



**The Underlay Abstraction in the
Spontaneous Virtual Networks
(SpovNet) Architecture**

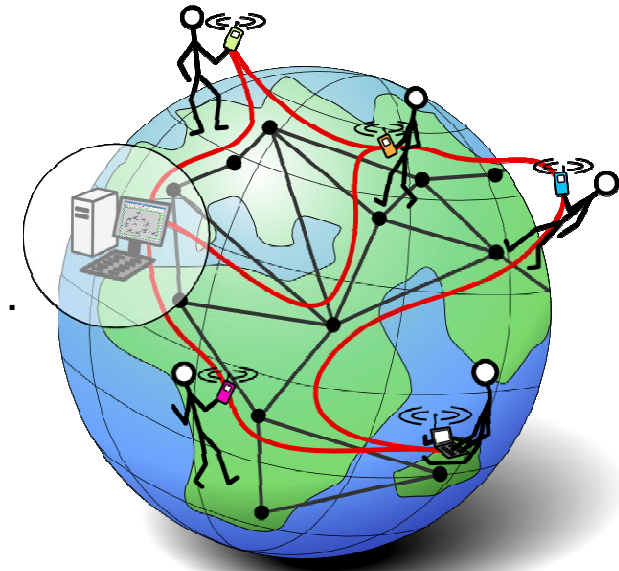
www.spovnet.de

R. Bless, C. Hübsch, S. Mies, and O. Waldhorst

Institute of Telematics, Universität Karlsruhe (TH), Germany

NGI 2008, 28. Apr. 2008

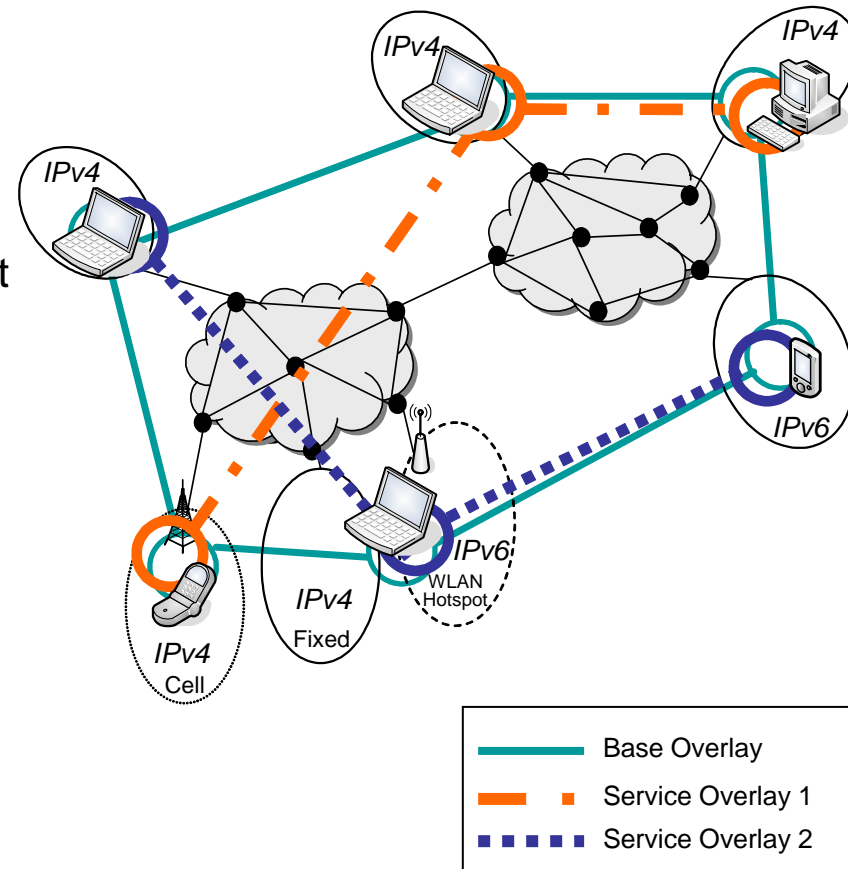
- Internet has evolved from 4-node network to ubiquitous, global communication network
 - But ... is it flexible enough for the future?
- Patchwork design and deployment problems
 - “Half” layers: IPsec, MPLS (2.5); TLS (3.5) ...
 - TCP adaptations to wireless, mobile etc.
 - Multicast, MobileIP → Deployment?
- How to improve flexibility?
 - Calls for new architectures !?
 - Clean slate ... time horizon of 10 years and more?
 - Overlay-based architecture ... the way SpovNet goes!





- 1) Provide communication services **flexibly, adaptively** and **spontaneously** on top of **heterogeneous** networks
- 2) Enable **seamless transition** from current to future networks

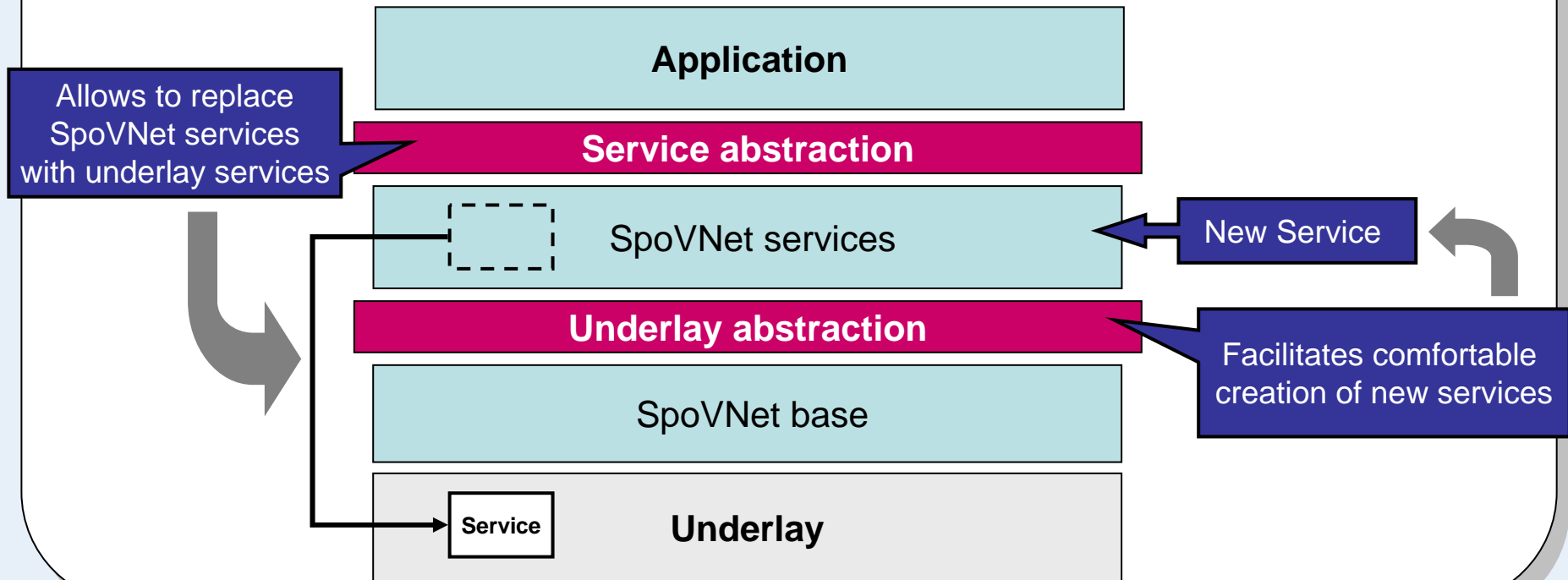
- Extensible set of services implemented by **overlays**
 - Spontaneous and flexible per application
 - No infrastructure support required
 - Self-organizing, scalable and robust
- What is different to other Overlay-based approaches?
 - Framework provides **generic (transport-)mechanisms** for convenient realization of overlay-based services and innovative applications
 - Optimization and adaptivity using **Cross-Layer Information**



SpoVNet Architecture



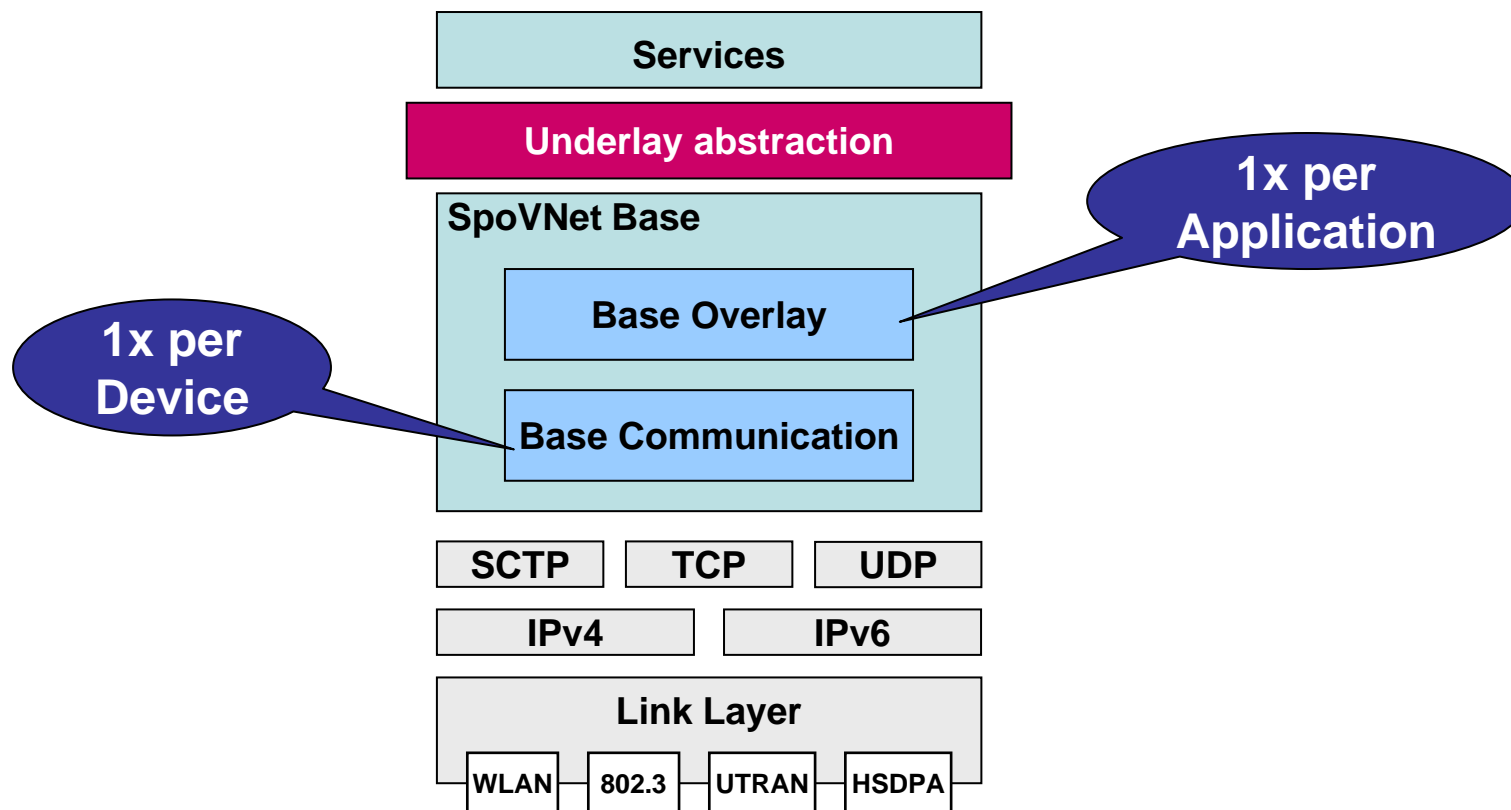
- Goals: Provide a **framework** that
 - allows comfortable creation of application supporting (overlay-)services in **heterogeneous** networks
 - assures that these services can be **incrementally replaced** by **evolving underlay** services



SpoVNet Underlay Abstraction



- Provides generic transport mechanisms hiding mobility, multi-homing and heterogeneity



- Implementation of a P2P VoIP service
 - May include **mobile** devices and **heterogeneous** networks (WLAN, ad-hoc, ...)
 - Underlays may (partially) provide **quality-of-service** support
- Requirements
 - **Identification** and localization of VoIP host
 - **Establish** connection in heterogeneous environments
 - **Keep up connectivity** at all time with suitable QoS

→ Can be implemented on top of the SpovNet underlay abstraction

VoIP Connection Establishment



○ Base Overlay

- Provides identifier-based addressing



Maps (SIP-)address to a set of locators

○ Base Communication

- Connections driven by requirements
- Selects a appropriate L4 protocol
- Provides persistent end-to-end connectivity, even when
 - nodes are **multi-homed**
 - **relaying** is necessary between different L3 protocols
 - mobility causes **locator changes**
- Maps quality-of-service requirements

QoS: max-delay=80ms;
Security: confidential;
Reliability: none;

Use DTLS/UDP
(Unreliable/Secure)

Use underlay mechanism (e.g., NSIS) to signal QoS or use cross-layer information for “best-effort” QoS



Further Features of the Underlay Abstraction



- Base Communication
 - Sending one-shot (datagram) messages
- Base Overlay
 - Creating a SpoVNet instance
 - Initiator fixes properties (ID, cryptographic functions, authentication policy)
 - Joining a SpoVNet instance
 - Requires **authorization** and **integration** into Base Overlay
 - Authentication Mechanisms
 - Cryptographic node identifiers
 - No spoofing of identifiers possible
 - Authorization Mechanisms
 - Centralized: Initiator decides on authorization
 - Decentralized: Initiator may authorize other nodes to perform authorization

- SpovNet ...
 - ... enables **flexible**, **adaptive**, and **spontaneous** provisioning of application-oriented and network-oriented **services**
- SpovNet Underlay Abstraction
 - Supports easy creation of **overlay-based services**
 - Generic transport mechanisms to deal with **underlay characteristics** (mobility, multihoming, heterogeneity, ...)
 - Identifier-based addressing scheme
 - **Transparent selection** of transport- and network-layer protocols to provide **persistent transport links**
- Current work
 - Evaluation by simulation models
 - Prototype implementation