

TECHNISCHE
UNIVERSITÄT
DARMSTADT

Seamless Service Provision in P2P Service Overlays

KuVS 5th Workshop (Fachgespräch) on
Next Generation Service Delivery Platforms
October 11, 2011

Kamill Panitzek

Muhammad Ikram

Immanuel Schweizer

Max Mühlhäuser



DFG Deutsche
Forschungsgemeinschaft

This work has been funded by the DFG research unit 733 QuaP2P



Introduction

- Peer-to-peer (P2P) systems
 - Large-scale decentralized systems
- P2P systems used to discover and distribute data or data streams
 - File sharing systems (e.g., eMule)
 - Voice over IP (VoIP) communication (e.g., Skype)
 - Video streaming (e.g., PPLive)
- Next generation P2P systems: P2P service overlays
 - Distribution and execution of applications
 - Composition of services



Introduction – cont.

- Heterogeneity of devices
 - Services are not executable on any device
- Device mobility requires mobile applications
 - Services and applications must be transferable during runtime
- P2P service overlays pose several problems
 - Focus on seamless service provision in this presentation
 - Meaning service provision during migration process



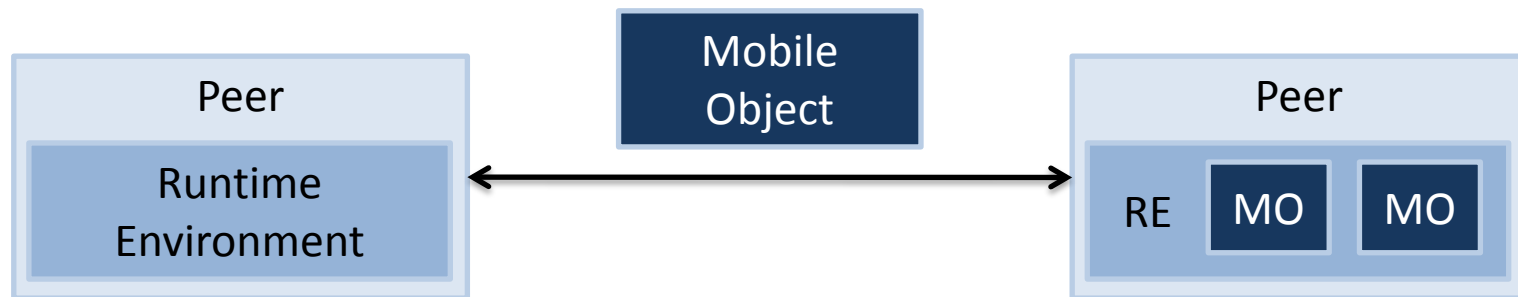
CONTENT

- Introduction
- P2P Service Overlays
- Seamless Service Provision
- Framework Implementation
- Summary and Future Work



P2P Service Overlays

- Interconnected peers
 - Creating P2P Service Overlay
- Peers provide resources
 - CPU, memory, storage
 - Energy supply
 - Bandwidth
 - Runtime environment
- Mobile objects
 - Data items or streams
 - Services or applications
- Mobile objects are stored in or executed on runtime environment





Basic requirements

- Functional requirements
 - Distribute and discover mobile objects
 - Execute services or applications

- Non-functional requirements
 - Availability of mobile objects
 - Robustness to dynamic changes and peer mobility



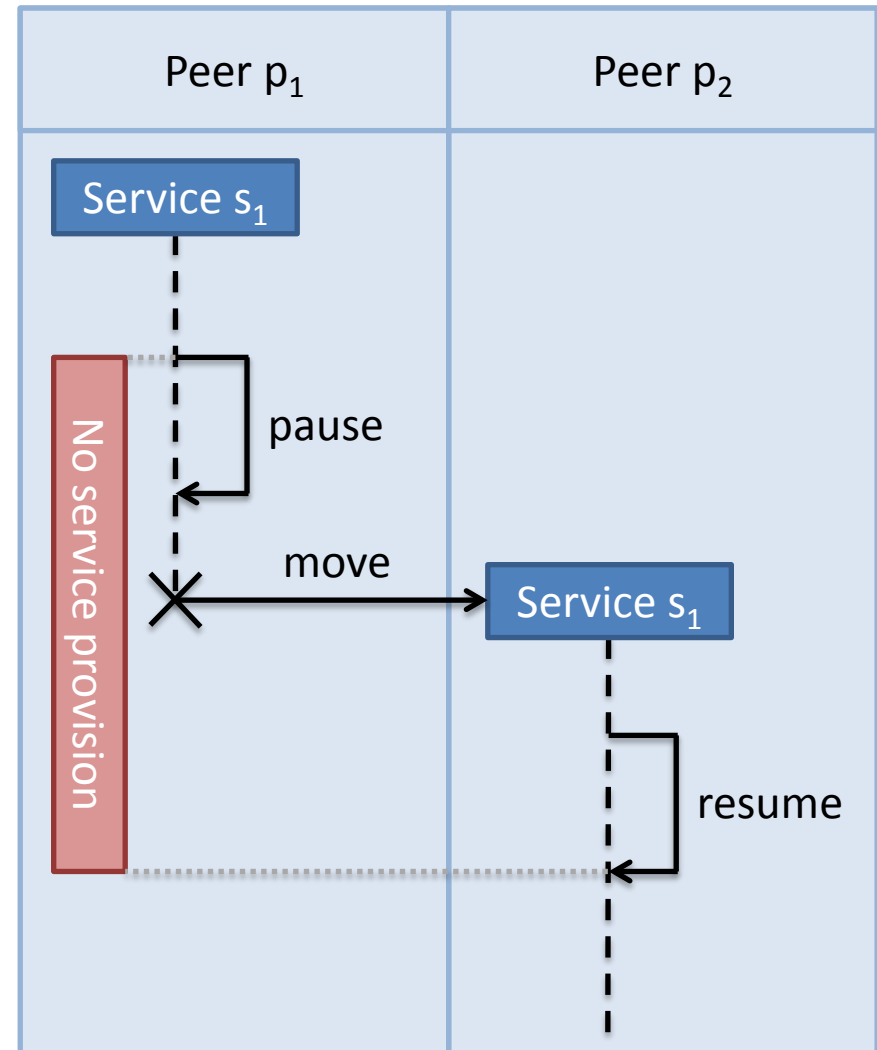
Components and Problems

- Distribute and discover mobile objects
 - Service registry using distributed hash table (DHT), for example
- Execute services or applications
 - Matching of mobile object requirements to peer resources
- Availability of mobile objects
 - Replicate popular mobile objects using a replication mechanism
- Robustness to dynamic changes and peer mobility
 - Prevent peer overloading using a pro-active load balancing mechanism
 - Seamless service provision using code migration and persistent connections



Seamless Service Provision – Problem

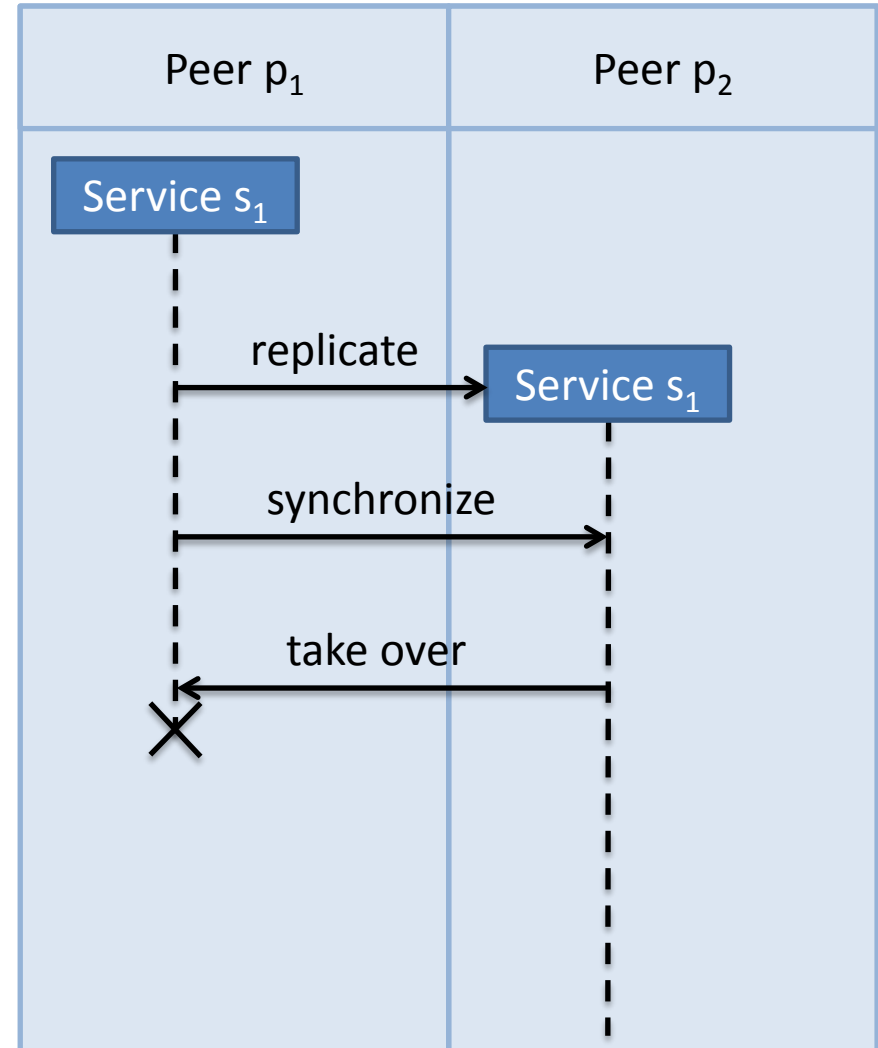
- Migrate services during runtime
 - Including current execution state
- Usually service is interrupted during migration process
- Avoiding interruption of service provision is crucial in many cases





Seamless Service Provision – Solution

- Replicate service to another peer
 - Copy code and execution state
- Synchronize execution states
 - Propagate state changes to new host
- New host takes over
 - Inform old host and all clients
- Old host shuts down the service





Interactive Services

- Interactive Services
 - Services client peers are connected to and constantly are interacting with
- For example, a voice communication service
 - Voice packets are sent from clients to communication server
 - Server then distributes voice packets to participating clients
 - Common approach in contemporary voice communication software
- Communication must not stutter, interrupt, or disconnect
 - Only transferring execution state is not sufficient
 - Reestablish connections between clients and host after migration



Framework Implementation

- Implemented framework for P2P service overlays
 - Supporting seamless service provision
- Based on Java and MundoCore¹ middleware [1]
 - Realization of code migration
- Used the Chord DHT [2] OpenChord²
 - Realization of decentralized service registry
- Persistent socket connections similar to MobileSocket [3]
 - Further enhancements to the approach
 - TCP channel for control messages and UDP channel for data transmission

¹<http://www.mundocore.org/>

²<http://open-chord.sourceforge.net/>



Persistent Socket Connections

- Migrating socket connections between clients and host
 - Two channels, one data channel and one channel for control messages
 - Connect new host with clients after migration of service
 - Propagate all control messages from old to new host
 - After take-over all data and control messages are sent to new host
 - Delayed packets are forwarded to new host
- Framework abstracts from this functionality
 - Client software transmits control or data messages w/o noticing migration
- Implemented voice chat service prototype using the framework
 - Voice encoding using JSpeex³, a Java port of the Speex speech codec

³<http://jspeex.sourceforge.net/>



First Empirical Study

- Two laptops interconnected over LAN and WiFi
 - Creating P2P service overlay using OpenChord
- Used internal stereo mix to record music
 - Compressed with highest quality available (44kHz, 10/10 Speex quality)
 - Play back using second laptop
- Service was migrated between both laptops
 - No interruptions during migration process noticeable
 - Network bandwidth utilization low during migration (~ 36kBytes/s for 2sec)
- Intensive and exhaustive evaluation started



Summary

- Description of P2P service overlays: next generation P2P systems
- Introduction to several problems
 - Focusing on seamless service provision in P2P service overlays
- Framework for P2P service overlays
 - Combination of different existing tools and own approaches
- Implementation of seamless voice chat service prototype
 - No interruptions during migration process noticeable
 - Evaluation in progress



Future Work

- Further enhancements to the framework
 - Improve migration process (e.g., compression of execution state)
 - Using Kademia DHT instead of Chord
- Tackle next problems posed in this presentation
 - Decentralized service discovery and invocation
 - Decentralized matching algorithm
 - Replication
 - Pro-active load balancing
- Focus on availability of services specifically in P2P based games
 - Discussed by Muhammad Ikram in next presentation



Thank you!



TECHNISCHE
UNIVERSITÄT
DARMSTADT

Thank you for your attention!
Questions?



Bibliography

- [1] E. Aitenbichler, J. Kangasharju, and M. Mühlhäuser, “MundoCore: A light-weight infrastructure for pervasive computing”, *Pervasive and Mobile Computing*, pp. 332–361, 2007.
- [2] I. Stoica, R. Morris, D. Karger, M. F. Kaashoek, and H. Balakrishnan, “Chord: A scalable peer-to-peer lookup service for internet applications”, in *Proceedings of the 2001 conference on Applications, technologies, architectures, and protocols for computer communications*, ser. SIGCOMM '01., pp. 149–160, 2001.
- [3] T. Okoshi, M. Mochizuki, Y. Tobe, and H. Tokuda, “MobileSocket: toward continuous operation for Java applications”, in *Proceedings Eight International Conference on Computer Communications and Networks (Cat. No.99EX370)*. IEEE, pp. 50–57, 1999.