

# An Object Driven Partitioning Approach for Distributed Virtual Environments

Christos Bouras<sup>1,2</sup>, Eri Giannaka<sup>1,2</sup>, Thrasyvoulos Tsiatsos<sup>1,3</sup>

<sup>1</sup> *Research Academic Computer Technology Institute, Patras, Greece*

<sup>2</sup> *Computer Engineering and Informatics Dept., Univ. of Patras, Greece*

<sup>3</sup> *Department of Informatics, Aristotle University of Thessaloniki, Thessaloniki, Greece*  
*bouras@cti.gr, giannaka@cti.gr, tsiatsos@csd.auth.gr*

## Abstract

*Distributed virtual environments tend to become a de facto solution for large-scale networked virtual environments. One of the key issues and problems that need to be handled in the design of a scalable and effective distributed virtual environment system is the partitioning problem, which refers to the efficient assignment of the system's workload to the available resources of the system. This paper presents an object driven partitioning approach based on the partitioning algorithm of Lui & Chan, who proposed a three step technique for balancing the workload among the servers of the Distributed Virtual Environment.*

## 1. Introduction

Distributed Virtual Environments (DVEs) emerged as a solution for facing the limitations and problems of large-scale networked virtual environments. One of the key issues in the design of a scalable DVE system is the partitioning problem. The partitioning problem is related to the efficient assignment of the workload to the servers of the system. To this direction many algorithms and techniques have been proposed, each of which aims at eliminating and overcoming certain aspects of this problem. This paper presents an object-driven approach, which is based on prior work conducted by Lui&Chan [1], for the effective assignment of the existing workload to the available servers of a DVE system.

## 2. Object Driven Partitioning Algorithm

The approach of Lui & Chan [1], introduces a quality function (denoted as  $C_p$ ), for evaluating each assignment of clients to servers. This quality function takes into account two parameters: a) the computing

workload ( $C_p^W$ ) generated by clients (avatars) in the DVE system, which should be shared among the available servers in regard to their computing resources and b) the overall inter-server communication ( $C_p^L$ ). In particular,  $C_p$  is calculated as the sum of  $C_p^W$  and  $C_p^L$  ( $C_p = 0.5 * C_p^W + 0.5 * C_p^L$ ). The aforementioned approach does not take into account the presence of shared (or active) objects within the virtual environment, which increase workload to the servers as well as the communication cost due to the object-avatar interaction. Based on these observations, the partitioning approach presented in this paper takes into account the presence of objects and handles these entities differently for optimizing the algorithm's efficiency and performance.

### 2.1 Object-Driven Recursive Bisection Partitioning (O-RBP) Algorithm

The main concept of the O-RBP algorithm, which constitutes the first step of our approach, is a prior assignment of the virtual world to the available servers of the system. In particular, in this step, the virtual world is divided into disjoint cells, each of which now comprises a number of avatars and active objects. In each step we consider a pair of servers, and initially all cells are placed in one server. For this server we calculate the total cost for moving one cell from the full server to an "empty" one. This process is realized for all cells of the virtual world. The cell with the lowest total cost is "virtually" assigned to the second server. After this cell is moved, we repeat the same process for the cells that remain in the first server and we again select the one that minimizes the total cost. The total cost for moving cells from one server to another is denoted as partition policy. When this process is completed for all cells, we select the

partition policy with the minimum cost and assign the cells it involves to the “empty” server. When the assignment is completed, the server with the higher cost is again selected and the same process takes place for another “empty” server.

## 2.2 Object Driven Layering Partitioning (O-LP) Algorithm

The main objective of this algorithm is to find a new partition based on the results of the O-RBP algorithm. In particular, this step takes into account the entities of the platform (avatars and active objects) and reassigns them with an optimal way to the available servers, so as to balance the existing workload. The communication cost in this case is calculated by the interactions between the avatars, which is bidirectional, and the interaction between avatars and objects, which is unidirectional.

## 2.3 Object Driven Communication Refinement Partitioning (O-CRP) Algorithm

After the initial assignment of the existing cells to the available servers (O-RBP algorithm) and the rebalancing of the workload cost (O-LP) the step that follows is the rebalancing of the communication cost, which is realized through the O-CRP algorithm. The concept of this algorithm is to exchange border avatars and objects that communicate with entities placed in other partitions, so as to minimize the server-server communication.

## 3. Experimental Results for a Large-Scale Virtual Environment

The basic concept of the experiments is to test the algorithm’s performance for a larger scale environment. We calculate the quality function  $C_p$  under the following entity distributions: a) uniform, skewed and c) clustered. We encounter an average workload in the order of 10 for the avatars and in the order of 5 for the objects that each of this type of entities introduces to the server. We assume that the large virtual world has dimensions  $25 \times 25$  units, 2000 avatars and 800 objects. The available servers of the system equal to 8, the average diameter  $D$  is set to 1.0.

Large-Scale Virtual Environment - $C_p$			
Entities Scattered under Uniform distribution			
Algorithm	O-RBP	O-LP	O-CRP
Lui-Chan	17356.0	17338.5	16092.0
Object Driven	15915.5	15787.5	14705.0
Entities Scattered under Skewed distribution			

Algorithm	O-RBP	O-LP	O-CRP
Lui-Chan	22722.5	22702.5	20582.0
Object Driven	18719.5	18661.0	17251.0
Entities Scattered under Clustered distribution (3 clusters)			
Algorithm	O-RBP	O-LP	O-CRP
Lui-Chan	26223.0	25438.0	24851.0
Object Driven	23636.5	22900.5	21765.0

**Table 1: Total system cost for large-scale virtual environments**

Table 1 presents the results of the experiments conducted for the large-scale virtual environment for the avatar driven (Lui&Chan) approach as well as for the object driven approach proposed in this paper.

In all cases, the incorporation of objects achieves better results. As it can be extracted by the results presented, the large size of the virtual environment as well as the high number of the existing entities (both avatars and objects) increase significantly the value of the quality function. The high value of the quality function would be significantly lower in the case that the average diameter, which defines the AOI of the avatars, is lower. In particular, the wider the AOI, the higher the number of entities that interact, thus, increasing the communication cost.

## 4. Conclusion and Future Work

This paper described an object driven approach for solving the partitioning problem in Distributed Virtual Environments and presented the performance results extracted from the experiments conducted for testing the efficiency of the algorithm. The results of the experiments indicated that without increasing the algorithm’s complexity the actual incorporation of objects in the partitioning algorithm improves the efficiency, performance and value of the quality function. Based on the observation that objects can play a significant role within a virtual environment, some of our planned next steps include the extension of the approach for predicting avatars’ behavior in the virtual environment based on the objects’ attributes and degree of interaction.

## 10. References

- [1] John C.S. Lui, M.F. Chan: An Efficient Partitioning Algorithm for Distributed Virtual Environment Systems, IEEE Transactions on Parallel and Distributed Systems, Volume 13, No. 1, (Jan 2002)