CatNet

ITS-2001-34030

Evaluation of the Catallaxy paradigm for decentralized operation of dynamic application networks

Delivery 4: Assessment Final Report

Covering period 01.03.2002-28.02.2003

Report Version: 0.2

Report Preparation Date: 28.02.03

Classification: public

Contract Start Date: 01.03.2002 Duration: 12 months

Project Co-ordinator: Universitat Politècnica de Catalunya

Partners: Albert-Ludwigs-Universität Freiburg

Project funded by the European Community under the "Information Society Technologies" Programme (1998-2002)

Table of contents:

Delivery summary sheet

D4: Assessment Final Report

DELIVERABLE SUMMARY SHEET

Project Number: ITS-2001-34030

Project Acronym: CatNet

Title: Evaluation of the Catallaxy paradigm for decentralized operation of dynamic application networks

Deliverable N°: 4 – Assessment Final Report

Due date: 04/2003

Delivery Date: 04/2003

Short Description:

In this deliverable we give a summary of the technical results obtained in WP1, WP2, and WP3, explained in detail in the corresponding deliverables D1, D2 and D3, respectively.

Secondly, we describe the project management and coordination and finally the public dissemination of the project.

Partners owning: Albert-Ludwigs-Universität Freiburg

Partners contributed: Universitat Politècnica de Catalunya

Made available to: Public

1. Objectives and achievements of the project

Objectives:

The main goal of the project is to evaluate a decentralized economic control and coordination mechanism, the Catallaxy paradigm, for the decentralized control of dynamic application layer networks. By means of simulations the performance of the catallactic operated dynamic application network is compared with a Baseline system consisting of a centrally controlled application network, to assess the feasibility of the approach.

We consider application networks from real world, which are organized in distributed network nodes. These networks can be categorized on one hand in terms of node dynamics into quasi-static application networks whose resource nodes are highly available and very dynamic application networks whose resource nodes are disconnected for many intervals of time, and on the other hand in terms of node density into low node density application networks, whose ratio resource nodes number / total resources is very low, and high node density application networks, whose ratio resource nodes number / total resources is high.

The evaluation criterion focuses on the system performance as a whole, instead of the individual performance of single nodes. From an economic view, the overall success criterion is "maximum social welfare", which is the sum of all utilities of the participating nodes. This criterion balances both costs and revenue incurred by the nodes. Several technical criteria contribute to "social welfare", which are measured separately to provide additional information on the system's performance, such as the resource allocation efficiency of the system.

The goal of this project is to state relative orders of magnitude of "social welfare" in the different scenarios, rather than giving absolute values. Whether it should be considered to change to a Catallactic system from a Baseline system is beyond the technological focus of the project, as such issues depend on the concrete application, its technical setup, the business model and finally on the behavior of the human users.

The application network operated under the Catallaxy paradigm in comparison to the Baseline system is evaluated based on the performance under both coordination mechanisms in 4 of the scenarios of the design space.

Main technical achievements and results:

1) Development of the CATNET simulator (WP1).
2) Implementation of different coordination mechanisms in the simulator (WP1).
3) Simulations of different kinds of scenarios of application networks coordinated by the Catallactic and Baseline mechanism (WP2).
4) Performance evaluation of the application network simulated with the two control mechanisms (WP3).
5) Evaluation of the Catallactic coordination paradigm (WP3).

2. Project management and co-ordination

2.1 Institutions and involved persons

Institutions:

- Technical University of Catalonia (UPC), Computer Architecture Department.
- Albert-Ludwigs-Universität Freiburg, Institute for Computer Science and Social Studies (UF-IIG).

Persons involved in the project:

From UF-IIG: Torsten Eymann, Michael Reinicke.
From UPC: Oscar Ardaiz, Pau Artigas, Luis Diaz, Felix Freitag, Roc Messeguer, Leandro Navarro, Dolors Royo.

2.2 Co-operation and code development within the consortium

Co-operation within the consortium:

- Collaborative work within the consortium was done using **BSCW** (Basic Support for Cooperative Work**).**

Code development:

- Code development was done using a **CVS** (Concurrent Version System) repository.

2.3 Project meetings and research stays

Project meetings:

1) Project meeting at UPC in Barcelona, Spain.
Date 03/2002.
Objective: Kick-off meeting.
Participants from UF-IIG: Torsten Eymann, Michael Reinicke.
Participants from UPC: Oscar Ardaiz, Luis Diaz, Felix Freitag, Roc Messeguer, Leandro Navarro, Dolors Royo.

2) Project meeting at UF-IIG in Freiburg, Germany.
Date: 08/2002.
Objective: WP1 evaluation.
Participant from UPC: Oscar Ardaiz.
Participants from UF-IIG: Torsten Eymann, Michael Reinicke.

3) Research stay at UPC in Barcelona, Spain.
Date: 10/2002.
Objective: Collaborative work and documentation on WP2.
Participants from UF-IIG: Michael Reinicke.
Participants from UPC: Oscar Ardaiz, Pau Artigas, Felix Freitag, Leandro Navarro.

4) Research stay at UPC in Barcelona, Spain.
Date: 01/2003.
Objective: Collaborative work and documentation on WP2 and WP3.
Participants from UF-IIG: Michael Reinicke.
Participants from UPC: Oscar Ardaiz, Pau Artigas, Felix Freitag, Leandro Navarro.

5) Research stay at UF-IIG in Freiburg, Germany.
Date: 01/2003
Objective: Collaborative work and documentation on WP2 and WP3.
Participants from UF-IIG: Pau Artigas.
Participants from UPC: Michael Reinicke.

6) Project meeting at UF-IIG in Freiburg, Germany.
Date: 02/2003
Objective: Collaborative work and documentation on WP3.
Participants from UF-IIG: Felix Freitag.
Participants from UPC: Michael Reinicke.

3. Global results

In this chapter we resume the technical achievements and results obtained in WP1, WP2, and WP3, described in the deliverables D1, D2, and D3.

3.1 Application layer network simulator

This section resumes the achievements and results of WP1 described in the deliverable D1.

The application layer network simulator developed in the project can be downloaded from http://research.ac.upc.es/catnet.

3.1.1 Simulator overview

The CATNET application layer network (ALN) simulator is implemented on top of the JAVASIM network simulator. It can be configured to simulate the characteristics of a specific ALN, such as a content distribution network or peer-to-peer network. Different agent types can be instantiated, namely clients, resource agents, and service agents. Network resources to be allocated encompass service access, bandwidth and storage. The simulation is built on a TCP/IP network model supported by JAVASIM. JAVASIM describes the generic structure of a node (either an end host or a router) and the generic network components, which can both be used as base classes to implement protocols across various layers.

3.1.2 General simulation setup

The CATNET application simulates two main control mechanisms for network coordination: a Baseline control mechanism and a Catallactic control mechanism. The Baseline control mechanism computes the resource allocation decision in a centralized service/resource provider. The Catallactic control mechanism has the characteristic that its resource allocation decisions are carried out by self-interested agents with only local information about the environment. Each agent has its own negotiation strategy module. The following class types are defined:

- Client: A computer program on a certain host, which needs access to a web service to fulfill its design objectives. The Client (C) tries to access that "service" at an arbitrary location within the computer network, use it for a defined time period, and then continues with its own program sequence. Client programs run on a connected network "resource".
- Service: An instantiation of a general application function, embodied in a computer program.
- Service Copy: One instance of the "service". The Service Copy (SC) is hosted on a "resource" computer, which provides both storage space and bandwidth for the access of the service.
- Resource (R): A host computer, which provides a limited number of storage space and access bandwidth for service transmission. Resources are connected via network connections defined in a topology script.
- Network Connections: These connections are intended to be of equal length and thus of equal transmission time and costs.

3.1.3 Message Flows in the Baseline Model

In order to simulate different control mechanisms, we consider the Baseline system as a special case of the generic Catallactic control mechanism. Through configuration in input scripts of the simulator, different behavior of the simulator can be set up. As a consequence, the comparison of simulation results should become easier to control and the development efforts focus on a single, generic system.

The centralized Baseline mechanism employs a dedicated service coordinator (the master service copy, MSC), which is known to the individual Service Copies. The client broadcasts a "request_service" message on its network connections. Either the receiving resource (R) provides a Service Copy (SC) of the requested type or not. If a SC is available, the resource routes the request to that Service Copy, adding its costs for storage and bandwidth consumption. The SC directs the request to the Master Service Copy (MSC), provided with information about costs and the amount of the message's hop counter, i.e. the number of passed resources, indicating the distance to the requesting client.

Resource hosts (R) forward the received request – independent of the successful detection of the service – to their neighboring resource hosts, increasing the message's hop counter. Using this procedure, all adjacent resources will be inquired. If the hop counter exceeds a given number, the message is discarded.

The MSC receives all the information from the R/SC pairs, is able to compute the costs of providing a service and sends back an accept/propose message revealing the "cheapest" SC to the client. In addition, it informs the selected R/SC pair. The resource allocates a timeslot and the SC provides the service. After that, the client sends the formerly agreed reward to the SC, which redirects the payment share for bandwidth and storage to its R host.

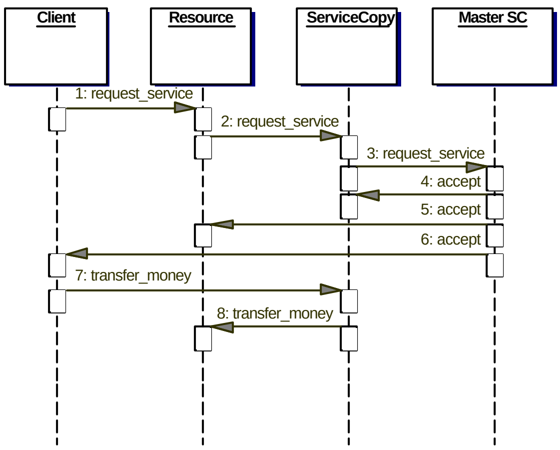

Figure 1. Money and Message Flows: Baseline Approach.

3.1.4 Message Flows in the Catallactic Model

The Catallactic control mechanism has the characteristic that its resource allocation decisions are carried out by decentralized SCs with only local information about the environment.

Again, the clients send out a "service_request" message on its network connections. The receiving resource forwards the message to the neighboring resource hosts. If the resource holds a SC of the requested type, the resource routes the request to it. In order to return a valid quote to the client, the SC has to inquire the resource about the provisioning costs by initiating a negotiation for bandwidth costs. A successful negotiation on this behalf allows the SC then to negotiate for the price for the provision of the service with the client.

The client stores all incoming proposals in its inbox and subsequently negotiates for service access. It is guided in its strategy by the subjective market price, which is computed from all price quotes the agent receives from the SCs, regardless of the particular sender. If the initial offer price does not match within an interval around the market price, the negotiation will be discontinued. Otherwise, the agents will engage in a bilateral alternating offers protocol till acceptance or final rejection of the offer.

An accept message from the client lets the SC confirm both negotiations (with the resource for bandwidth and with the client for service provision). The resource reserves bandwidth and the contracts are sealed. The service provision is mirrored by the according money flow. On the other hand, a reject message from the client immediately stops further negotiation and initiates a reject message from the SC to the resource.

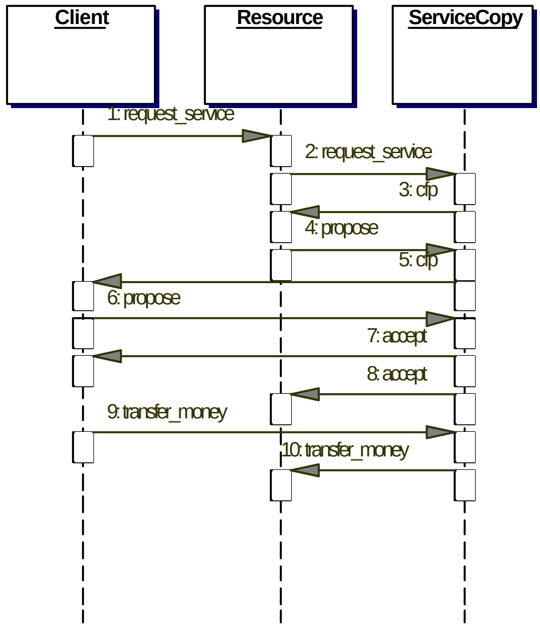

Figure 2. Money and Message Flows: Catallactic Approach.

To maximize utility, the agents will change their initial offer prices, starting with demand and supply prices given in an input data script, according to the following scheme: Rs and SCs as sellers will lower their offer price by one money unit if the negotiation was not successfully finished. They will raise their initial price by one money unit after an offer has been accepted. The clients and SCs as buyers will change their initial prices vice versa.

3.2 Simulation of the application layer network

This section resumes the achievements and results of WP2 described in the deliverable D2.

The application layer network is build on top of a physical network topology. The physical network topology is specified in the input script of the simulator. The topology could be random or having a determined structure specified by the user. In Figure 3 we show one of the physical topology, which we used in the experiments, and which was captured with NAM (Network Animator).

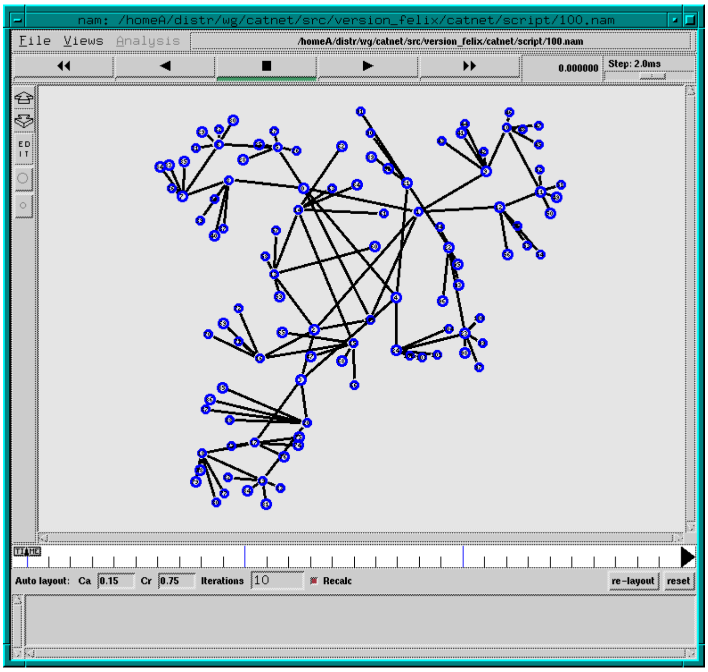

Figure 3. Example of a network topology, approx. 100 nodes.

On top of the physical nodes, a number of different software agents are created, which form the application layer network. The software agents are Clients, Service Copies, and Resources. Each node can host Clients, Resources, and/or Service Copies. A node can host several agents or none at all. In the latter case, the node just acts as a router.

The application layer network formed by these agents is varied in the experiments. In order to simulate the node density of the application layer network, we vary the number of Resource agents and Service Copies available to Clients. In order to simulate the dynamics of the application layer network, we connect and disconnect the available Service Copies during the simulation.

For each agent, particular data such as the capacity of the Resources can be specified in the initialization of the simulator. The capacity of the resources is high in the low node density scenario, and low in the high node density scenario, due to the correspondence with content distribution networks and Peer-to-Peer networks, respectively. The initial prices of Clients, Service Copies, and Resource agents are specified in the initialization of the simulator, also the initial budget of the Clients.

The type of control mechanism is another parameter specified in the setup of the simulator. The main control mechanisms implemented in the simulator are the Catallactic and the Baseline approach. The modular design of the simulator, however, also allows testing variations of them to investigate the effect of different parameters in each control approach.

Real world distributed applications like multimedia content distribution networks (for instance AKAMAI), Grid implementations, and Peer-to-Peer systems (for instance Gnutella) can be characterized in a simplified form by a number of a few common features, which inspired the design of the application layer network implemented in the simulator. Though different in many particular mechanisms, these real world applications can be mapped into the two-dimensional design space given by

1) the node dynamics and
2) the node density of the application layer network, see Figure 4.

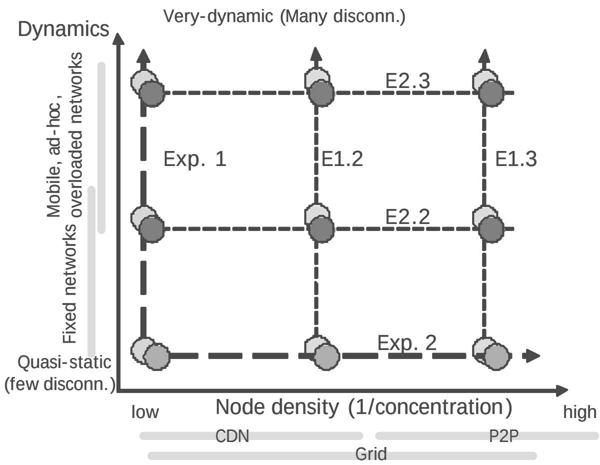

Figure 4. Illustration of the main experiments in a two-dimensional design space.

Node dynamics measures the degree of availability of service-providing nodes in the network. Low dynamics mean an unchanging and constant availability; high dynamics are attributed to a network where nodes start up and shut down with great frequency.

Node density measures the relation of resource nodes to the total number of network nodes. The highest density occurs when every network node provides the described service to others; the lowest density is reached if only one resource node in the whole network exists.

By varying node dynamics from null to medium to high, and node density from low to medium to high, each of the two control mechanism is simulated with 9 scenarios, as illustrated in Figure 4, which leads to 18 basic experiments.

The obtained traces during the simulation are used to compare the performance of the Catallactic and the Baseline system in the different scenarios.

3.3 Experimental results

This section resumes the achievements and results of WP3 described in the deliverable D3.

The base experiment provides results for 3x3=9 combinations for both Catallactic and Baseline of dynamics and density. This section describes the experimental results with the setting of the corners of the design space. The values presented in the following figures depict the difference between Catallactic and Baseline and have been normalized with respect to Baseline. This representation allows a fast visual representation of the change in behavior with respect to Baseline. For instance, the value for SWF in % is computed:

$$SWF\% = \frac{(SWF_C - SWF_B)}{SWF_B} * 100 \qquad (1)$$

3.3.1 Quasi-static scenario

This type of scenario resembles content distribution or Grid Computing applications which have low node dynamics, so that resource nodes are highly available and permanently connected to the network, and low node density so that resource nodes are few in terms of the total nodes which form the application layer network.

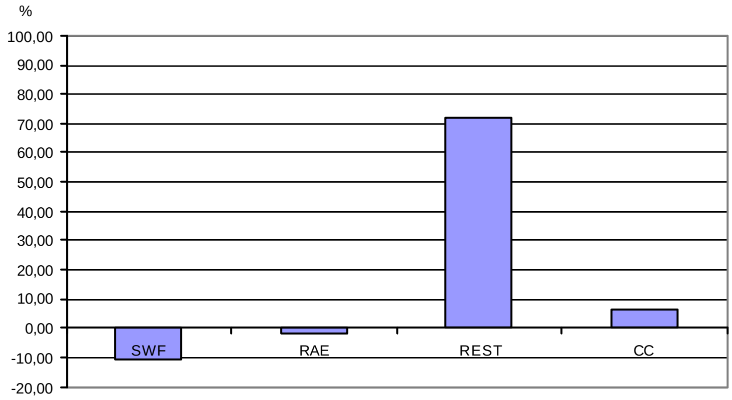

Figure 5. Results of the Quasi-static scenario.

The results of the SWF show that the Baseline model outperforms Catallactic by 10%. In terms of RAE and CC, Baseline is better by a small margin. Baseline also achieves a faster response time (REST).

3.3.2 Highly-dynamic scenario

Peer-to-Peer (P2P) networks can be described as networks with high node dynamics and high node density. In this case, high node dynamics is given by the high level of connection and disconnection found in P2P networks. High node density is illustrated by the fact that each peer carries out the function of a C, SC and R.

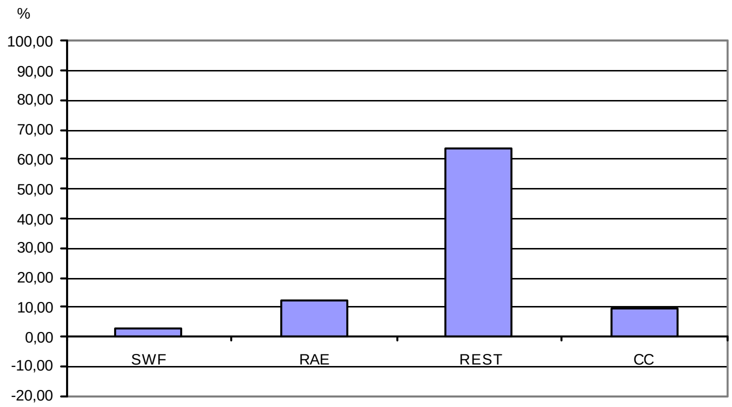

Figure 6. Results of a highly-dynamic scenario.

The SWF value and the resource allocation efficiency (RAE) are slightly better in the Catallactic case. Response time (REST) and communication cost (CC) are higher in the Catallactic system.

3.3.3 Scenario low node density

Low density (which also means high resource concentration) may correspond to web service scenarios with one or a few web or computing servers. At the same time, the dynamics of the network is quite high, which reflects to unreliable long distance connections between the nodes.

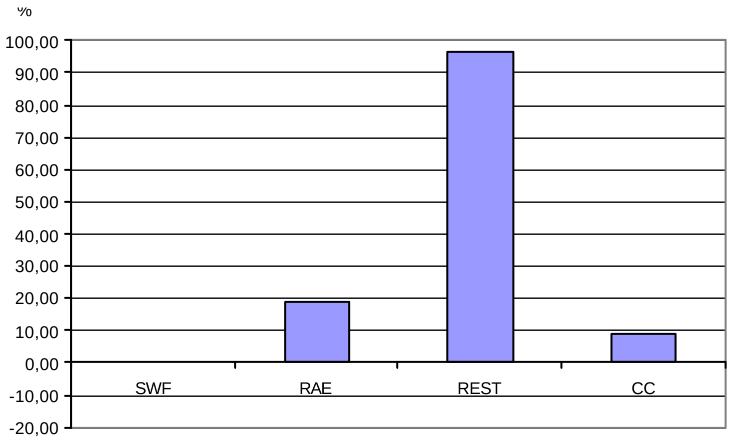

Figure 7. Results of low-node density experiment

The social welfare utility (SWF) does not differ but the Catallactic model has a higher RAE. The response time (REST) is significantly longer for the Catallactic model probably as an effect of the dynamic demand.

3.3.4 Scenario high node density

A high density scenario may correspond to a P2P network, an extreme case of Grid, clustered processors in a local network or parallel supercomputing. Overall, the dynamics of the scenario is low, so the links between the nodes are stable. Once paths and routes have been found, they need not change.

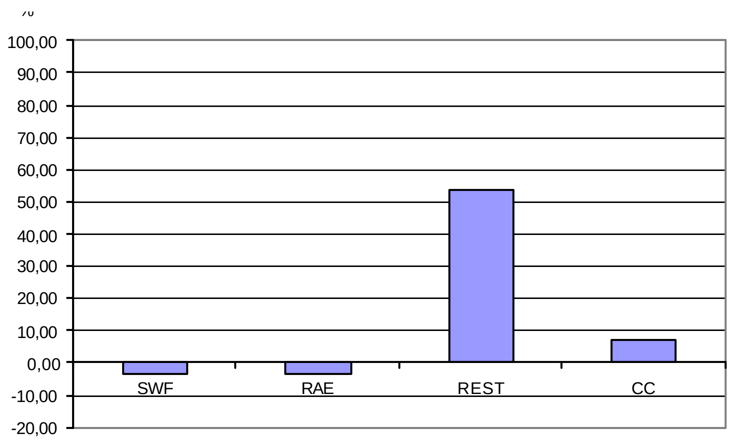

Figure 8. Results of the high density scenario.

Both models achieve similar results in SWF and RAE. The response time (REST) and the CC are higher in the Catallactic model, as expected.

3.3.5 Experimental results by criterion

In this section we resume the results from deliverable D3 referring to the 9 scenarios. Different to D3, where the performance of both systems is shown in one single graphic for comparison, we present here the results in separate graphics, which allows a better observation of tendencies.

Social welfare (SWF):

Figure 9 shows the value of the SWF parameter in the different scenarios. The behavior of the Catallactic system changes rather smoothly from one scenario to the other, showing a straight decrease with increasing density and dynamics, while the gradient in Baseline system is higher and less regular. For Baseline, a decrease of dynamics leads to a decrease of SWF, while for density an unexpected extremum exists in the medium density regime. With the help of sensitivity experimentation, it could be shown that this effect can be influenced by the MSC Allocation Time. Concerning only the current parameter setting and network topology, the Baseline SWF seems to be correlated to the MSC Allocation Times as shown in the respective annex chapter.

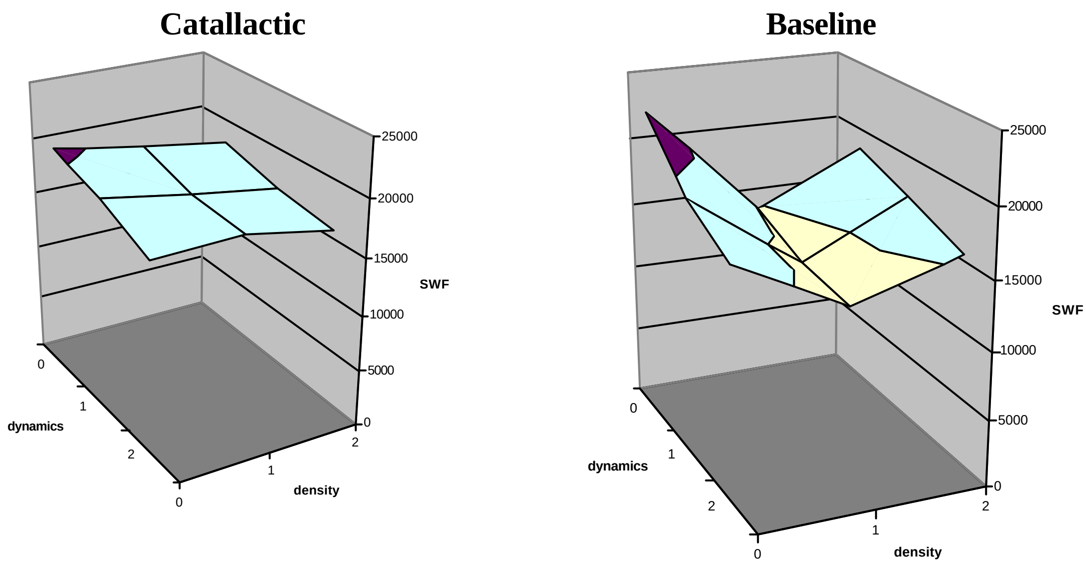

Figure 9. SWF of the Catallatic and Baseline in the 9 scenarios.

Resource Allocation Efficiency (RAE):

In Figure 10 the value of the RAE parameter in all considered scenarios is shown. It can be seen that the RAE of the Catallactic system changes smoothly from one scenario to the other. The RAE achieved in the high density scenarios is almost independent of the dynamics, but shows some negative correlation to the factor density. The RAE of Baseline is less regular. Higher dynamics decrease the RAE achieved. Observing density, the same behavior is exhibited like in SWF. Obviously a minimum of RAE can be noticed in the medium density scenarios which is suspected to be due to the same explanation.

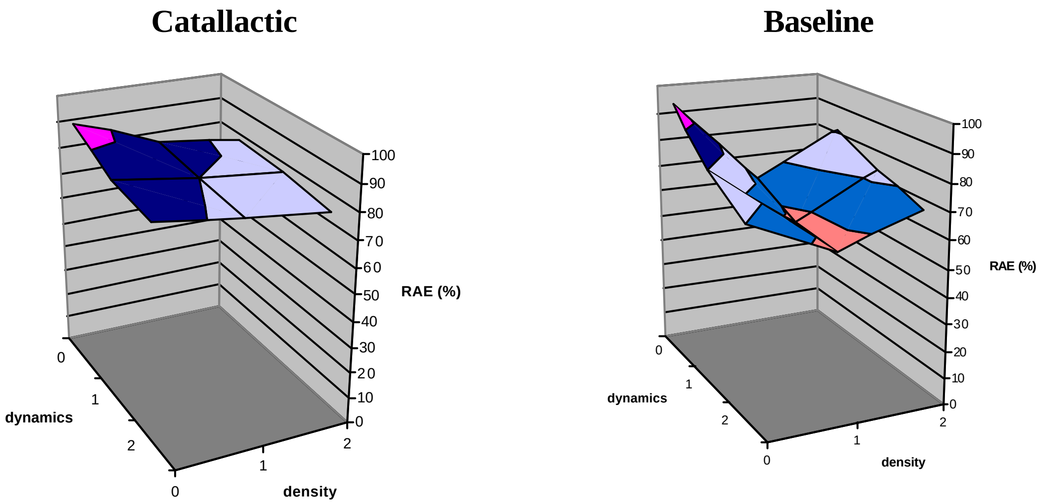

Figure 10. RAE of the Catallactic and Baseline in the 9 scenarios.

Response Time (REST):

Figure 11 displays the value of the response time (REST) in the different scenarios. The response time of the Catallactic system is influenced both by the dynamics and density of the simulated scenario. In this model, a minimum is shaped at medium density and a maximum at medium dynamics.

In Baseline, the response time grows with the increase of node density and shows independence to dynamics.

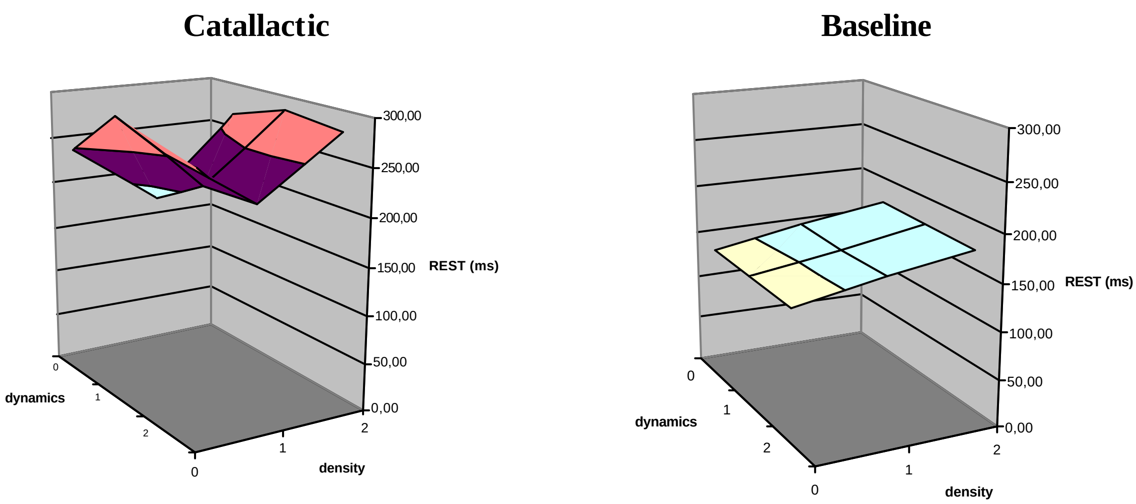

Figure 11. REST of the Catallactic and Baseline in the 9 scenarios.

Communication Cost (CC):

In Figure 12 the value of the CC parameter in the different scenarios is presented in comparison to dynamics and density. It can be pointed out that the CC both of the Catallactic and the Baseline system increases with higher node density but does not change by altering dynamics. Catallactic shows a steady increase, whereas in Baseline model, an increase firstly comes up when changing from medium density to high density.

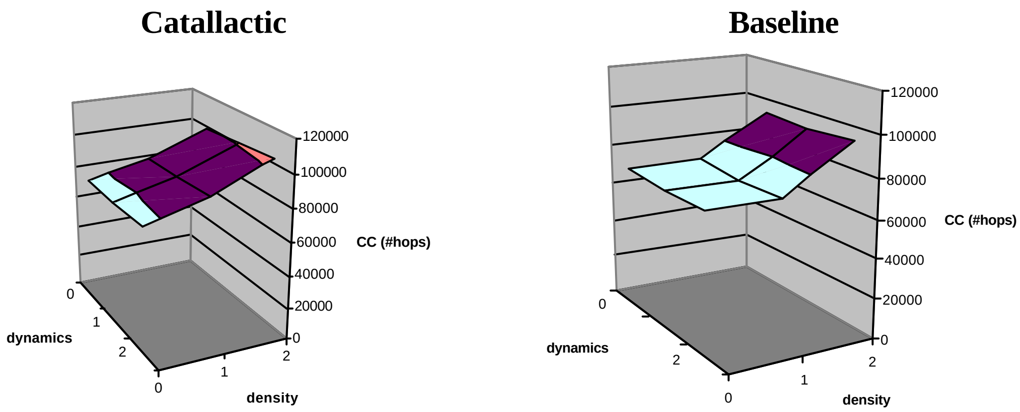

Figure 12. CC of the Catallactic and Baseline in the 9 scenarios.

3.3.6 Conclusions[1]

Comparing the results of the four scenarios, we can observe the following common behavior in the experiments:

- The response time (REST) is significantly longer in the Catallactic model.
- The communication cost (CC) is always slightly higher in the Catallactic model, which means that more messages are exchanged and naturally a higher bandwidth consumption.
- SWF (social welfare) and RAE (resource allocation efficiency) are better in the Catallactic model for the high dynamic and high node environment, while SWF and RAE are better in the Baseline model for the low dynamic and low node environment.
- The behavior of SWF and RAE of the Catallactic model changes smoothly over all scenarios, appearing to be less influenced by each setting of the node dynamics and node density. In the Baseline model, the node density influences SWF and RAE differently. In the displayed scenarios it can be observed that the increase of dynamics leads to a reduction of the SWF and RAE whereas an increase in density leads to a minimum in medium density.

[1] A detailed comparison and evaluation is given in deliverable D3.

3.4 Potential impact of the results

The performance results we obtained in the simulations are favorable for the Catallactic model in scenarios with high node density and high dynamics. We saw in the scenario with high node density that the performance appears to be rather robust to the node dynamics.

Robustness to dynamics could be an interesting feature for applications, which operate in highly dynamic environments and which require reliable service provision. The Catallactic coordination could be an alternative mechanism for such application cases when further explored. It could influence in the design, implementation and control of application layer networks formed in the framework of Grids and Peer-to-Peer systems, but also in service provision frameworks with a mobile context.

3.5 Exploitation of the results

The research work of this project is of a generic nature, but it could ultimately underpin a wide range of application areas. Building an application layer network composed of autonomous decentralized entities for environments in which the network structure and nodes change over time, may lead to a variety of business opportunities in the longer run, if further developed and researched.

Our experiments provided results on the performance of Catallactic coordinated systems. The confirmation of this predicted performance in real Grid and Peer-to-Peer environments is an open research issue, which shall be pursued in future work.

4. Public dissemination

Project website:

The consortium set up a project web site at http://research.ac.upc.es/catnet/. The simulator and the publications can be downloaded from the website.

Participation in conferences/workshops & public visibility:

- We participated in the 2nd Workshop on Global and Peer-to-Peer Computing on Large Scale Distributed Systems in the 2nd IEEE/ACM International Symposium on Cluster Computing and the Grid, 21–24 May 2002, Berlin, Germany.
- Participation in the 4th European Agent Systems Summer School on Autonomous Agents 2002 in Bologna. Monday, 8 July 2002- 12 July 2002.
- We participated in the 1st European Across Grids Conference, 14–16 February 2003, Santiago de Compostela, Spain.

- We participated in the 2nd International Workshop on e-Learning and Grid Technologies, organized by the Learning GRID of Excellence Working Group (LeGE-WG), 03-04.03.2003 in Paris.
- We will participate in the Agent-based Simulation 4 Workshop in April 2003, Montpellier, France.
- We will participate in the Workshop on Agent based Cluster and Grid Computing within the 3rd IEEE/ACM International Symposium on Cluster Computing and the Grid in May 2003, Tokyo, Japan.

Publication of papers:

1. O. Ardaiz, F. Freitag, L. Navarro, T. Eymann, M. Reinicke. CatNet – Catallactic Mechanisms for Service Control and Resource Allocation in Large Scale Application-Layer Networks. *Workshop on Global and Peer-to-Peer Computing on Large Scale Distributed Systems*, *2nd IEEE/ACM International Symposium on Cluster Computing and the Grid*, May 2002, Berlin, Germany.

2. T. Eymann, M. Reinicke, O. Ardaiz, P. Artigas, L. Díaz de Cerio, F. Freitag, R. Messeguer, L. Navarro, D. Royo. Decentralized vs. Centralized Economic Coordination of Resource Allocation in Grids. *1st European Across Grids Conference*, February 2003, Santiage de Compostela, Spain.

3. T. Eymann, M. Reinicke, O. Ardaiz, P. Artigas, L. Díaz de Cerio, F. Freitag, R. Messeguer, L. Navarro, D. Royo. Exploring decentralized resource allocation in application layer networks. *Agent-based Simulation 4*, April 2003, Montpellier, France.

4. T. Eymann, M. Reinicke, O. Ardaiz, P. Artigas, F.Freitag, L. Navarro. Decentralized resource allocation in application layer networks. *Workshop on Agent based Cluster and Grid Computing. 3rd IEEE/ACM International Symposium on Cluster Computing and the Grid*, May 2003, Tokyo, Japan.

Other related activities:

- We were invited and participated in an EoI for FP6 – Agent based Grid Computing.
- Leandro Navarro was invited to participate in the Program Committee of the 3rd International Workshop on Agent based Grid Computing within the CCGrid Conference (a joint IEEE/ACM conference) in Tokyo, Japan in May 2003.

5. Conclusions

The achievements of the CATNET project have been:

- The development of an experimental simulator for investigating different allocation mechanisms.
- The definition of a two-dimensional problem space, defined by dynamics and density, which can be used to categorize application layer networks.
- The conduction of an experimental evaluation study of two coordination mechanisms.
- The comparison and evaluation of the two coordination mechanisms.

With the results of the experiments, some hypotheses have been confirmed, some rejected. In the studied scenarios, the Catallactic model became superior over the Baseline model with increasing dynamics. It also showed less sensibility to the dynamics. Its observed behavior changed rather smoothly over the different scenarios. Catallaxy achieved this result at the expense of higher communication cost and lower response time. We believe that a robust behavior in dynamic environments is a feature interesting for several applications.

Our design space was described mainly by the node dynamics and node density. However, a larger design space could be considered for both systems. Other parameters are, for instance, the effect of scale on the coordination mechanisms, the influence of particular characteristics of the demand queue, design parameters of the Baseline system to handle highly dynamic environments, and parameters of the strategy used in the Catallactic coordination to determine prices. We have carried out several sensibility studies along with the base experiments for validation and to evaluate the influence of other parameters in the obtained results.

The experimental approach followed in this project to achieve an evaluation of the Catallactic model required the definition of an experimental framework. We observed how the Catallactic and Baseline systems behaved in the different scenarios, how they differed in particular cases and what their performance tendencies were on changes of different parameters. The transfer and application of these findings to real Grid and Peer-to-Peer environments is an open research issue, which shall be pursued in future work.