Diplomarbeitspräsentationen der Fakultät für Informatik

Naming and Communication Management for MozartSpaces

Hannu-Daniel Goiss

FAKULTÄT FÜR !NFORMATIK

Masterstudium:
Software Engineering/
Internet Computing

Technische Universität Wien
Institut für Computersprachen
Arbeitsbereich: Programmiersprachen und Übersetzer
BetreuerIn: Ao. Univ.-Prof. Dr. Dipl.-Ing. eva Kühn

Locating information in the Internet is a demanding task because of the information flood. In telematics there is the challenge of finding the right data, due to the high amount of users and the high amount of information. This Master's Thesis introduces a "Naming Layer" for **MozartSpaces**, which is a middleware based on the Space Based Computing paradigm.

Space Based Computing is a modern form of communication between various participants in a computer network. The participants communicate by using objects, which they store in a common communication area.

One of the challenges is to locate objects, which have been created by other participants. This can be done by using the object identifier, if it is known in which communication area the object is located. The "Naming Layer" has the purpose to allow the location of objects that are located in various communication areas. A "Communication Layer" is introduced as well, which allows the use of various protocols for data communication.

MozartSpaces is a space based computing middleware based on the eXtensible Virtual Shared Memory (XVSM) standard.

The basic principle of space based computing is that the communication happens via a space and not directly between the participants. The space is a shared memory between the participants of the space, in which objects can be written and read. Communication between two participants happens by writing objects into the space and reading objects from the space.

TRANSPORT MANAGER

The implementation has been divided into two parts:
- **Transport-Manager**: interface for the user
- **Transport-Units**: implemented using the various technologies. They provide the actual functionality.

Transport-Manager functionality:
- manages various transport protocols dynamically
- protocol negotiation

LOOKUP MANAGER

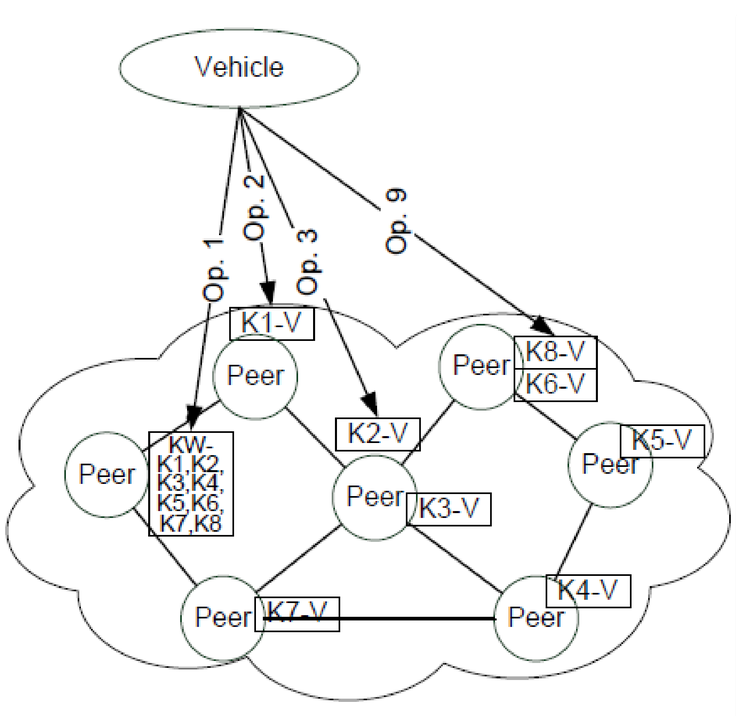
The implementation has been divided into two parts:
- **Lookup-Manager**: interface for the user
- **Lookup-Units**: implemented using the various technologies. They provide the actual functionality.

Lookup-Manager functionality:
- Management of various Lookup-Units
- publish containers
- unpublish containers
- lookup containers
- list all published containers

XVSM is slower than the other technologies, but it provides more complex query mechanisms, which make up for the slower performance.

Lookup-Unit	average publish	min. publish	max. publish	average lookup	min. lookup	max. lookup	average unpublish	min. unpublish	max. unpublish
XVSM	1040.35	797	1219	639.22	593	719	1029.32	906	2422
LDAP	12.67	0	63	1.1	0	16	19.16	15	32
Gnutella	0.8	0	16	34.12	31	96	0.9	0	16
FreePastry	24	24	24	23.3	23	24	23	23	24

Lookup-Manager: Comparison between the Lookup-Units

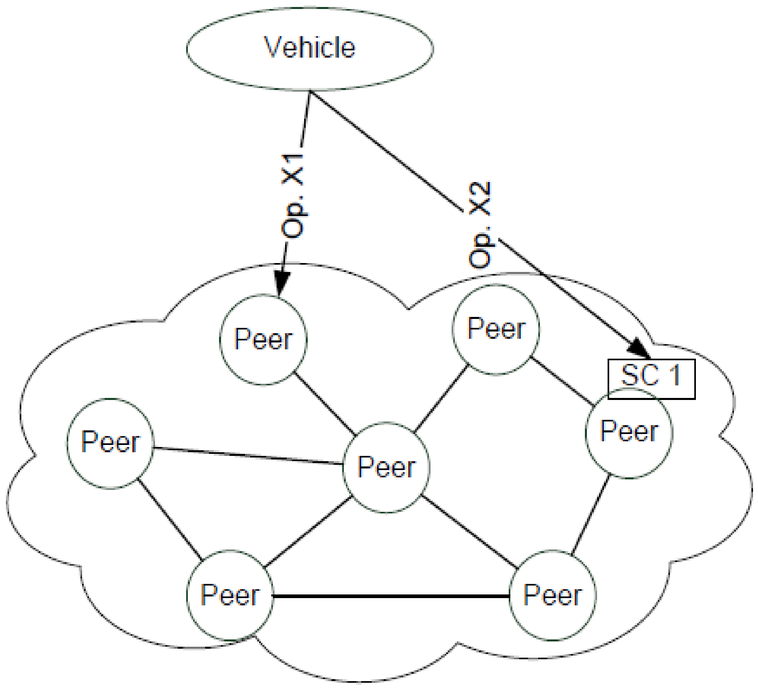


Complexity of a lookup using solely a DHT

The following scenario illustrates the difference of the complexity between a lookup of a specific value solely performed on a DHT and a combination of a DHT and Space Based Containers.

Performing a lookup using solely a DHT requires the following actions:
1. Lookup of a well-known DHT, which stores the keys of all entries.
2. Lookup of **all** found DHT keys to retrieve their entries and compare them to the searched strings.

Performing a lookup using a combination of DHT and space based computing requires the following actions:
1. Lookup of a the container name to acquire the address of the space based container.
2. Read operation on the container to acquire the searched string.



Complexity of a lookup using a combination of a DHT and Space Based Computing

Kontakt: h.goiss@gmail.com