MPP2006

Pisa, Italy - March, 17 - 2006

ERICSSON

On the Use of Peer-to-Peer Architectures for the Management of Highly Dynamic Environments

Carlos Kamienski (cak@gprt.ufpe.br)

Djamel Sadok (jamel@cin.ufpe.br)

Joseane Fidalgo (joseane@cin.ufpe.br)

Jennifer Lima (jennifer@cin.ufpe.br)

Börje Ohlman (Borje.Ohlman@ericsson.com)

Introduction

- Traditional management approaches not designed to deal with Internet wide services for mobile users
- Policy-based Management (PBM) may be an option
  - Policies are the aggregation of rules
    - Each one consists of one or more conditions and actions
    - Describe how to allocate resources to meet business goals
- IETF PBM framework
  - Policy Decision Points (PDP)
  - Policy Enforcement Points (PEP)
  - Policy Repository

Motivation for a new framework

- The IETF framework is focused on simpler problems from typical corporate networks
  - We consider service usage in the global Internet
- 3G/4G: huge number of **mobile users** with highly dynamic mobility and service usage patterns
- New requirements associated with those environments, provided by the **p2p technology**
  - Scalability, fault tolerance, self-configuration, load balancing, …

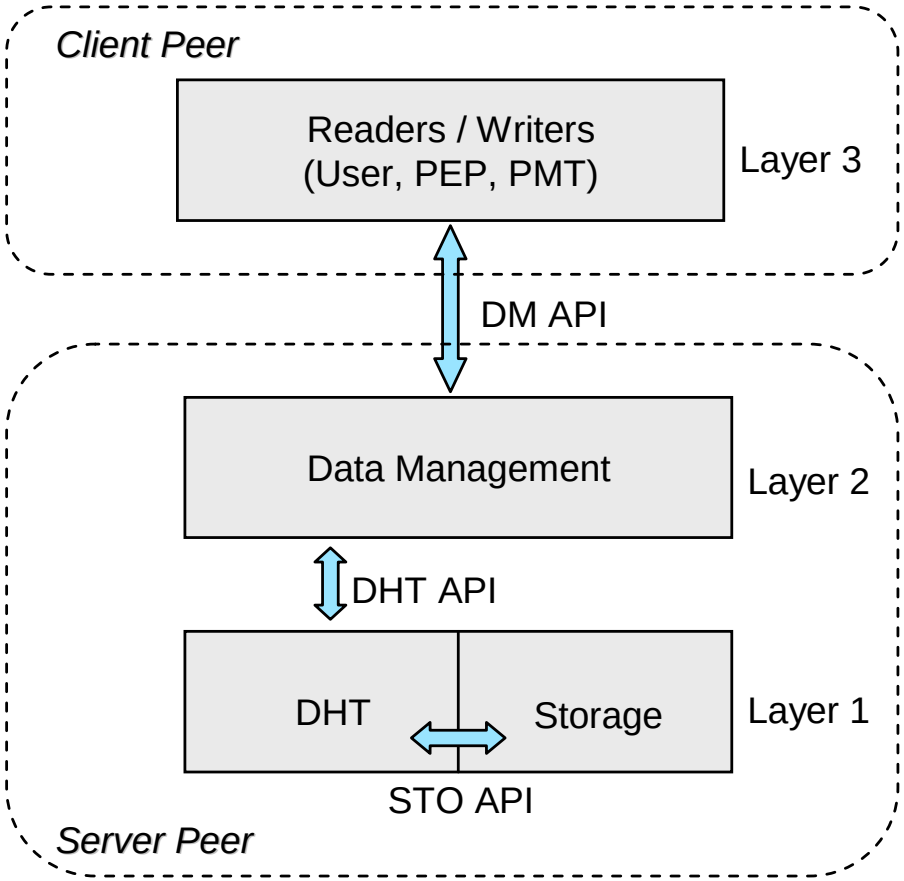
Paper Proposal

- P2P Policy Management Infrastructure ⇒ **P4MI**
  - PBM framework based on the p2p technology as the main enabling mechanism
  - Based on a hierarchical p2p architecture, using Distributed Hash Tables (DHT)
  - Abstract framework
- PBM for **Ambient Networks** ⇒ **PBMAN**
  - An application (instantiation) of P4MI
  - Solutions for mobile and wireless systems beyond 3G

P4MI Architecture

- Policy Decision Network (PDN): core of P4MI
  - Responsible for policy management and P2P issues
  - Take decisions upon receiving service requests
    - Policies are retrieved and processed
  - Comprised of:
    - Decision Points: P-Nodes (abstraction of policy servers)
    - Repositories
  - Based on **DHT (PDN-Ring)**
- Policy Agents (composed of)
  - User and Policy Enforcement Point (PEP)

P4MI Data Model



- **Layer 1:** DHT network + a simple storage system for dealing with records
- **Layer 2:** higher-level view of the data management system, allowing more complex data structures
- **Layer 3:** includes all readers and writers of policies, such as Users and PEPs

Ambient Networks

- Ambient Networks (AN):
  - New concept aimed at creating network solutions for mobile and wireless systems beyond 3G
- Enables cooperation of **heterogeneous** networks
  - belonging to different **operators**
  - or **technology** domains
- The key enabling concept is **network composition**
  - For allowing instant and dynamic user access to services and resources

Network Composition

- Network Composition enables control-plane interworking and sharing of control functions among networks
- Composition can be thought of as a mechanism for automatic negotiation of roaming and/or service level agreements (SLAs)
  - Today negotiations are done manually

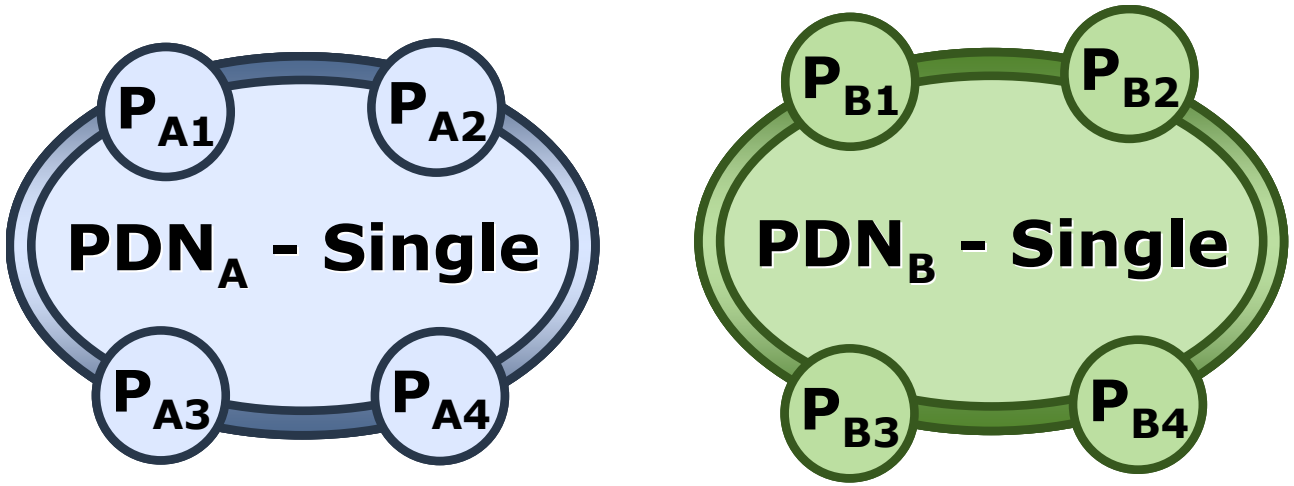
PBMAN

- PBMAN is concerned with the design and implementation of a management infrastructure for Ambient Networks
- Enabling technologies: PBM and P2P
- PBMAN goals
  - Framework specification: **abstract**
  - Prototype implementation: **real**
    - X-PBMAN: based on the X-Peer middleware (DHT)
  - Spiral methodology
    - Concepts ⇒ technology ⇒ implementation ⇒ validation ⇒ concepts ⇒ …

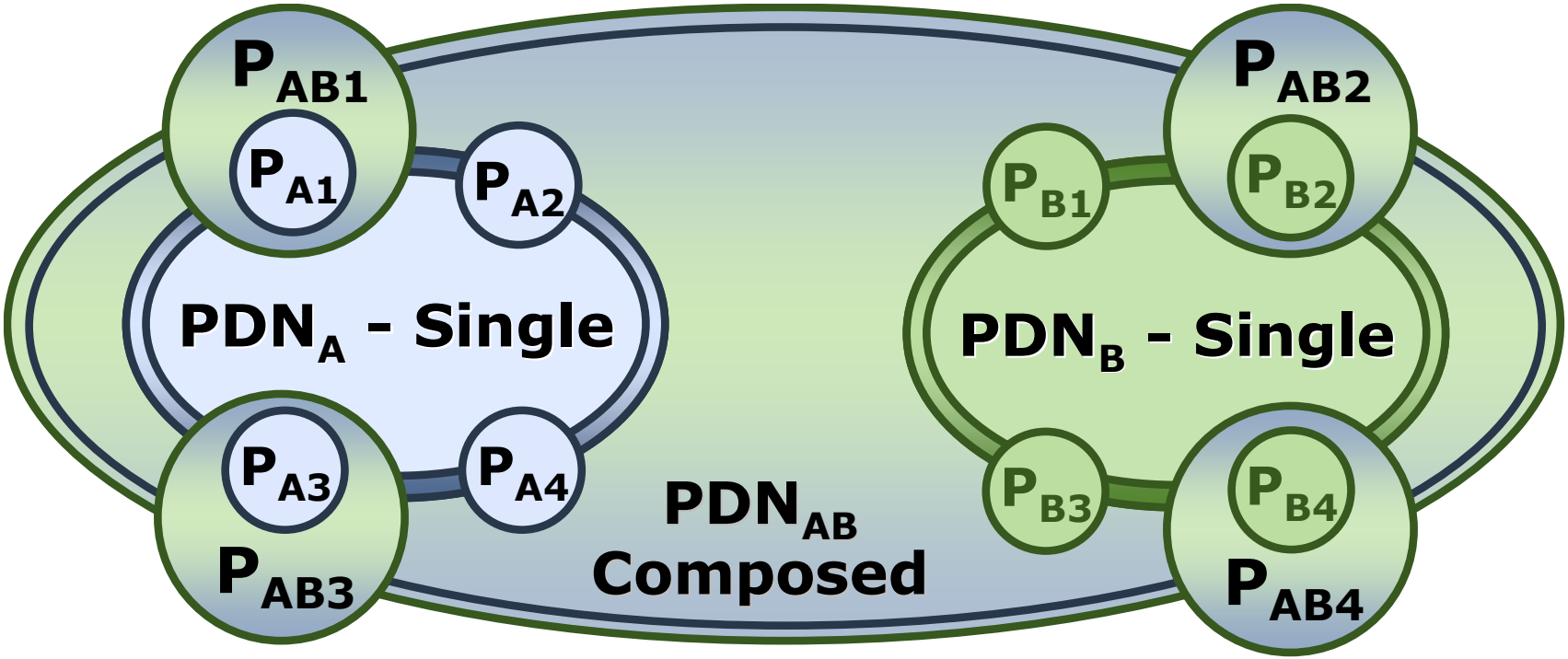
Network Composition in PBMAN

- Composition types
  - PDN/PDN
    - P2P composition: complex
    - Creation of a new composed DHT out of the single ones
  - Agent/PDN
    - Typically a login process
    - Client/Server composition
  - Agent/Agent
    - P2P composition managed by the PDN
    - Allows virtual compositions

PDN/PDN Composition - Before



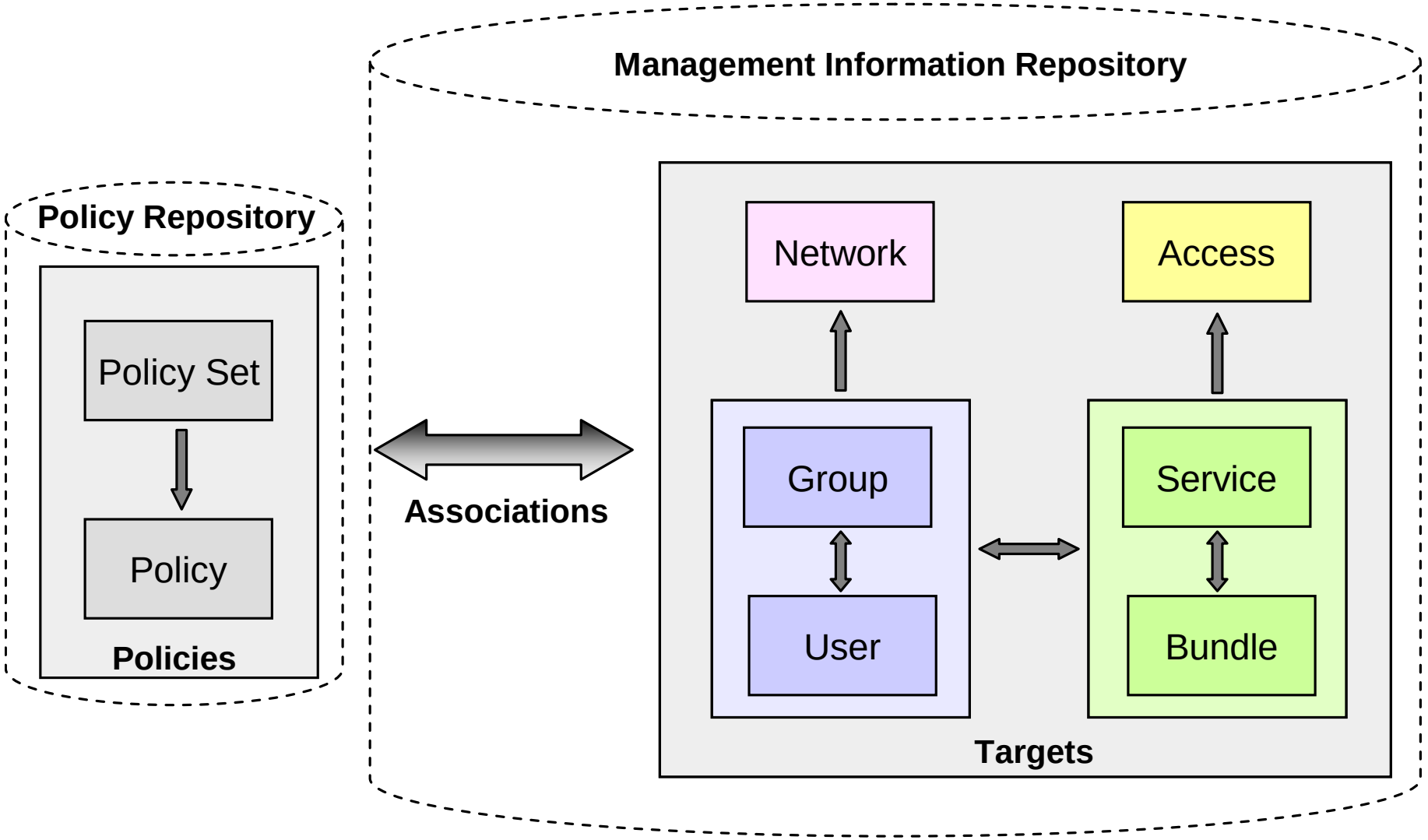
PDN/PDN Composition - After



PDN/PDN Composition: Algorithm

- Step 1: Composition request from PDN_A to PDN_B
  - Two negotiators: $P\text{-}Node_A$ and $P\text{-}Node_B$
- Step 2: Composition accepted by PDN_B
  - $P\text{-}Node_A$ (source) creates new DHT
- Step 3: Join requests to the new PDN
  - $P\text{-}Node_A$ and $P\text{-}Node_B$ send "join" messages to other P-Nodes
- Step 4: Joining and republishing information
  - All important information (policies) is republished
- Step 5: Composition finished
  - All P-Nodes joined the PDN or timeout exceeded

PBMAN Information Model



Mapping: Information ⇒ Data Model

- PBMAN information model is mapped to the P4MI data model
- Basic DHT layer stores Key ⇒ content
  - DHT extension to allow complex data structures
  - Idea: build a distributed database atop the DHT
  - Key is divided into 3 parts: Key1, Key2 and Key3
  - Content is used to create lists, etc.

Data Management Layer: Examples

- Examples: Key ⇒ content
- {AN Id + Policy Id + info}
  - AN1 + PVoD1 + info ⇒ if service-req == vod then …
- {AN Id + Target Id + policy}
  - AN3 + VoD + policy ⇒ PVoD1, PVoD2, PVoD3, …
- {AN Id + Policy Id + target}
  - ISP_XYZ + PVoD1 + target ⇒ VoD, Jose, PremiumGrp

Future work

- Improvement of composition concepts
- Continuation of the prototype implementation
- Performance evaluation
  - Different types of compositions
  - Data Management Layer

Conclusions

- P4MI
  - Abstract PBM framework: hierarchical DHT network
- PBMAN
  - Instantiation of P4MI for Ambient Networks
  - P2P network for managing highly dynamic environments
  - Enables network composition
- X-PBMAN: Proof-of-concept prototype built
- Framework and prototype are work in progress
- Composition implemented: How to merge DHTs efficiently?

MPP2006

Pisa, Italy - March, 17 - 2006

ERICSSON

On the Use of Peer-to-Peer Architectures for the Management of Highly Dynamic Environments

Thank You!

This work was supported by the Research and Development Centre, Ericsson Telecomunicações S.A., Brazil

Composition: Motivation

- Networks become dynamic regarding structure and constituents
  - Network configuration must become flexible and autonomous
- Composition: generic Plug & Play method for creating, extending and merging networks
  - Automatic configuration of joint control plane
- Difference between end-system and network blurs
  - Users own networks that they attach to other networks