

Performance of mobile Web Service Access using WAP

Guido Gehlen

Communication Networks
Aachen University, Germany

guge@comnets.rwth-aachen.de

World Wireless Congress

San Francisco, May 25-28, 2004

Outline

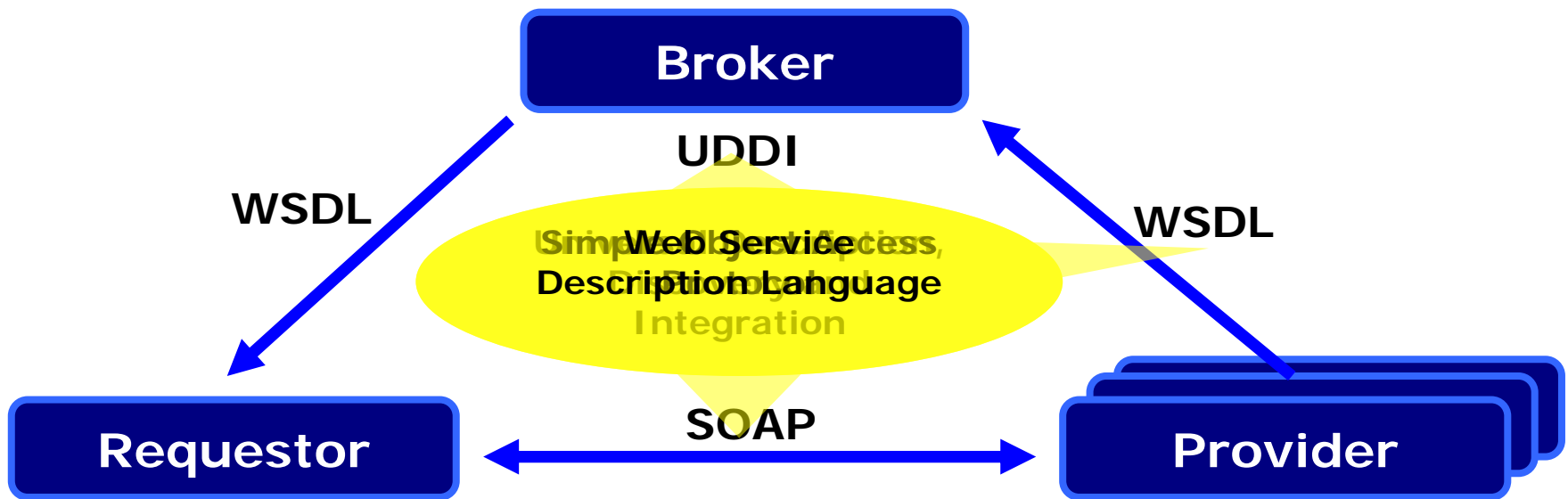
- Introduction
- Mobile Web Services
 - Technologies (SOAP, WSDL, UDDI)
 - SOAP
- Performance Analysis
 - SOAP over WAP vs. HTTP (GPRS)
 - SOAP over BEEP vs. HTTP (GPRS)
- Questions

Introduction

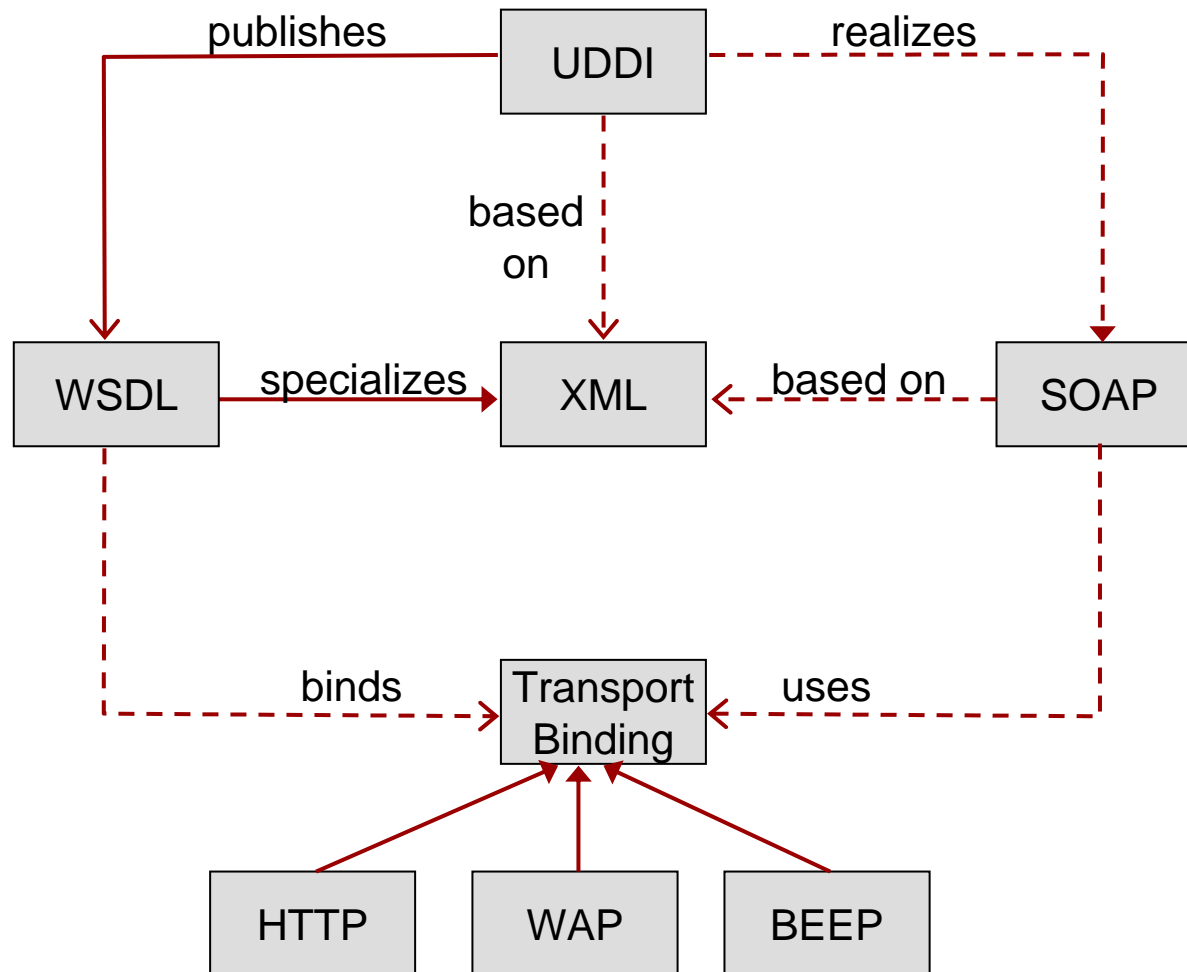
- XML Web Services spread quickly across the Internet (see www.xmethods.com)
- Mobile Access to Web Services is promising
 - Web Services are independent of their presentation
 - Leading to a common middleware, even for mobile devices
- Performance considerations for mobile Web Services are important, since ...
 - Device CPU power is low
 - Bandwidth limited in mobile communication system and reliability is bad

Web Services: Architecture

- Service oriented architecture (SOA)
 - WS-Requestor
 - WS-Provider
 - WS-Broker
- **Communication:** SOAP
- **Description:** WSDL
- **Service-Management:** UDDI



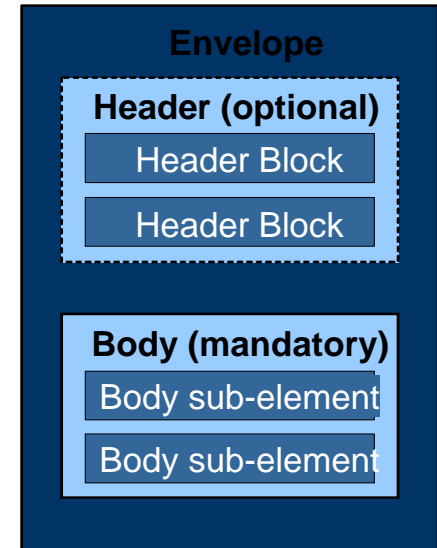
Web Services: Technologies



Web Services: SOAP

■ Enclosing envelope with two sub-elements

- Header (optional)
 - “out-of-band” control information (≠ application payload)
- Body (mandatory)
 - carries end-to-end information
 - RPC method names + parameters



■ Bindings

- How messages are passed between SOAP nodes
 - ⇒ define transmission protocol and serialization of objects
- Standard binding: HTTP
 - Server and SOAP node on server identified by URI
 - Using HTTP POST methods
- Provide flexibility (using alternative transport protocols, e.g. WAP, BEEP)

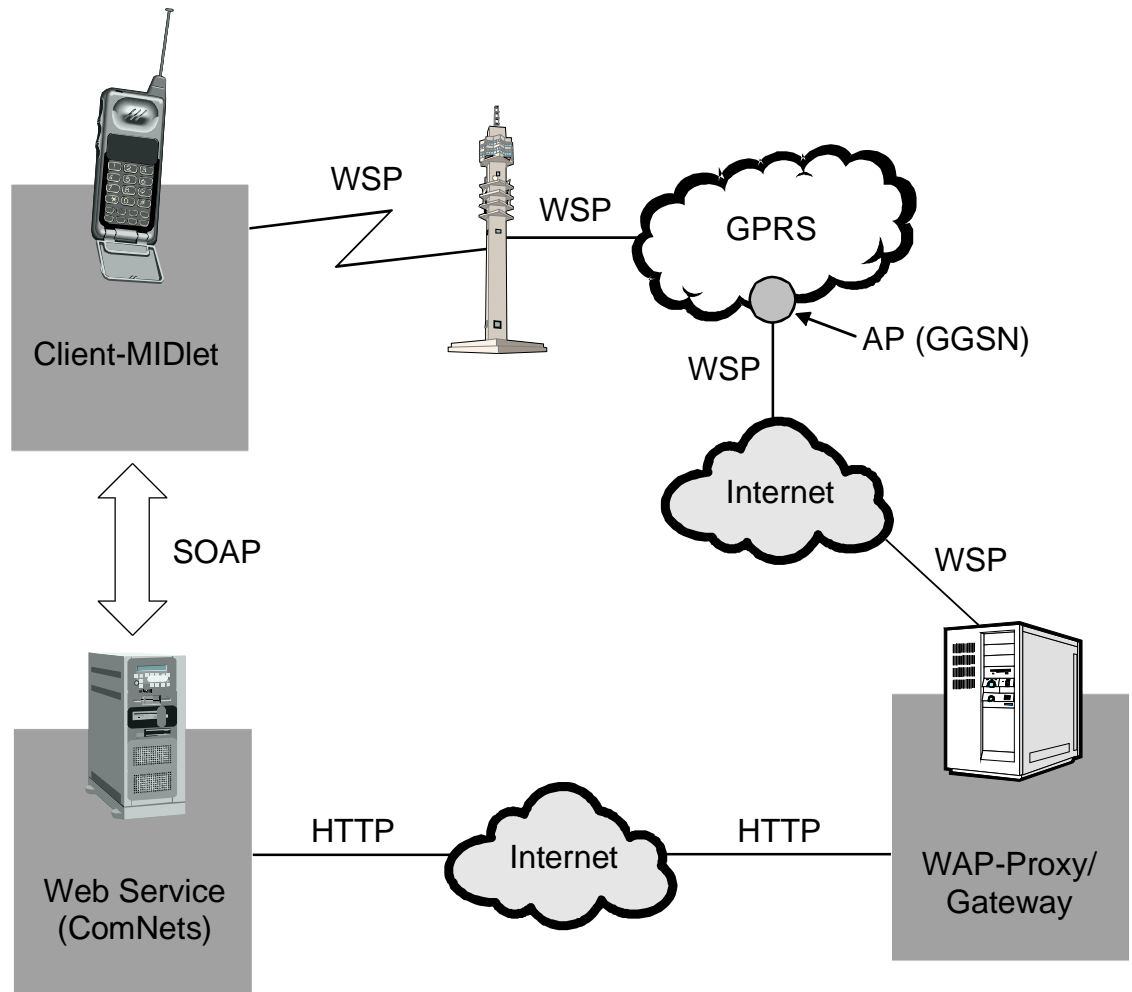
HTTP vs. WAP

- Traditional reliable protocols designed with wired networks in mind (e.g. TCP):
 - packet delay interpreted as network congestion
 - congestion window closed (TCP slow-start)
 - ⇒ performance degradation
 - three-way handshake: causes large latency values
 - ⇒ not well-suited for wireless communications
 - ⇒ design of Wireless Application Protocol family
- ⇒ Choice of WAP/connectionless WSP because of:
 - “light-weight” protocol (well-suited for thin clients)
 - encoded headers reduce protocol overhead
 - three-way handshake eliminated (**but** unreliable connection!)
 - reduction of payload volume by using WBXML

SOAP/WAP/GPRS

Internet-capable mobile phones use two-part solution:

- WAP Wireless Session Protocol on wireless link
- HTTP on wired link (connection to Internet)
- WAP Proxy as a gateway between these two systems



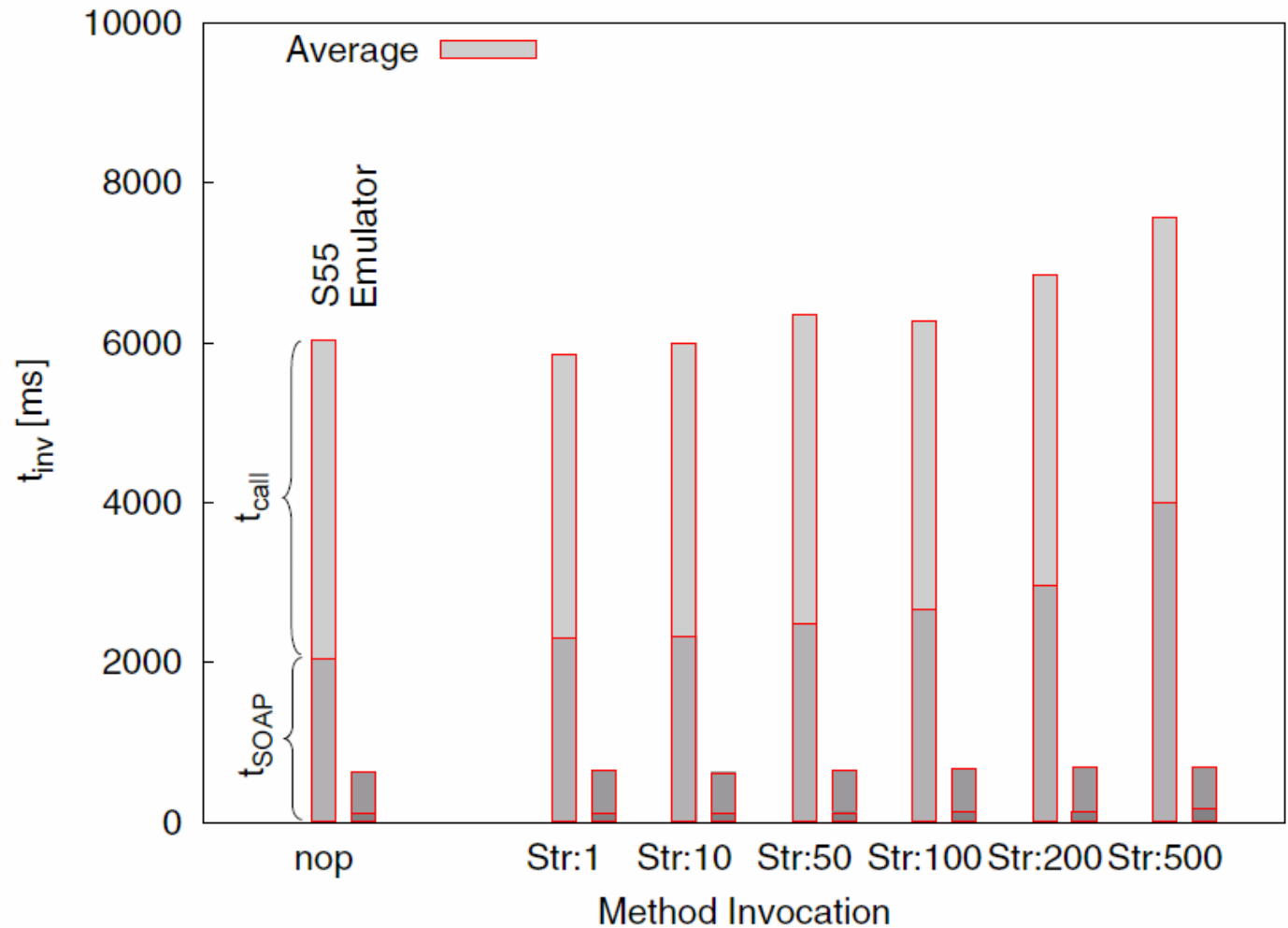
Performance: Criteria & Measurements

- Criteria:
 - Latency
 - Data Transfer Volume (com. costs)
 - Memory footprint
 - CPU use
- Measurements:
 - Reference Web Service
(nop(), getString(), getIntArray())
 - Client MIDlet
 - Logging Web Service
 - Create different test scenarios with only one varying parameter

Performance: SOAP/HTTP/[GPRS, Eth.]

S55:
HTTP over
GPRS

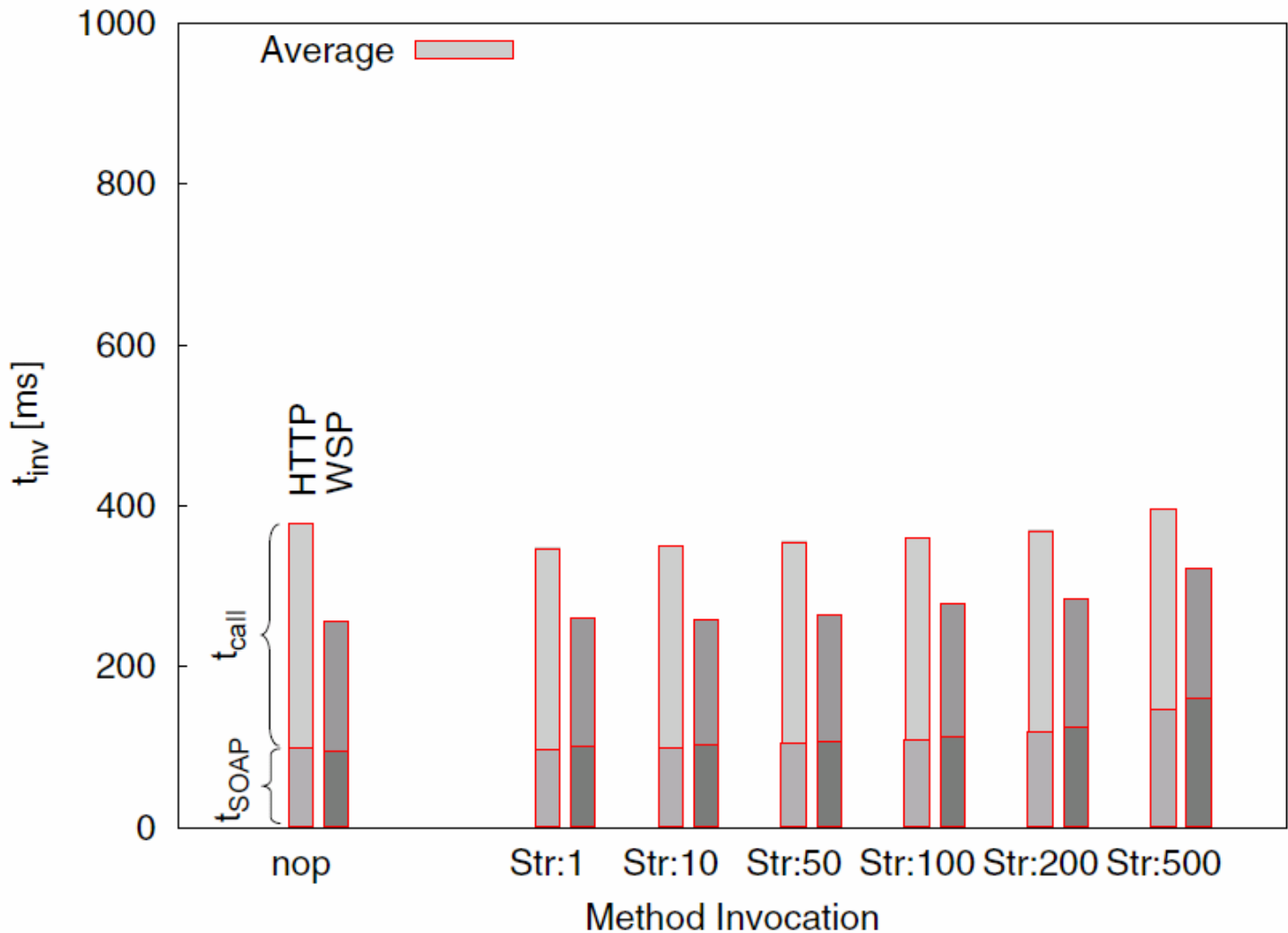
Emulator:
HTTP over
Ethernet



Performance: SOAP/[HTTP,WSP]/Eth.

left:
Emulator
using HTTP
over Ethernet

right:
Emulator using
WSP over
Ethernet



Evaluation: SOAP/WAP/GPRS

- Sizes of SOAP messages reducible by a third using WBXML (Binary XML encoding)
- Connectionless WSP as transport protocol avoids TCP's 3-way-handshake
 - reduces latency by about 1/3
 - **but:** one loses reliability of connection-mode TCP
- Reduction of protocol data overhead by more than 1/3 by selecting WSP instead of HTTP
- Adding WSP support to MIDlets increases size by about 35 K
 - sandbox → native WAP implementation inaccessible
- Lack of performance on mobile phones
 - ⇒ *complex* SOAP-based applications not yet possible
 - ⇒ simple applications possible and useful

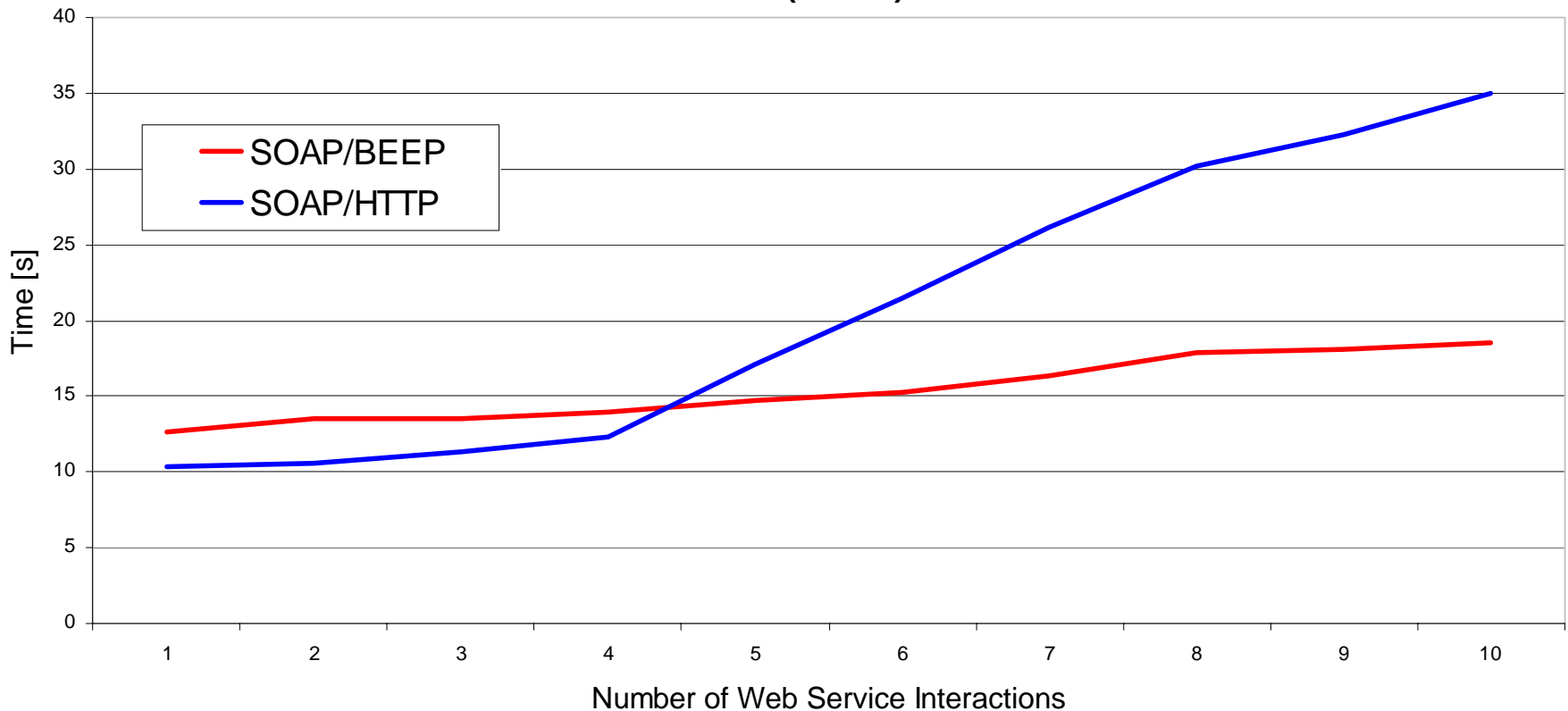
SOAP/BEEP/GPRS

- **BEEP** (Blocks Extensible Exchange Protocol)
 - Framework for Simplifying and Optimizing Application-Protocols
 - BEEP is asynchronous and message-oriented
 - BEEP supports multiplexing
 - Concept of Channels
 - Concept of Profiles
 - Advantages:
 - Only one TCP-Connection, but many channels
 - ⇒ Only one three way hand shake
 - ⇒ Performance enhancement in terms of latency
 - Less overhead

Performance: SOAP/BEEP/GPRS

- BEEP Protocol Binding for Java running on a PC

Latency (GPRS)



Evaluation: SOAP/BEEP/GPRS

- By using BEEP unnecessary three-way handshakes can be saved
- Compared to HTTP SOAP Binding a gain resp. latency and data volume it can be achieved exceeding 4 interactions
- Reduction of protocol data overhead by more than 1/3 by selecting WSP instead of HTTP
- P2P feature of BEEP will ease the development of Web Service Provision and Access from one device

Conclusion

- Bad Performance of mobile access to Web Services using SOAP over HTTP
- SOAP over WAP
 - Data volume and latency can be reduced
 - Using existing Web Services bound to HTTP possible (through WAP gateway)
 - WAP support on mobile device by default
- SOAP over BEEP
 - Performance improvement by multiple accesses (saving TCP 3-way-handshakes)
 - P2P capability is a interesting additional feature

Related Work/Outlook

- Mobile Web Service Provision
 - SOAP HTTP Server on mobile devices
 - SOAP WAP/BEEP Server
- Performance of Web Service Security on mobile devices
- P2P Web Service application for mobile systems
- Alignment of UPnP to Web Services
 - WS-Eventing
 - WS Dynamic Discovery

Questions?



Thank you for your attention!