

Java language bindings for extensible virtual shared memory: Design und implementation

Masterstudium:
Software Engineering/Internet
Computing

Klemens Kraus

Technische Universität Wien
Institut für Computersprachen
Arbeitsbereich: Programmiersprachen und Übersetzerbau
Betreuerin: A.o. Univ. Prof. Dr. Dipl.-Ing. eva Kühn

Overview

Extensible virtual shared memory – XVSM

Follows the space based computing paradigm

- Decouples communication of independent processes
- Provides features to coordinate activities
- Defines easy to use data structures

Goals of the Thesis

- Evaluate existing standards
- Create a high level Application Programming Interface (API) for Java
- Compare characteristics of developed API to competitive systems

Design Considerations

How to design a successful API?

- Design for change
 - Systems of this age tend to change quickly
- Make it easy to use
 - Identify common use-cases
 - Consult with users
- Integrate with existing APIs
- Create a thorough documentation
- Provide utility functionality

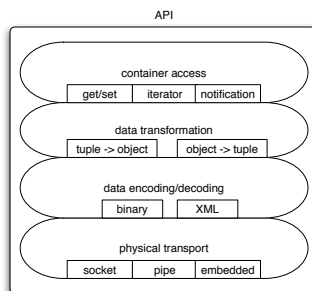
System features

- Integrates into Java collection classes
- Allows standardized access methods
- Hides underlying complexity from the user

Overview

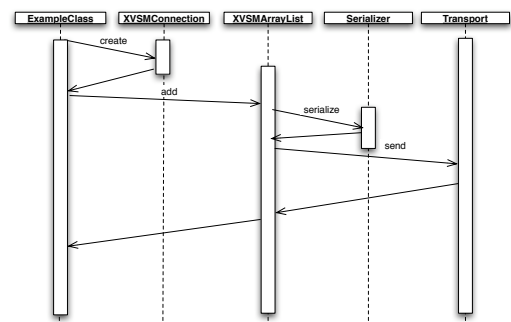
The Developed System

- Uses a layered structure
 - Container access
 - Data transformation
 - Data encoding & decoding
 - Physical transport
- Hides low level functionality
- Allows customizability of all layers



System overview

Example Workflow



Workflow that shows complexity hidden by the API

Comparison

Feature Comparison

- The industry's state-of-the-art systems compared to the presented system

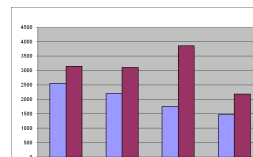
Feature	JavaSpaces	CORBA	CORSO	XVSM
Coordination Based System	X		X	X
High level interface		X		X
Integrated high level data structures				X
Data Replication Strategy	Multiple Servers / Peer to Peer	Single Server	Peer to Peer	No implementation yet
Java Language Integration	X	X		X
Platform and Language independence		X	X	X

State-of-the-art in comparison

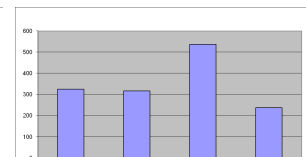
Evaluation & Conclusion

Evaluation

- Measures two characteristics
 - Lines of code
 - Runtime



Wall-time



Lines of Code

Concluding the Thesis

- Almost identical performance compared to the low level API
- Judging the two test characteristics XVSM and the developed API is comparable or better than other state-of-the-art systems