocols Report	Accepted	New revision
D3.2	Cluster Oriented Protocols Report	Accepted	New revision
D3.3	Replication Modules Reference Implementation	Accepted	New revision
D3.4	Modules Description and Configuration Guide	Accepted	
D3.5	Group Communication Protocols Report	Reviewed	New report

2.3.5 List of Milestones

Please refer to Section 2.9.

2.4 Workpackage 4 – Database Support

2.4.1 Objectives

This workpackage aims at providing reference implementations of the GORDA database API: a full implementation within an open-source database engine, and a middleware-level implementation.

2.4.2 Progress Towards the Objectives

This workpackage started in month 7 and lasts 24 months. The lead contractor for the workpackage is Continuent. Major contributors to this package were UMINHO, INRIA and Continuent. UMINHO defined the GAPI mappings and implemented the current in-core mapping in PostgreSQL. INRIA collaborated in the definition of the GAPI middleware mapping. Continuent started the implementation in MySQL.

The work is organized in two tasks that run in close coordination but in parallel. The first, is responsible for the “In-core mapping and implementation” of the GORDA database API (T4.1). Its major outcome will be the in-core GAPI implementation in the MySQL database and a proof-of-concept implementation in PostgreSQL. The second regards the “Middleware mapping and implementation” (T4.2) with an expected reference implementation and a hybrid proof-of-concept based on the HSQLdb database.

Along with the preliminary definition of the GAPI, in this workpackage were proposed two (still preliminary too) mappings of GAPI, a in-core mapping and a middleware mapping giving rise to two reports (D4.1 and D4.2).

Both tasks have a clear dependency on the results of WP2 (Architecture and APIs) and because the definition of the GAPI is preliminary at this stage,

In the in-core approach the database must provide a set of functionalities in order to fully exploit the group-based replication protocols. Specifically, according to the transaction's processing model, it must inform the replication engine when a transaction begins, commits or aborts and when an operation walks through the different stages of the transaction's processing model.

The events directly related to a transaction, namely the transaction's begin, commit and abort are handled by triggers. In contrast to the traditional triggers, in the current proposal they are not associated to a relation. Surprisingly, the efforts to build such triggers are not so high because most DBMSs have non-standard APIs that provide similar functionalities. Our current prototype does not handle all the stages throughout an operation passes. Currently, it only deals with events that produce read and write sets and with group-based replication protocols that use these sets upon transaction commit. So, there is no notifications or special treatment for those events. If it was required a solution based on triggers would also be developed.

To gather the write set, we used to different approaches. In the first approach, this set is updated and maintained by using the traditional triggers. Although it is a simple solution, the triggers may turn the replication process infeasible. The overhead imposed by the additional inserts to store the operations' result as well selects to retrieve them is not negligible. In the second approach, we use the same solution that gathers the read set.

To do that, a set of intrusive modifications were made to the database engine, since there is no notion about read data as required by the group based protocols. Specifically, we augmented the DBMSs by using special iterators (roughly, an iterator is a pattern used by the database developers to process SQL statements) to gather and copy the read data into an in-memory relation. Basically, we call this approach light-weight triggers as it is implemented by using low level interfaces provided by the database engines and does not require to fire up a high level procedure for each tuple accessed by the operations.

In the second year of the project, the work was exclusively dedicated to the in-core implementations of the GORDA API in PostgreSQL and Apache Derby. This resulted into two prototypes of GORDA compliant PostgreSQL and Apache Derby databases reported in deliverables D4.5: "In-core proof- of-concept for the PostgreSQL" and D4.6: "Hybrid proof-of-concept for Apache Derby".

UMINHO implemented GAPI in PostgreSQL. The prototype is not yet fully compliant although the third and current release (0.3) has improved it substantially. This implementation consists of four modules. The first module extends the PostgreSQL and provides triggers on a connection startup and shutdown and on a transaction begin, commit and rollback. The second module enables to define high priority transactions. The third module is responsible for serializing information from previous triggers along with write set data. The fourth module is a set of classes in Java that directly renders the GORDA API and is built on the information serialized from the PostgreSQL.

FFCUL led the implementation of the hybrid proof-of-concept of GAPI in Apache Derby with the close collaboration of UMINHO. This implementation exploits the fact that Derby is written in Java. Most of the rendering consists on wrappers of already existing objects, thus reducing performance problems when providing reflection. Except for the request context and some phases defined by the GAPI (eg., parser, optimizer and logger), the prototype is fully operational.

In the last one and half year of the project, the work was exclusively dedicated to the implementation of the GORDA API (GAPI) in PostgreSQL, MySQL, Apache Derby and Sequoia. This resulted in three prototypes of GORDA compliant PostgreSQL, Apache Derby and MySQL databases. It resulted, also, in a middleware-level implementation prototype based on Sequoia. The PostgreSQL, Apache Derby and Sequoia prototypes are reported in Deliverables D4.3: "In-Core Proof-of-concept", D4.4: "Middleware Proof-of-concept" and D4.5: "Hybrid Proof-of-concept" and D4.6: "DB Support Description and Configuration Guide". Deliverables D4.1: "In-Core Mapping Report" and D4.2: "Middleware Mapping Report" were superseded by Deliverables D4.3 and D4.4 respectively.

UMINHO implemented the GAPI in PostgreSQL. This implementation consists of four modules. The first module extends PostgreSQL and provides triggers on connection startup and shutdown and on transaction begin, commit and rollback. The second module enables the definition of high priority transactions. The third module is responsible for serializing information from previous triggers along with write set data. The fourth module is a set of classes, in Java, that render the GAPI directly and is built on the information serialized from PostgreSQL. The current release (0.4) was substantially improved, regarding bug fixes and new features. New features include integration with the Spring and Maven frameworks and with the ESCADA Replication Server through dependency injection. This work is reported in Deliverable D4.5: "Hybrid Proof-of-concept"

FCUL led the implementation of the in-core proof-of-concept of GAPI in Apache Derby with the close collaboration of UMINHO. This implementation exploits the fact that Derby is written in Java. Most of the rendering consists of wrappers of already existing objects, thus reducing some performance problems when providing reflection. In this reporting period, FCUL upgraded the existing implementation of the GAPI to Apache Derby version 10.2.2.0 and GAPI version 0.4. This new version was also integrated with the Spring framework to support dependency injection. The configuration of the GAPI and the parts that should be reflected were also changed from a static to a dynamic configuration, allowing parts of the GAPI to be enabled/disabled in runtime. UMINHO contributed to the in-core implementation of the GAPI with priority transactions, which allow transactions from remote replicas to always be committed, extraction of parsed statements, and the implementation of a log miner to allow database recovery. This work is reported in Deliverable D4.3: "In-Core Proof-of-concept".

UMINHO implemented parts of the GAPI in MySQL. This implementation includes the context reflection interfaces (database, connection and transaction contexts) and the extraction of the write set of update transactions. This provides integration with the ESCADA Replication Server and therefore the ability to replicate MySQL databases using the GORDA framework.

UMINHO also contributed to the middleware-based implementation of the GAPI inside Sequoia. Sequoia is a middleware database wrapper that intercepts client connections to enable replication. Clients connect to a Sequoia server and the DBMS becomes a backend of this server. UMINHO implemented a subset of the GAPI in the Sequoia server. This implementation includes the GAPI contexts, extraction of the write set of update transactions for MySQL and PostgreSQL backends, the reflection of the parsed statement and priority transactions. This work is reported in Deliverable D4.4: "Middleware Proof-of-concept".

UMINHO studied Write-ahead Log algorithms and corresponding variants of the ARIES protocol for concurrency, implemented in databases such as PostgreSQL, Oracle, MSQL Server and DB2. This study aimed at developing guidelines for an efficient read-set extraction protocol. During this period UMINHO started to develop a prototype for read-set extraction on PostgreSQL, showing the chosen approach is viable even when considering the amount of changes needed on PostgreSQL. This work is planned to be submitted to a conference.

Finally, FCUL and UMINHO worked together on the analysis of a general purpose DBMS reflection architecture, supporting multiple extensions while, at the same time, admitting efficient implementations. The analysis illustrates the usefulness of this approach with concrete examples and is reported in the paper [16].

Continuent's contribution to this work package encompasses the following activities:

Sequoia Code Stabilization Continuent has done an enormous amount of work to stabilize and evolve Sequoia in order to allow it to support the GORDA/M model. This has included over 100 bug fixes, improvements, and new feature additions to the Sequoia core. The main obstacle to using Sequoia effectively with GORDA is not the implementation of GORDA/M APIs, which represent a relatively small portion of the code base, but instead the instability of Sequoia itself, originally developed as a research effort and featuring, still, a number of quite immature components.

The stabilization effort has focused on the following areas, among others:

Transaction ordering Ordering SQL requests is difficult to implement and involves subtle problems with concurrency. A number of bug fixes have been done in this area;

Virtual database management Improving the robustness of management procedures to bring virtual databases on-/off-line and to ensure failover conditions are properly handled;

Fixes to the recovery log The recovery log component stores updates for replay and records backend database state. This component underwent a major refit to eliminate problems with concurrency that caused later recovery failures.

These and other fixes have resulted in a cleaned-up Sequoia codeline, that forms the basis of Sequoia 4.0. (See below.)

Hosting Open Source Code This includes hosting of the open source code in CVS/SVN repositories, JIRA bug tracking, Wikis, mailing lists, and the open source codelines. 1.5 man-months were consumed by migration of the open source projects from the Grenoble site to permanent hosting facilities at LogicWorks in New York City. This location was chosen due to the high level of support offerred as well as the excellent connectivity to sites world-wide and especially to Europe.

Myosotis Connector Development The myosotis Connector provides transparent connections from native clients to Sequoia. This significantly extends the utility of GORDA/M by allowing non-Java applications to connect to clusters and came as a consequence of additional work done with users on key requirements for clustering. Connector work, during this phase of GORDA, has focused on implementing support for PostgreSQL as well as prepared statement support for both PostgreSQL and MySQL. A number of issues related to data types have also been successfully resolved. At the end of this phase, Connector support for these two databases is excellent.

Planning for Launch of Sequoia 4.0 Sequoia 4.0 is the next revision of Sequoia, designed to support GAPI for the long-term. By the end of the current work phase, Continuent had completed plans with Emmanuel Cecchet, the original author of Sequoia, to use the stabilized Sequoia code, maintained by Continuent, as the basis for a completely new release of Sequoia. The Sequoia 4.0 codeline will include a large number of feature changes, including the following:

- Removal of dead code, refactoring of the remaining code, and clean-up of configuration functions;
- Pluggable recovery log implementation for database-specific preconfigured implementations;
- Many usability improvements;

The code clean-up is necessary to allow long-term support of GAPI interfaces, which depend on clean support for session and request management. The GAPI interfaces will be partially re-implemented to enable long-term support.

A partial re-implementation of the GAPI interfaces, along with code clean-up, as GAPI depends on a clean support for session and request management, are necessary for the GAPI to be supported int the long term.

2.4.3 Changes in the Workprogramme

There were no deviations from the planned workprogramme. The leadership of the workpackage has changed however due to internal arrangements on INRIA's and Continuent's teams. INRIA became the leader contractor of WP5: "Integration, Validation and Benchmarking". The effort of both partners has been adjusted accordingly, without impact on each workpackage total manpower. Continuent performed work corresponding to two man-month.

The work on the GAPI implementation in the MySQL database and in the middleware wrapper, both led my Continuent, has been delayed. A running prototype of the latter, based on Sequoia, is expected soon (D4.4).

The in-core implementation of the GAPI in the MySQL DBMS was developed by UMINHO.

2.4.4 List of Deliverables

No.	Deliverable name	Status	Notes
D4.1	In-Core Mapping Report	Accepted	Superseded by D4.3
D4.2	Middleware Mapping Report	Accepted	Superseded by D4.4
D4.3	In-Core Proof-of-concept -	Accepted	
D4.4	Middleware Proof-of-concept	Submitted	Pending evaluation
D4.5	Hybrid Proof-of-concept	Accepted	
D4.6	DB Support Description and Configuration Guide	Accepted	

2.4.5 List of Milestones

Please refer to Section 2.9.

2.5 Workpackage 5 – Integration, Validation and Benchmarking

2.5.1 Objectives

This workpackage integrates the results from the previous workpackages, in order to provide a running prototype. This includes the development of tools to configure and manage replicated databases.

This workpackage also validates and benchmarks the implementations of the GORDA integrated prototype. The emphasis is on realistic and industry standard benchmarks, such as those proposed by the Transaction Processing Council (TPC).

This workpackage provides a comprehensive testing and benchmarking environment for replication algorithms, implementations and the API. This should allow for both a fine grained evaluation of system components as well as of overall system performance, and still, to address performance and reliability issues.

2.5.2 Progress Towards the Objectives

This workpackage started on month 18 of the project and ended on month 42.

During the first months of the workpackage work has been devoted to create the needed infrastructure and "glue" to implement both communication and the database replication protocols in the simulation framework and as real prototypes and to evaluate and benchmark early implementations of the various modules. Complementary, the implementation of a component-based management architecture for managing replicated databases, has started.

Integration Development of a centralized simulation library - MinhaSSF, that provides transparent means of injecting real implementations prototypes into simulation, combining them with abstract simulation models (UMINHO).

FFCUL mapped the most commonly used group communication toolkits: Spread, Jgroups and Appia into the jGCS API this way providing a uniform interface to using these toolkits by the project replication protocols.

Integration of Appia and jGCS group communication implementations in MinhaSSF centralized simulation library (UMINHO, FFCUL). This required some refactoring of Appia and development of the MinhaLib support library.

Development of a component-based management architecture managing secure and heterogeneous replicated databases, typically in cluster-like environment. This includes in particular failure manager able to support automated repair plans, following the occurrence of hardware or software failures; a QoS/Performance Manager, able to maintain the performance of a multi-tier system within a performance range, despite large variations in request load, subject to the availability of sufficient resources (INRIA).

Benchmarking Evaluation and benchmarking of jGCS with Appia bindings group communication implementations in MinhaSSF centralized simulation library (UMINHO).

Testing and evaluation of the replication algorithms (WP3), which resulted in the "Algorithms performance and reliability assessment report" (D5.1).

During the final one and last year of the project, several tasks involving the collaboration of the teams of each partner were executed in order to implement an integrated prototype of the GORDA system. These tasks involved:

- The GORDA Replication Service (GRS)
  - It is based on the ESCADA Replication Server.
  - It includes several replication protocols (e.g. P/B, DBSM, WICE).
- The GORDA Communication Service (GCS)
  - It is based on the Appia group communication system.
  - It includes several total ordering protocols (TO, SETO)
- The GORDA Management Service (GMS)
  - It is based on the JADE autonomic management system.
  - It builds upon software technologies such as OSGi, JMX and Fractal/Julia.

FCUL worked with UMINHO on the integration and validation of the Appia communication framework with the ESCADA Replication Server and the management tools. Appia interacts with ESCADA using the jGCS interface defined in WP2 and is dynamically configured using Injection of Control. The Appia toolkit also interacts with the management tools using JMX. Several actions (such as bootstrapping the group) and variables (e.g. the current view and the throughput of messages) are exported by Appia using JMX and can be managed using any JMX compliant client. Several bugs where found and solved in this integration process. The integrated system was benchmarked using the TPC-W and TPC-B benchmarks. This work is reported on Deliverables D5.2: " Prototype of the integrated system", D5.3: "Interface and modules performance assessment report" and D5.4: "Management tools set".

The protocols produced for total ordering where benchmarked by the FCUL team and the results are reported in the papers that describe the protocols. This work is reported in Deliverable D5.1: "Algorithms performance and reliability assessment report".

Continuent developed and published the Gorda/Sequoia testing tools as the Bristlecone testing toolkit. Bristlecone was released as open source, and can be downloaded from: http://bristlecone.continuent.org./Applications/TextMate.app/Cont
warning: regexp has invalid interval Bristlecone provides tools for database performance testing . There are two main tools included in the package:

- The Evaluator generates mixed loads of inserts, updates, deletes and selects. Output can be built-in graphics, HTML, XML, or CSV.

- The Benchmark runs performance test cases with systematically varying parameters. Output can be HTML or CSV.

The Bristlecone tools were designed for comparative evaluation of database clusters. They include built-in support for threading, the ability to generate new tests quickly with simple configuration file changes, and the ability to do systematic tests across different database implementations. Bristlecone test tools include so-called 'scale-out' benchmarks that can be used to test a wide variety of clusters including those based on the three general replication approaches supported by GORDA. This work is reported in Deliverable D5.2: "Prototype of the integrated system".

UMINHO worked with INRIA on the GORDA management tools. These include several modules, among which the Configuration Module, the Monitoring Module, the Deployment Module, the Automatic Recovery Module, the Dynamic Optimization Module and the Supervision Console. This work is reported in Deliverable D5.4: "Management tools set".

The Configuration Module aims at setting up, deploying and configuring the replicated databases, such as by specifying the number of database replicas, or database IP addresses and port numbers. This is performed through an XML-based ADL (Architecture and configuration Description Language), specialized for configuring and deploying clusters of replicated databases (INRIA).

The Monitoring Module's objectives are to collect statistics about the behavior of replicated databases, and to make those statistics available to the administrator/user. Statistics exhibited may be low-level (e.g. cpu, memory usage), middleware-level (e.g. statistics about GORDA internal services), or higher-level (e.g. transaction abort rate). The Monitoring Module was implemented using JMX-based beans to monitor system behavior (FCUL, INRIA, UMINHO).

The Deployment Module aims at providing online deployment of GORDA components. It integrates, on the one hand, GMS - the Jade deployment service, built on top of OSGi, and on the other hand, GRS/GCS - the replica/group member addition features as handled by ESCADA and Appia (FCUL, INRIA, UMINHO). It also includes JadeBoot, an OSGi bundle that allows the OSGi framework to be managed through JMX, thus allowing other modules to interact with it, namely by getting its state (log files, running bundles) and deploying/undeploying bundles. This work was been published in the paper [1].

The Automatic Recovery Module's objective is to perform automatic replica failure detection and automatic replica addition to replace a failing database replica. in the form of an autonomic manager implementing a control loop which integrates, on the one hand, the Monitoring Module, and on the other hand, the Deployment Module (INRIA).

The Dynamic Optimization Module aims at performing online redeployment of database replicas to adjust resource usage to load variations, and thus optimize (minimize) resource usage while providing QoS guaranties. This module takes the form of an autonomic manager implementing a control loop that integrates: (i) the monitoring module, (ii) the deployment module, (iii) heuristics-based algorithms to calculate the nearly optimal amount of cluster resources as a function of database cluster load, and possibly (iv) model-based algorithms to calculate the optimal configuration of a database cluster (INRIA).

The Supervision Console aims at reflecting the monitored state of the database cluster, and providing the means to perform manual (re-)configuration of the database cluster. The console makes use of the jManage open source JMX console; it integrates jManage with the JMX-based Monitoring Module, and defines new dashboards related to GORDA components (UMINHO).

2.5.3 Changes in workprogramme

There were no deviations from the planned work program.

2.5.4 List of Deliverables

No.	Deliverable name	Status	Notes
D5.1	Algorithms Performance and Reliability Assessment Report	Accepted	
D5.2	Prototype of the Integrated System	Accepted	Companion report submitted.
D5.3	Interface and Modules Peformance Assessment Report	Submitted	Pending evaluation.
D5.4	Management Tools Set	Submitted	
D5.5	Deployment Guides For Replication Strategies		

2.5.5 List of Milestones

Please refer to Section 2.9.

2.6 Workpackage 6 – Dissemination and Standardization

2.6.1 Objectives

The goal of this workpackage is the dissemination of information on the project's activities and results as well as the promotion of the project's impact on the communities of interest. Furthermore, an effort will be carried out to achieve a standard for interoperable database replication protocols based on GORDA.

This is a ongoing workpackage started in the beginning of the project.

2.6.2 Progress Towards the Objectives

This workpackage runs through all the project's lifetime.

The workpackage is divided in two taskqs: one devoted to the "Project publicity and support" (T6.1) and the other to "Standardization initiatives" (T6.2). In both tasks a great deal of effort has been put by the project partners.

A web site with a rich information structure and workflow control has been implemented fulfilling two of the workpackage deliverables (D6.1: "Web site structure and front page" and D6.2: "Project presentation - Web site").

UMINHO, with the contribution of the other partners, organized a workshop in conjunction with 2005 edition of the Very Large Data Bases conference (one of major conferences in the area). The workshop was entitled "Design, Implementation and Deployment of Replicated Databases", consisted of 8 presentations, a discussion panel, and counted with 22 participants. The discussion panel consisted of Prof. Patrick Valduriez (University of Nantes), Nimar Arora (Oracle), Emmanuel Cecchet (Emic Networks), Prof. José Pereira (University of Minho) and was moderated by Prof. Ricardo Jimenez-Peris (Polytechnic University of Madrid). The proceedings of the workshop correspond to deliverable D6.3: "Workshop proceedings".

All academic partners published scientific papers in the context of the project in international symposiums such DSN, SRDS, LADC, and the project's VLDB workshop. These publications are listed in Annex 5 and the papers appended to this report. An initiative worth to mention was the tutorial given by Prof. Luís Rodrigues at the International Conference on Dependable Systems and Networks 2005 where database replication played a central role and the GORDA project was duly advertised.

On the standardization front, small but firm steps were taken. All the work that is being done in the implementation of the GAPI, both in MySQL and PostgreSQL, is being advertised and feedback requested in the main forums of these two databases. The aim is to have these implementations as small, modular changes that can be easily applied and maintained so that they get into the main development trees. Examples of this are the PostgreSQL patches currently available at the project's site.

Effort has also been applied in supporting open source releases of prototypes. Namely, the combined usage of Appia and Sequoia has resulted in a large user feedback.

Continuent advertised the middleware-based active replication approach used in GORDA at ApacheCon US 2005, PostgreSQL user conference, ApacheCon Europe 2006 in Dublin, ApacheCon Asia in 2006 and MySQL User conference to collect feedback.

The consortium promoted technical meetings with two industrial partners:

- SUN Trondheim
- Telbit

from which some results are expected in the near future.

The consortium participated in PostgreSQL and MySQL conferences in order to promote GORDA.

- PGCon 2007 The PostgreSQL Conference
- MySQL User Group 2007
- PGDay-IT Conference
- MySQL Conference and Expo 2006

The consortium established contacts with standardization entities such as:

- COPRAS
- OpenGroup
- SAForum

The consortium also established contacts with and participated in several development communities such as:

- PostgreSQL http://www.postgresql.org/community/lists/
- Apache Derby http://db.apache.org/derby/derby_mail.html
- Appia https://forge.continuent.org/mail/?group_id=11
- jGCS
- Sequoia https://forge.continuent.org/mail/?group_id=6

UMINHO presented seven invited talks at universities and research labs:

- Hong-Kong Baptist University (Jianliang Xu)
- CWI (Martin Kersten)
- SUN Trondheim (Oystein Grovlen and Maitrayi Sabaratnam)
- Tokyo Institute of Technology (Haruo Yokota)
- Peking University (Jinyu Zhang)
- Universidade Federal da Bahia (Raimundo Macêdo)
- Universidad Pública de Navarra (José González de Mendívil)

The consortium organized scientific events:

- Dependable and Adaptive Distributed Systems (DADS) track at ACM Symposium on Applied Computing SAC'2006. In joint collaboration with two EU FP6 IST projects, Dedisys (http://www.dedisys.org) and Rodin (http://rodin.cs.ncl.ac.uk/).
- Dependable and Adaptive Distributed Systems (DADS) track at ACM Symposium on Applied Computing SAC'2007. In joint collaboration with two EU FP6 IST projects, Dedisys (http://www.dedisys.org) and Rodin (http://rodin.cs.ncl.ac.uk/).
- Software Dependability Workshop at SAFECOMP 2007.

The consortium distributed project's posters and deliverables at the conferences NCA'07, DSN'07, SAFECOMP'07, SRDS'07, DOA'07, ADSS'07.

UMINHO participated in EuroSys 2007, presenting two posters to disseminate the work being done in the scope of GORDA.

UNISI co-organized a seminar about replication techniques, entitled "A 30-year perspective on replication", in which about 60 participants from all over the world participated. Together with UNISI, members of UMINHO and FCUL also took part in the seminar, which was by invitation only. Luis Rodrigues from FCUL contributed with a talk. As a result of this seminar, a book is planned to be edited. GORDA participants are expected to contribute with two chapters.

FCUL participated in the workshop “30-year perspective on replication” with the presentation “From Quorum to Consensus-based Replication” to disseminate new work that was done in the last reporting period to the replication community.

FCUL participated in Euro-Par 2006 with a presentation entitled “Run-Time Switching Between Total Order Algorithms” to disseminate work previously done in the scope of GORDA.

FCUL participated in the 12th IEEE International Symposium Pacific Rim Dependable Computing (PRDC’06) with a presentation entitled “On Statistically Estimated Optimistic Delivery in Wide-Area Total Order Protocols” to disseminate work previously done in the scope of GORDA.

FCUL participated in the Large-Scale Distributed Systems and Middleware (LADIS) workshop with the presentation “Adaptive Optimistic Total-Order Protocols for Wide-Area Database Replication” to disseminate the work previously done in the scope of GORDA.

FCUL and UMINHO participated in the SAFECOMP 2007 conference to show a demonstration of the GORDA prototype.

FCUL and UMINHO gave a tutorial on “Database Replication and Clustering” on the 23rd Annual ACM Symposium on Applied Computing.

FCUL participated in the 2nd Workshop on Dependable Distributed Data Management (WDDDM’08) to present the work entitled “Versioned Transactional Shared Memory for the FénixEDU Web Application” to disseminate new work, done during the last reporting period, that explores ways of exploiting GORDA results in new projects.

INRIA carried out several initiatives to present and demonstrate GORDA autonomic management results, such as at the INRIA’s 40 years anniversary celebration in December 2007 (http://www.inria.fr/40ans/), the ”Forum 4i” in April 2007 (http://www.forum4i.fr/), and at ”Fête de la Science” in 2006 (http://www.fetedelascience.fr/).

Continuent designed Tungsten and collaborated with MySQL and PostgreSQL to publicize and push the GAPI:

1. Papers describing Tungsten architecture explicitly refer GORDA. A a preview of the technology has been presented at the most recent MySQL User Conference. The following link describes the talk. The next link shows the presentation, which specifically mentions GORDA pluggable replication. The overall design is based on notions from GORDA.

This is the first public review of the architecture, which has been in development internally since May 2007 and has been circulated to a number of customers.

2. Papers describing test technology for generic scale-out. Continuent has presented testing concepts based on GORDA ideas as well as code developed as part of GORDA (Bristlecone) at both MySQL and PostgreSQL conferences. Links to the papers are shown below:

    - http://en.oreilly.com/mysql2008/public/schedule/detail/2582
    - http://assets.en.oreilly.com/1/event/2/Continuent%20Tungsten_%20Proxies%20on%20Steroids%20for%20HA%20and%20Performance!%20Presentation.pdf

3. Continunet is working with members of both the PostgreSQL and MySQL communities to push replication approaches based on APIs from GORDA. Most of this work is informal and hence not documented. However, the notion of using GAPIs was quite favorably received at the PG-East conference in March 2008.

4. Planning and roadmap work for Sequoia 4.0, which will include supported versions of GORDA APIs. Please refer to http://sequoia.continuent.org.

2.6.3 Changes in the Workprogramme

There were no deviations from the planned workprogramme.

2.6.4 List of Deliverables

No.	Deliverable name	Status	Notes
D6.1	Web Site Structure and Front Page	Accepted	
D6.2	Web Site	Accepted	
D6.3	Workshop Proceedings	Accepted	
D6.4	Draft Standard- GAPI	Accepted	
D6.4b	Draft Standard - jGCS (NEW)	Accepted	
D6.5	Final Dissemination and Exploitaion Plan	Resubmited.	
D6.6	Report on Raising Public Participation and Awareness		To be replaced by "Publishable Final Activity Report"

2.6.5 List of Milestones

Please refer to Section 2.9.

2.7 Workpackage 7 - Demonstration

2.7.1 Objectives

This work package aims at deploying a prototype implementation of the GORDA architecture in the context of a realistic information system. Selection of the target is done just-in-time to ensure that the demonstration scenario is demand driven and thus maximizes the dissemination of project results and the impact of GORDA technology.

2.7.2 Progress Towards the Objectives

This workpackage started on month 27 of the project and ended on month 42.

UMINHO and FCUL worked together in the setup and deployment of the demo scenarios that were presented in the in the October 2007 review meeting, and in the SAFECOMP 2007 conference.

The replication scenarios presented in the demo are:

- An in-core replication scenario using a primary-backup replication protocol on the Apache Derby database.
- An hybrid replication scenario using the DBSM protocol on a PostgreSQL database with self-adaptive cluster, running a *TPC-C light* benchmark.
- A middleware replication scenario using Sequoia on a MySQL database running a TPC-C like benchmark
- An hybrid replication scenario using the DBSM protocol on a PostgreSQL database, running a TPC-C like benchmark.

2.7.3 Changes in the Workprogramme

The workpackage was extended by six months.

2.7.4 List of Deliverables

No.	Deliverable name	Status	Notes
D7.1	Deployment Plan	Accepted	
D7.2	Demonstration	Accepted	
D7.3	Evaluation Report		Due 42

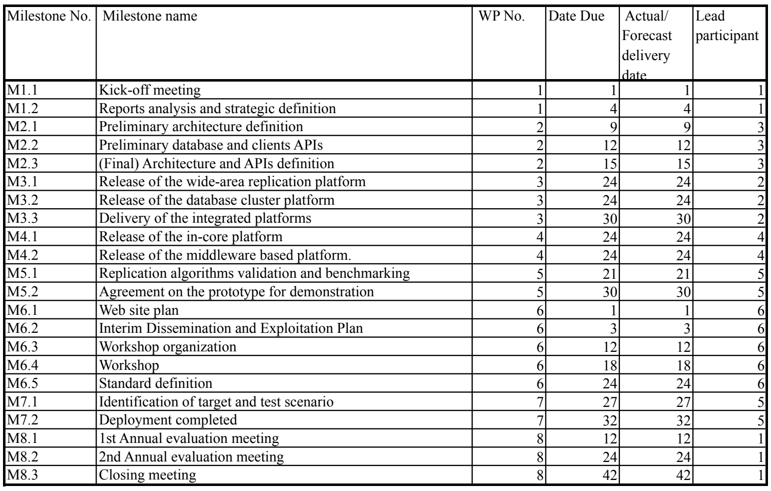
2.7.5 List of Milestones

Please refer to Section 2.9.

2.8 Deliverables

Del. no.	Deliverable name	Workpackage no.	Date due	Actual / Forecast delivery date	Estimated indicative person-months	Used indicative person-months	Lead contratcor
D1.1	State of the Art Report	1	4	27	9	9	1
D1.2	User Requirements Report	1	4	44	9	9	1
D1.3	Strategic Research Directions Report	1	6	7	1	1	1
D2.1	Preliminary Architecture and APIs Report	2	12	12	25	23	3
D2.2	Architecture Definition Report	2	18	36	15	15	3
D2.3	APIs Definition Report	2	18	19	10	10	3
D3.1	Wide-area Protocols Report	3	18	31	43	43	2
D3.2	Cluster Oriented Protocols Report	3	18	31	43	43	2
D3.3	Replication Modules Reference Implementation	3	24	27	40	40	2
D3.4	Modules Description and Configuration Guide	3	30	36	7	7	2
D3.5	Group Communication Protocols Report (NEW)	3		36	6	6	2
D4.1	In-core Mapping Report	4	12	12	10	10	4
D4.2	Middleware Mapping Report	4	12	12	10	10	4
D4.3	In-core Proof of Concept	4	24	44	24	24	4
D4.4	Middleware Proof-of-concept	4	24	43	28	28	4
D4.5	Hybrid Proof-of-concept	4	24	27	8	8	4
D4.6	DB Support Description and Configuration Guide	4	30	42	3	3	4
D5.1	Algorithms Performance and Reliability Assessment Report	5	24	31	3	3	5
D5.2	Prototype of the Integrated System	5	30	36	30	30	5
D5.3	Interface and Modules Peformance Assessment Report	5	42	43	3	3	5
D5.4	Management Tools Set	5	42	43	12	12	5
D5.5	Deployment Guides For Replication Strategies	5	42	44	2	2	5
D6.1	Web site structure and front page	6	1	1	1	1	1
D6.2	Project Prsentation - web site	6	6	1	1	1	1
D6.3	Workshop proeedings	6	18	11	6	6	1
D6.4	Draft Standard- GAPI	6	24	33	6	6	1
D6.4b	Draft Standard - jGCS (NEW)	6		31	6	6	1
D6.5	Final Dissemination and Exploitaion Plan	6	33	37	6	6	1
D7.1	Deployment plan	7	30	36	5	5	5
D7.2	Demonstration	7	33	44	14	14	5
D7.3	Evauation Report	7	42	36	3	3	5
D8.1	1st Periodic Management Report	8	12	12	6	6	1
D8.2	2nd Periodic Management Report	8	24	24	6	6	1
D8.3	Final Report	8	42	45	6	6	1

2.9 Milestones



Milestone No.	Milestone name	WP No.	Date Due	Actual/ Forecast delivery date	Lead participant
M1.1	Kick-off meeting	1	1	1	1
M1.2	Reports analysis and strategic definition	1	4	4	1
M2.1	Preliminary architecture definition	2	9	9	3
M2.2	Preliminary database and clients APIs	2	12	12	3
M2.3	(Final) Architecture and APIs definition	2	15	15	3
M3.1	Release of the wide-area replication platform	3	24	24	2
M3.2	Release of the database cluster platform	3	24	24	2
M3.3	Delivery of the integrated platforms	3	30	30	2
M4.1	Release of the in-core platform	4	24	24	4
M4.2	Release of the middleware based platform.	4	24	24	4
M5.1	Replication algorithms validation and benchmarking	5	21	21	5
M5.2	Agreement on the prototype for demonstration	5	30	30	5
M6.1	Web site plan	6	1	1	6
M6.2	Interim Dissemination and Exploitation Plan	6	3	3	6
M6.3	Workshop organization	6	12	12	6
M6.4	Workshop	6	18	18	6
M6.5	Standard definition	6	24	24	6
M7.1	Identification of target and test scenario	7	27	27	5
M7.2	Deployment completed	7	32	32	5
M8.1	1st Annual evaluation meeting	8	12	12	1
M8.2	2nd Annual evaluation meeting	8	24	24	1
M8.3	Closing meeting	8	42	42	1

3 Consortium Management

The project management for the current reporting period ran smoothly with the timely achievement of all planned milestones and fulfillment of the workpackages' deliverables.

The major problem faced by the consortium was the unavailability of one of the partners, MySQL AB, to contribute to the main R&D workpackages as originally planned. MySQL based its temporary withdrawal from the project's activity on its the need to concentrate efforts on the upcoming release of their main product that was being successively delayed by the lack of resources. While this was unfortunate, the consortium, as a whole, had no trouble in overcoming the situation and taking over MySQL's duties in the project. In the current reporting period, all partners assumed MySQL's work in WP1 and WP2, and EMIC ensured MySQL's planned preliminary work in WP4. The consortium is currently deciding on the work distribution of workpackages WP4 and WP5 for the coming year.

Due to changes occurred in INRIA's team in the second quarter of the current reporting period, a joint proposal from INRIA and EMIC in the sense of exchanging duties and responsibilities in WP4 and WP5 for the 2nd and 3rd years has been presented to the consortium.

An updated resource plan for the 2nd and 3rd years of the project reflecting the above changes will be submitted to the Commission's consideration in the very beginning of the next reporting period. No changes are foreseen to the current planned project timetable, though.

The consortium met twice: in the first days of the project for the Kick-off meeting in Braga, Portugal, and for the mid-year Executive Board meeting during 26/27 May in Grenoble, France. The reports for the meetings as well as the partners' presentations are available through the project's web site.

In January 2005, Edward Archibald (EMIC) traveled to Braga and stayed there for an extended period in order to gain a solid understanding of the DBSM (Database State Machine) as well as to make his own contributions on the topic of an active replication technology that uses in-core database modifications to provide for deterministic execution of SQL statements.

During July 14 and 15, the teams from USI, FFCUL and UMINHO met in Braga for a two-day technical meeting to discuss GAPI issues and the in-core implementation in PostgreSQL. Recently, taking advantage from the presence of most partners at the VLDB 2005 conference, a meeting to discuss and prepare the end year deliverables took place.

Initially promoted by the Commission organized Concertation Meetings a fruitful collaboration is flourishing between GORDA and three other FP6 IST projects, namely Dedisys, Rodin and AOSD-Europe.

The project management for the current reporting period ran smoothly with the timely achievement of all planned milestones and fulfillment of the work-packages' deliverables.

The consortium met once for an Executive Board meeting during 15/19 February in Lugano, Switzerland. The reports for the meeting as well as the partners' presentations are available through the project's web site.

Initially promoted by the Commission organized Concertation Meetings a fruitful collaboration is flourishing between GORDA and three other FP6 IST projects, namely Dedisys, Rodin and AOSD-Europe.

The project management for the current reporting period ran smoothly with the timely achievement of all planned milestones and fulfillment of the work-packages' deliverables.

The consortium met once for an Executive Board meeting during 15/19 February in Lugano, Switzerland. The reports for the meeting as well as the partners' presentations are available through the project's web site.

Initially promoted by the Commission organized Concertation Meetings a fruitful collaboration is flourishing between GORDA and three other FP6 IST projects, namely Dedisys, Rodin and AOSD-Europe.

4 Annex – Plan for using and disseminating the knowledge

4.1 Exploitable Knowledge and its Use

Overview table

Exploitable knowledge (description)	Exploitable product(s) or measure(s)	Sector(s) of application	Timetable for commercial use	Patents or other IPR protection	Owner & Other Partner(s) involved
Database replication protocols	LAN + WAN database replication protocols		2007		ALL
Group-communication toolkit	Group-communication protocols		2007		ALL

4.2 Dissemination of Knowledge

Overview table

	Planned / actual Dates	Type	Type of audience	Countries addressed	Size of audience	Partner responsible / invloved
1	Oct 2004	Project web-site	General public			UMINHO/ALL
2	Aug 2005	workshop	Research		22	UMINHO/ALL
3	2005	Publications	Research			ALL
4	Apr 2006	Conference Track	Research			UMINHO/ALL

1. The project web-site is available at http://gorda.di.uminho.pt

2. Workshop on "Design, Implementation and Deployment of Database Replication" co-located with the Very Large Data Bases 2005 Conference, Trondheim, Norway, August, 2005.

3. Publications listed in Annex 5.

4. Track on “Dependable and Adaptive Distributed Systems” in the 21st ACM Symposium on Applied Computing, to be held in Dijon, France in April 2006.

5 Annex - Publications

[1] M. Matos, Jr. A. Correia, J. Pereira, and R. Oliveira. Serpentine: adaptive middleware for complex heterogeneous distributed systems. In *SAC '08: Proceedings of the 2008 ACM symposium on Applied computing*, pages 2219–2223. ACM, 2008.

[2] R. Schmidt N. Schiper and F. Pedone. Optimistic algorithms for partial database replication. In *10th International Conference on Principles of Distributed Systems (OPODIS'2006)*, 2006. (Brief announcement at 20th International Symposium on Distributed Computing, DISC'2006).

[3] A. Correia J. Pereira R. Oliverira J. Grov, L. Soares and F. Pedone. A pragmatic protocol for database replication in interconnected clusters. In *12th IEEE International Symposium on Pacific Rim Dependable Computing (PRDC'2006)*, 2006.

[4] F. Pedone L. Camargos and R. Schmidt. A primary-backup protocol for in-memory database replication. In *5th IEEE International Symposium on Network Computing and Applications (NCA'2006)*, 2006.

[5] E. Madeira L. Camargos and F. Pedone. Optimal and practical wab-based consensus algorithms. In *European Conference on Parallel Computing (EuroPar'2006)*, 2006.

[6] S. Dropsho S. Elnikety and F. Pedone. Tashkent: Uniting durability with transaction ordering for high-performance scalable database replication. In *EuroSys 2006*, 2006.

[7] F. Pedone and S. Frolund. Pronto: High availability for standard off-the-shelf databases. *Journal of Parallel and Distributed Computing*, 68(2):150–164, 2008.

[8] F. Pedone L. Camargos and M. Wieloch. A highly available log service for distributed transaction termination. In *7th International Symposium on Parallel and Distributed Computing (ISPDC 2008)*, 2008.

[9] R. Schmidt L. Camargos and F. Pedone. Multicoordinated agreement protocols for higher availability. In *7th IEEE International Symposium on Network Computing and Applications (NCA 2008)*, 2008. (also appears as a brief annoucement in the Symposium on Principles of Distributed Computing (PODC 2007)).

[10] N. L. Duarte and F. Pedone. dsmdb: A distributed shared memory approach for building replicated database systems. In *2nd Workshop on Dependable Distributed Data Management (WDDDM 2008)*, 200.

[11] V. Zuikeviciute and F. Pedone. Conflict aware load balancing techniques for database replication. In *23rd ACM Symposium on Applied Computing (ACM SAC 2008)*, 2008.

[12] N. Schiper and F. Pedone. Optimal atomic broadcast and multicast algorithms for wide area networks. In *9th International Conference on Distributed Computing and Networking (ICDCN 2008)*, 2008. (also appears as a brief annoucement in the Symposium on Principles of Distributed Computing (PODC 2007)).

[13] R. Schmidt and F. Pedone. A formal analysis of the deferred update technique. In *11th International Conference On Principles Of Distributed Systems (OPODIS'2007)*, 2007. (also appears as a brief announcement in the 21st International Symposium on Distributed Computing (DISC'2007)).

[14] R. Schmidt L. Camargos and F. Pedone. Multicoordinated paxos, brief announcement. In *26th Symposium on Principles of Distributed Computing (PODC'2007)*, 2007.

[15] F. Pedone L. Camargos and M. Wieloch. Sprint: A middleware for high-performance transaction processing. In *2nd European Conference on Systems Research (EuroSys'2007)*, 2007.

[16] J. Pereira L. Rodrigues R. Oliveira N. Carvalho, A. Correia Jr. and S. Guedes. On the use of a reflective architecture to augment database management systems. *Journal of Universal Computer Science*, 13(8):1110–1135.

[17] L. Rodrigues N. Carvalho, João Cachopo and António Rito Silva. Versioned transactional shared memory for the fenixedu web application. In *Second Workshop on Dependable Distributed Data Management (in conjunction with Eurosys 2008)*, 2 2008.

[18] N. Carvalho L. Rodrigues and Emili Miedes. Supporting linearizable semantics in replicated databases. In *The 7th IEEE International Symposium on Network Computing and Applications (IEEE NCA08)*, 7 2008. (short paper).

[19] S. Bouchenak and N. de Palma. Message Queuing Systems. In *Springer Encyclopedia of Database Systems)*, 2008. to appear.

[20] C. Taton, N. de Palma, and S. Bouchenak. Adaptive Middleware for Message Queuing Systems. In *Springer Encyclopedia of Database Systems)*, 2008. to appear.

[21] N. De Palma, S. Bouchenak, F. Boyer, D. Hagimont, S. Sicard, and C. Taton. Jade : Un Environnement d'Administration Autonome. *Journal of Technique et Science Informatiques (TSI)*, 2008. to appear.

[22] C. Taton, N. De Palma, S. Bouchenak, and D. Hagimont. Self-Optimization of Clustered Message-Oriented Middleware. In *9th International Symposium on Distributed Objects, Middleware and Applications (DOA 2007)*, Vilamoura, Algarve, Portugal, November 2007.

[23] C. Taton, N. De Palma, J. Philippe, and S. Bouchenak. Self-Optimization of Clustered Message-Oriented Middleware. In *4th IEEE International Conference on Autonomic Computing (IEEE ICAC 2007)*, Jacksonville, FL, USA, June 2007. short paper.

[24] S. Sicard, F. Boyer, and N. de Palma. Using Components for Architecture-Based Management: The Self-Repair Case. In *30th International Conference on Software Engineering (ICSE'08)*, Leipzig, Germany, May 2008.

[25] J. Arnaud and S. Bouchenak. Modeling and Capacity Planning of Internet Services. Technical report, INRIA, 2008. to appear.

[26] D. Hagimont S. Krakowiak A. Mos N. de Palma V. Quema S. Bouchenak, F. Boyer and J. B. Stefani. Architecture-based autonomous repair management: An application to j2ee clusters. In *Proceedings of 24th IEEE Symposium on Reliable Distributed Systems*, 2005.

[27] D. Hagimont S. Krakowiak N. de Palma V. Quema S. Bouchenak, F. Boyer and J. B. Stefani. Architecture-based autonomous repair management - application to j2ee clusters. In *Proceedings of the 2nd IEEE International Conference on Autonomic Computing (ICAC-05)*, 2005.

[28] N. de Palma S. Bouchenak and D. Hagimont. Autonomic administration of clustered j2ee applications. In *Proceedings of IFIP/IEEE International Workshop on Self-Managed Systems Services (SelfMan 2005)*, 2005.

[29] F. Boyer N. de Palma D. Hagimont C. Taton, S. Bouchenak and A. Mos. Self-manageable replicated servers. In *roceedings of VLDB Workshop on Design, Implementation, and Deployment of Database Replication*, 2005.

[30] F. Pedone S. Elnikety and W. Zwaenepoel. Database replication using generalized snapshot isolation. In *Proceedings of 24th IEEE Symposium on Reliable Distributed Systems*, 2005.

[31] R. Schmidt and F. Pedone. Consistent main-memory database federations under deferred disk writes. In *Proceedings of 24th IEEE Symposium on Reliable Distributed Systems*, 2005.

[32] V. Zuikeviciute and F. Pedone. Revisiting the database state machine approach. In *Proceedings of the VLDB Workshop on Design, Implementation, and Deployment of Database Replication*, 2005.

[33] L. Soare A. Correia Jr. L. Rocha R. Oliveira A. Sousa, J. Pereira and F. Moura. Testing the dependability and performance of group communication based database replication protocols. In *Proceedings of the International Conference on Dependable Systems and Networks (DSN'05)*, 2005.

[34] L. Soares J. Pereira F. Moura A. Correia Jr., A. Sousa and R. Oliveira. Group-based replication of on-line transaction processing servers. In *2nd Latin-American Symposium on Dependable Computing*, 2005.

[35] N. Carvalho S. Guedes, V. Conceição. Plataforma de desenvolvimento e simulação de protocolos. In *8^a Conferência sobre Redes de Computadores, CRC 2005*, 2005.

[36] J. Mocito L. Rodrigues, N. Carvalho. Total order: How optimistic can you be. In *Submitted to the 21st ACM Symposium on Applied Computing, Track on Dependable and Adaptive Distributed Systems*, 200.

[37] N. Carvalho L. Rodrigues, J. Mocito. From spontaneous total order to uniform total order: different degrees of optimistic delivery. In *Proceedings of the 2006 ACM Symposium on Applied Computing*, 2006.

[38] J. Mocito and L. Rodrigues. Run-time switching between total order algorithms. In *In Proceedings of the European Conference on Parallel Computing, Euro-Par 2006*, 2006.

[39] J. Mocito. Run-time switching between total order algorithms. Master's thesis, University of Lisbon, Portugal, 2006.

[40] L. Rodrigues N. Carvalho, J. Pereira. Towards a generic group communication service. In *In Proceedings of the 8th International Symposium on Distributed Objects and Applications (DOA)*, 2006.

[41] A. Respício J. Mocito and L. Rodrigues. On statistically estimated optimistic delivery in large-scale total order protocols. In *In Proceedings of the 12th IEEE International Symposium Pacific Rim Dependable Computing (PRDC'06)*, 2006.

[42] A. Correia R. Oliveira N. Carvalho S. Guedes L. Rodrigues, J. Pereira. Database interfaces for replication support (extended abstract). In *In Dagstuhl Seminars*, 2006.

[43] A. Correia R. Oliveira, J. Pereira and E. Archibald. Revisiting 1-copy equivalence in clustered databases. In *Proceedings of the 2006 ACM Symposium on Applied Computing*, 2006.

[44] F. Moura J. Pereira R. Oliveira A. Sousa, A. Correia. Evaluating certification protocols in the partial database state machine. In *DILSOS workshop - in the Proceedings of ARES 2006 - The First International Conference on Availability, Reliability and Security*, 2006.

[45] R. Schmidt N. Schiper and F. Pedone. Optimistic algorithms for partial database replication. Technical Report 2006/02, UNISI, 2006.

[46] V. Zuikeviciute and F. Pedone. Conflict-aware load-balancing techniques for database replication. Technical Report 2006/01, UNISI, 2006.

[47] F. Pedone L. Camargos and R. Schmidt. A primary-backup protocol for in-memory database replication. In *5th IEEE International Symposium on Network Computing and Applications (NCA'2006)*, 2006.

[48] E. Madeira L. Camargos and F. Pedone. Optimal and practical wab-based consensus algorithms. In *European Conference on Parallel Computing (EuroPar'2006)*, 2006.

[49] S. Dropsho S. Elnikety and F. Pedone. Tashkent: Uniting durability with transaction ordering for high-performance scalable database replication. In *EuroSys 2006*, 2006.

[50] D. Hagimont S. Bouchenak, N. De Palma and C. Taton. Autonomic management of clustered applications. In *In IEEE International Conference on Cluster Computing*, 2006.

[51] D. Hagimont S. Krakowiak S. Bouchenak, N. De Palma and C. Taton. Autonomic management of internet services: Experience with self-optimization (short paper). In *In 3rd International Conference on Autonomic Computing (ICAC)*, 2006.

[52] N. De Palma D. Hagimont C. Taton, S. Bouchenak and Sylvain Sicard. Self-optimization of clustered databases. In *In 2nd International IEEE WoWMoM Workshop on Autonomic Communications and Computing (ACC 2006)*, 2006.

[53] N. De Palma S. Sicard and D. Hagimont. J2ee server scalability through ejb replication. In *In 21st ACM Symposium on Applied Computing (SAC'06)*, 2006.

[54] S. Bouchenak F. Boyer J. Philippe, N. De Palma and D. Hagimont. A black-box approach for web application sla. In *In 21st ACM Symposium on Applied Computing (SAC'06)*, 2006.