Towards Exploitation of Event Semantics in Event Dissemination Architectures for Massive Multiuser Virtual Environments

Thomas Fischer[*] , Richard Lenz[†]
University of Erlangen-Nuremberg
Department of Computer Science 6 (Data Management)
Martensstrasse 3
D-91053 Erlangen
{thomas.fischer,richard.lenz}@cs.fau.de

ABSTRACT

Triggered by the fast evolving technical capabilities for implementing distributed global scale applications, online games have grown to a huge industry in recent years. Particularly, *Massive Multiuser Virtual Environment*s (MMVEs), which allow for simultaneous activity of thousands of players in a virtual world, have been tremendously successful. Current architectures, however, use centralized approaches, which obviously do not scale beyond a certain point. Distributed event-based systems are a promising approach to reach both performing and scalable architectures. The potential of this approach can only be fully exploited if event semantics is used to optimize event handling. Existing approaches actually do this to some degree, but typically in a very application specific manner. There is no generally applicable framework for classifying events according to their relevant semantic properties. In this paper, we give a thesis proposal to address the challenge of such a general framework for MMVEs.

1. INTRODUCTION

Computer games have become part of our social culture and grown to an important branch for the computer industry. MMVEs define a distributed virtual world shared by thousands of participants, each represented by an avatar, who compete and cooperate in one enormous persistent world. This world may be a game world as in *Massive Multiplayer Online Game*s (MMOGs), a large scale simulation or a virtual world like Second Life. The design of such a living virtual world requires armies of artists and the software backing a world of such enormous dimensions must satisfy several requirements, like a consistent interactive experience, high availability and a persistent world.

In contrast to conventional computer games, MMVEs must be able to maintain high quality regarding these requirements even if the user-base grows beyond any predictions. This states the challenge of MMVE architectures: Preserve scalability, whilst all other requirements are met to the desired quality standard.

Current industry strength MMVE architectures favour a client/server architectural style, mostly motivated by the fear of cheats and the easy maintainability. Skibinsky showed in [8] that this type of architecture has "high operational cost, capable of serving in the low thousands of users in the same world and having scalability limits for future growth". To cope with more and more players, tremendous effort is invested by the usage of grid approaches like in Linden Lab's *Second Life* [1] or the deployment of large hierarchical clusters like for the operation of CCP's *Eve Online*[2]. These architectures serve currently a confirmed maximum of about 47 thousand users (Eve Online) in one persistent virtual world [3].

Recent deployed virtual environments have shown that those approaches alone do not suffice to solve the challenge of scalability. Distributed systems like specialized *Peer-to-peer* (P2P) systems are a promising approach of handling the snowballing number of players. This includes both, systems using P2P for organization of a cluster or cloud and completely decentralized systems, with the usage of all peers, user operated or not. But even in such systems, the main challenge in handling player concentration in one region is the enormous amount of *events* that have to be delivered to every single player.

Existing multiplayer game architectures often use a broadcast mechanism to keep the distributed world state consistent. Ideally, all events are securely delivered and processed in the same order nearly at the same time on all clients. In a distributed environment, this requires an effort of $O(n^2)$, n being the participating clients of the world. It is obvious that architectures do not scale well beyond a certain amount of clients. Multiplayer games like Quake 3 Arena [3] have shown that without any optimizations, the limit is reached by about 64 clients, depending on the required update rate

[*]PhD student
[†]supervising Professor

[1]`http://secondlife.com`
[2]`http://www.eveonline.com/`
[3]`http://www.idsoftware.com/games/quake/quake3-arena/`

of the game. Enabling scalability beyond that limit states a challenge. Many recent event-dissemination architectures for MMVEs [5, 2, 9, 4] optimize the event dissemination based on one basic idea: Exploitation of event semantics to reduce messages exchanged between clients. But they all address only one or two aspects of the semantics and propose an optimized event-dissemination architecture for that specialized aspect.

2. THESIS PROPOSAL

In the ***M**assive **M**ultiuser **E**ven**t** **I**nfra**S**tructure* (m^{2}etis) project, we strive for a distributed event dissemination architecture enabling optimization by the exploitation of specific semantic requirements of each event type. In contrast to those existing approaches this project aims for an holistic approach, incorporating all performance relevant aspects of events in MMVEs. We believe that the consideration of all semantic aspects a certain event type has and the customized optimization of each event type according to its semantic properties, will pay off in a significant performance boost over existing optimization strategies in this field.

In a domain with an homogeneous software architecture like an MMVE, each event has defined semantic properties. Therefore they may be classified a priori based on those properties. With such a classification of event types, optimization may take place along those semantic properties. We currently build a framework which in future shall be capable to choose from a catalogue of optimization strategies which method is optimal in respect to the semantic properties of each event type in the system. This framework is based on an existing P2P overlay architecture (Chimera [1]) which is currently extended by a distributed publish/subscribe system providing the communication paradigm for our system.

In order to support optimizable publish/subscribe trees, a central semantic classification is required, as well as a corresponding cost model which feeds an optimizer algorithm. Moreover, the publish/subscribe system must provide a model to incorporate arbitrary optimization strategies and a managing component, coping with all different dissemination strategies. In the following, an outline of the different challenges is given, which have to be addressed to complete the development of such a system. This also defines the scope of the ongoing PhD thesis founding this project. First we briefly discuss the classification approach we use, before a sketch of the planned architecture is given.

2.1 Classification

Events used in MMVEs feature many characteristics like a spatial context in which the event is valid. That may be exploited to develop specialized architectures for their delivery to the clients. The decision which optimization suites best for a certain event type is a non-trivial challenge, whose solution in our approach needs a solid model of the events semantics.

The semantics of each event, which is exchanged between clients of a virtual world, may be analyzed and a definition e.g. of the address, the priority, the relationship to other events or the context in which this event is valid can be deduced. In order to classify the event types properly, we propose a multidimensional classification with orthogonal dimensions to model independent aspects of the event semantics. We define disjoint characteristic classes for each

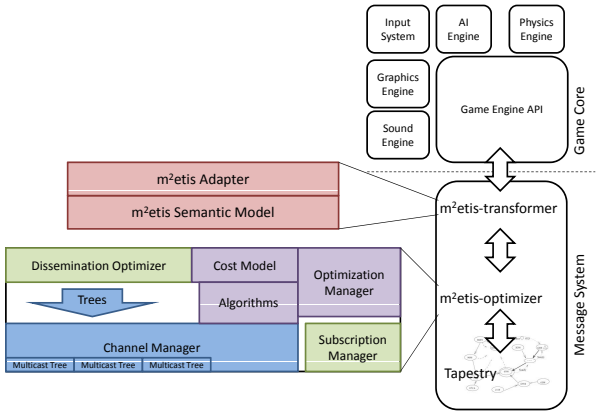

Figure 1: m2etis architecture

dimension. To classify an event type, it has to be assigned to one class along each dimension. Based on this scheme, the class of an event type is defined as the sum of the characteristics along each dimension. The power of such an multidimensional class space enables optimization of each event type along each dimension with a different strategy in order to gain a better overall optimization footprint of the system, than if a fixed strategy is used for the whole system. We defined an initial set of dimensions: context, synchronization, persistency, validity, delivery and security. More dimensions are to be found as analysis of more MMVE architectures is required. All dimensions found, are subject to existing specialized optimization strategies that concentrate on those single dimensions. In our approach we incorporate all those dimensions, as we propose a multidimensional schema as basis of our optimization.

2.2 Architecture

The realization of an architecture supporting adaptable optimizations states many challenges. In this section we identify the required modules to implement such a framework. Singhal and Zyda [7] identify the main technical challenge for networked virtual environments as the management of their dynamic shared state. In our m^{2}etis project, we currently develop a framework to improve the event dissemination performance following our multidimensional classification.

2.3 m^{2}etis architecture

The m^{2}etis framework strives for the implementation of a networking middleware for MMVEs which integrates seamlessly as the messaging system into game engines. It is designed around the central multidimensional semantic event classification, which makes the framework adaptable to many scenarios. Probably even to applications beyond MMVEs, as long as the event processing models correspond to each other. For example simulations of complex emergency scenarios could be distributed and speed up with our approach. To enable such a general applicability, each application has to provide a semantic classification of its event types according to our schema. Based on the definition of the type itself and its semantic properties, the m^{2}etis system decides which algorithm is chosen to disseminate each event type across the participating nodes.

Fig. 1 shows the modules of our architecture. The system consists of 3 parts: The m^{2}etis transformer responsible for mediation between game engine and the internal event model, the m^{2}etis optimizer which optimizes the event dissemination and creates the according optimized channels for its publish/subscribe system and the underlying P2P overlay network based on Chimera [1] a Tapestry advancement, which provides the routing capabilities for event distribution.

m^{2}etis transformer The transformer component consists of the formalized semantic model based on the multidimensional classification and the m^{2}etis adapter. The model has to be instantiated for each application to reflect the semantic properties of the application's event types, meaning each event type used by the engine or the application has to be classified following the semantic model. The m^{2}etis adapter component provides the API to the other subsystems of the application engine. This API allows messaging and provides replicas of game states as required by the engine. The adapter moreover transforms the states and messages of the API to an internal event format which allows the m^{2}etis optimizer to calculate the optimized delivery properties.

m^{2}etis optimizer In this component the internal event representation is used to generate and operate optimized Publish/Subscribe multicast trees. Based on the semantic model and the internal event representation the dissemination optimizer deduces with the help of semantic application information the recipients and the dissemination strategy for each event type. This happens based on the catalogue, the optimization manager provides, which holds all optimization algorithms and their corresponding costs. Following a cost model the dissemination optimizer is able to calculate the optimal algorithm of a certain event type. Based on this decision the corresponding channel is created. The resulting optimized dissemination structure is controlled by the channel manager and the subscription manager, which controls all moving subscriptions and the resulting changes in the dissemination structure as well as the dissemination of events itself.

Tapestry Based on the managed dissemination structures for each type, the channel manager is able to route each occurring event following to the associated algorithm and recipients. The routing itself is conducted by an underlying P2P Overlay-Network like Pastry [6] or Chimera [1], a Tapestry implementation. Basically any structured Overlay-Network is applicable as long as it provides routing capabilities.

The challenges in the depicted architecture are on the one hand the automation of the transformation steps to minimize the manual work. We aim for two inputs to deduce the optimized publish/subscribe multicast trees: A semantic model of the event types and a list of optimization algorithms with their costs. Whereby the algorithms and costs must not be adapted for every application, as a catalogue of supported algorithms is provided by the middleware. On the other hand the optimizer component with its cost model and formal reasoner is the other challenge to address.

3. CONCLUSION AND FUTURE WORK

We introduce the m^{2}etis project which aims for a generic architecture to optimize the amount of messages required in an MMVE without hampering quality requirements. In this paper, we sketched the basic idea behind our approach and identified the challenges that arise from our planned solution in the m^{2}etis project, which will be addressed in the ongoing PhD thesis.

We are currently implementing a prototype for the described architecture. The current focus lies on the realization of a publish/subscribe system, capable of handling the required different event dissemination strategies and algorithms. The challenge is, handling moving spatial subscriptions and integrating arbitrary optimization algorithms in the publish/subscribe component. To enable such an architecture which is able to integrate arbitrary optimization algorithms, a generalization of the optimization strategies is required as well as the corresponding transformation methodology to map the generic event classification schema on optimization strategies, which may be deployed on the overlay network in form of a publish/subscribe channel. Moreover a formal semantic model based on the proposed classification has to be developed. Complementing it with a cost model, enables an optimizer component that allows automated customized optimization for application-specific requirements.

Existing approaches only cope with a subset of our identified dimensions, therefore our idea to compose different optimization strategies based on our multidimensional semantic classification poses a holistic approach. We understand the thesis outline presented in this paper as a first step on the road to generic event dissemination systems which are optimizable based on the knowledge of application-driven event semantics.

4. REFERENCES

[1] M. S. Allen and R. Alebouyeh. Chimera: A Library for Structured Peer-to-peer Application Development. Technical report, University of California, Santa Barbara, 2006.

[2] A. Bharambe, J. R. Douceur, J. R. Lorch, T. Moscibroda, J. Pang, S. Seshan, and X. Zhuang. Donnybrook: Enabling Large-Scale, High-Speed, Peer-to-Peer Games. *SIGCOMM Comput. Commun. Rev. (CCR)*, 38(4):389–400, 2008.

[3] H. Fannar, V. Coleman, R. Breen, and B. V. Slyke. The server technology of eve online: How to cope with 300,000 players on one server. Talk on GDC Austin, 2008.

[4] S. Ferretti. A Synchronization Protocol For Supporting Peer-to-Peer Multiplayer Online Games in Overlay Networks. In *2nd International Conference on Distributed Event-Based Systems (DEBS)*, pages 83–94, New York, NY, USA, 2008. ACM.

[5] S.-Y. Hu, J.-F. Chen, and T.-H. Chen. VON: A Scalable Peer-to-Peer Network for Virtual Environments. *IEEE Network*, 20(4):22–31, July 2006.

[6] A. I. T. Rowstron and P. Druschel. Pastry: Scalable, Decentralized Object Location, and Routing for Large-Scale Peer-to-Peer Systems. In *18th IFIP/ACM International Conference on Distributed Systems Platforms (Middleware)*, pages 329–350.

Springer-Verlag, 2001.

[7] S. Singhal and M. Zyda. *Networked Virtual Environments: Design and Implementation.* Addison-Wesley Professional, New York, NY, USA, 1999.

[8] M. Skibinsky. *Massive Multiplayer Game Development 2*, chapter 5 - The Quest for the Holy Scale-Part 1: Large-Scale Computing, pages 339–355. Game Development Series. Charles River Media, 2 edition, February 2005.

[9] K. Zhang, B. Kemme, and A. Denault. Persistence in Massively Multiplayer Online Games. In *7th ACM SIGCOMM Workshop on Network and System Support for Games (NetGames)*, pages 53–58, New York, NY, USA, 2008. ACM.