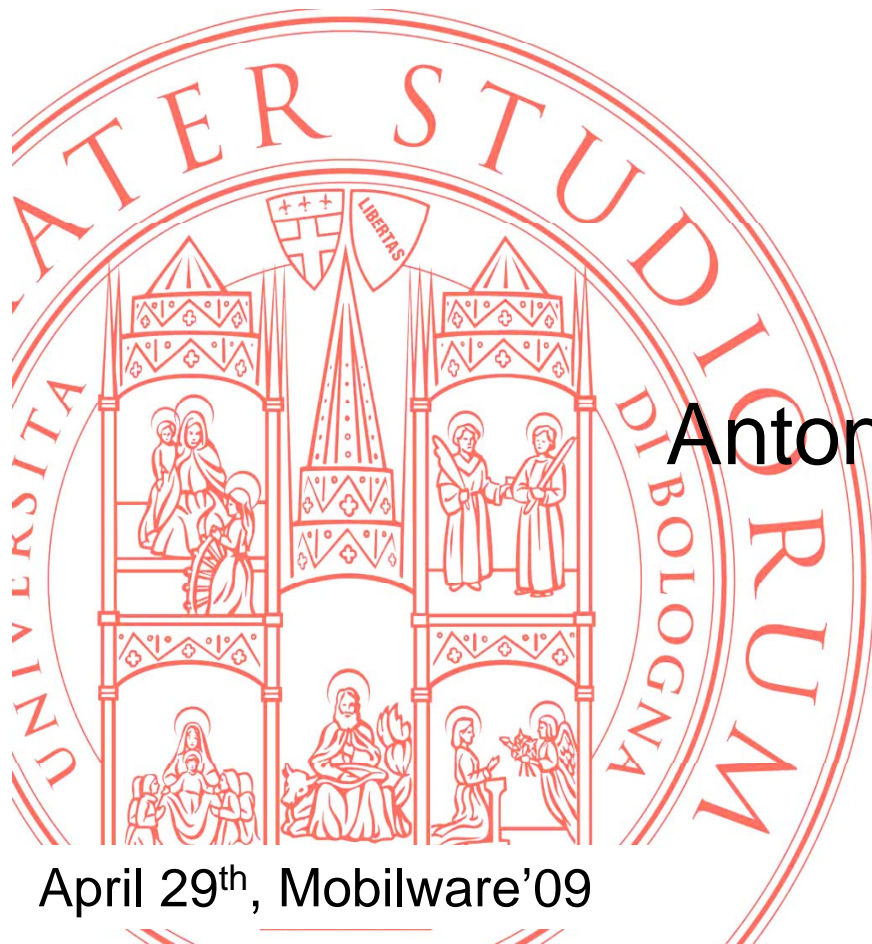


# An IMS-based Middleware Solution for Energy-Efficient and Cost-Effective Mobile Multimedia Services



Paolo Bellavista,  
Antonio Corradi, Luca Foschini

DEIS, University of Bologna, ITALY  
{[paolo.bellavista](mailto:paolo.bellavista@unibo.it), [antonio.corradi](mailto:antonio.corradi@unibo.it), [luca.foschini](mailto:luca.foschini@unibo.it)}  
@unibo.it

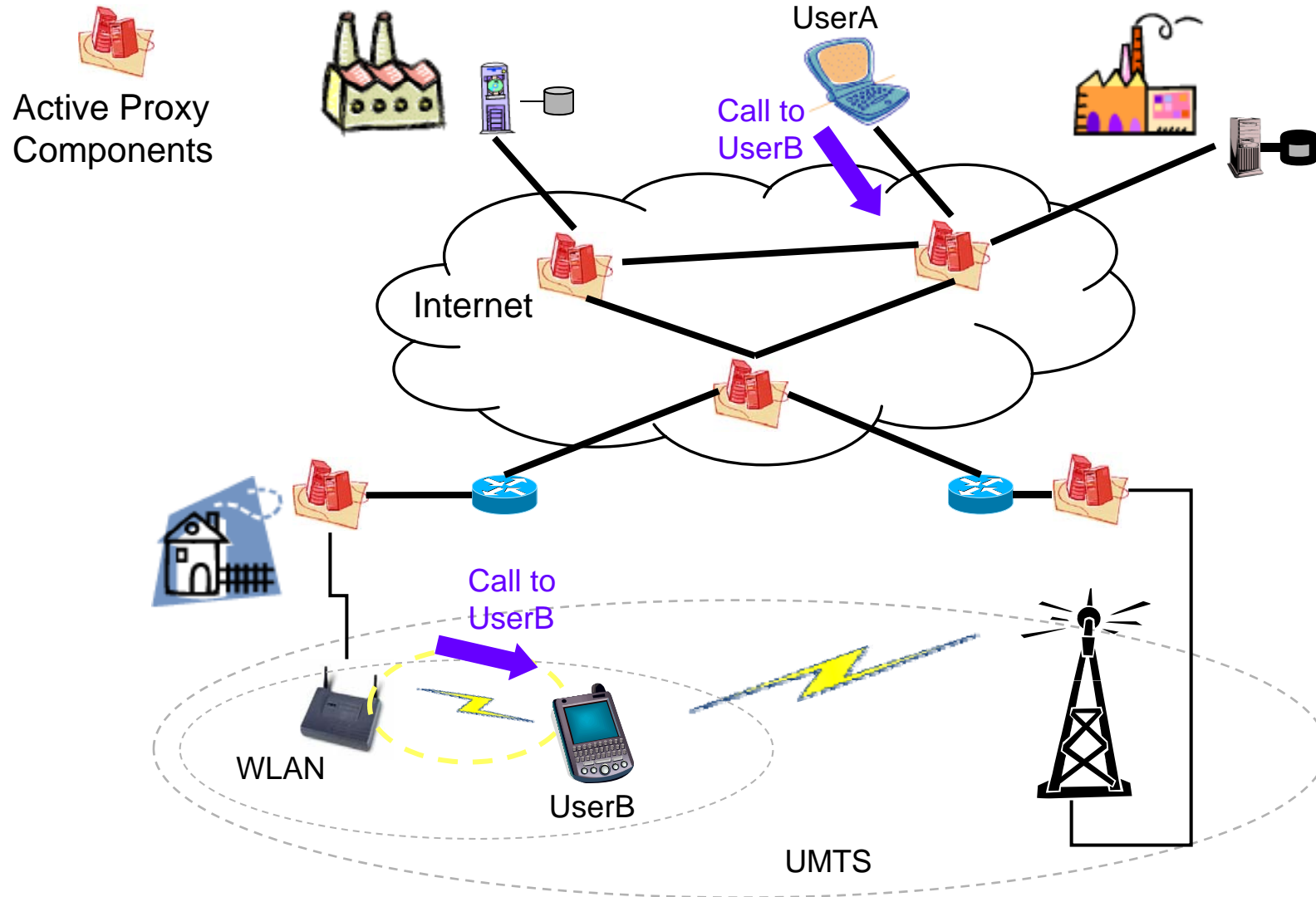
April 29<sup>th</sup>, Mobilware'09



- **Multimedia service delivery** for the *wireless Internet*
- **Middleware** for *IMS-based energy-efficient mobile multimedia services*
  - The IMS-compliant Handoff Management Application Server (IHMAS) multimedia middleware solution
  - **Context-aware, IMS-standard-based, and dynamic switching-on of wireless interfaces only for the duration of multimedia service sessions**
- Implementation insights and experimental evaluation
  - Switch on and session re-direction time evaluation
  - Energy-saving evaluation

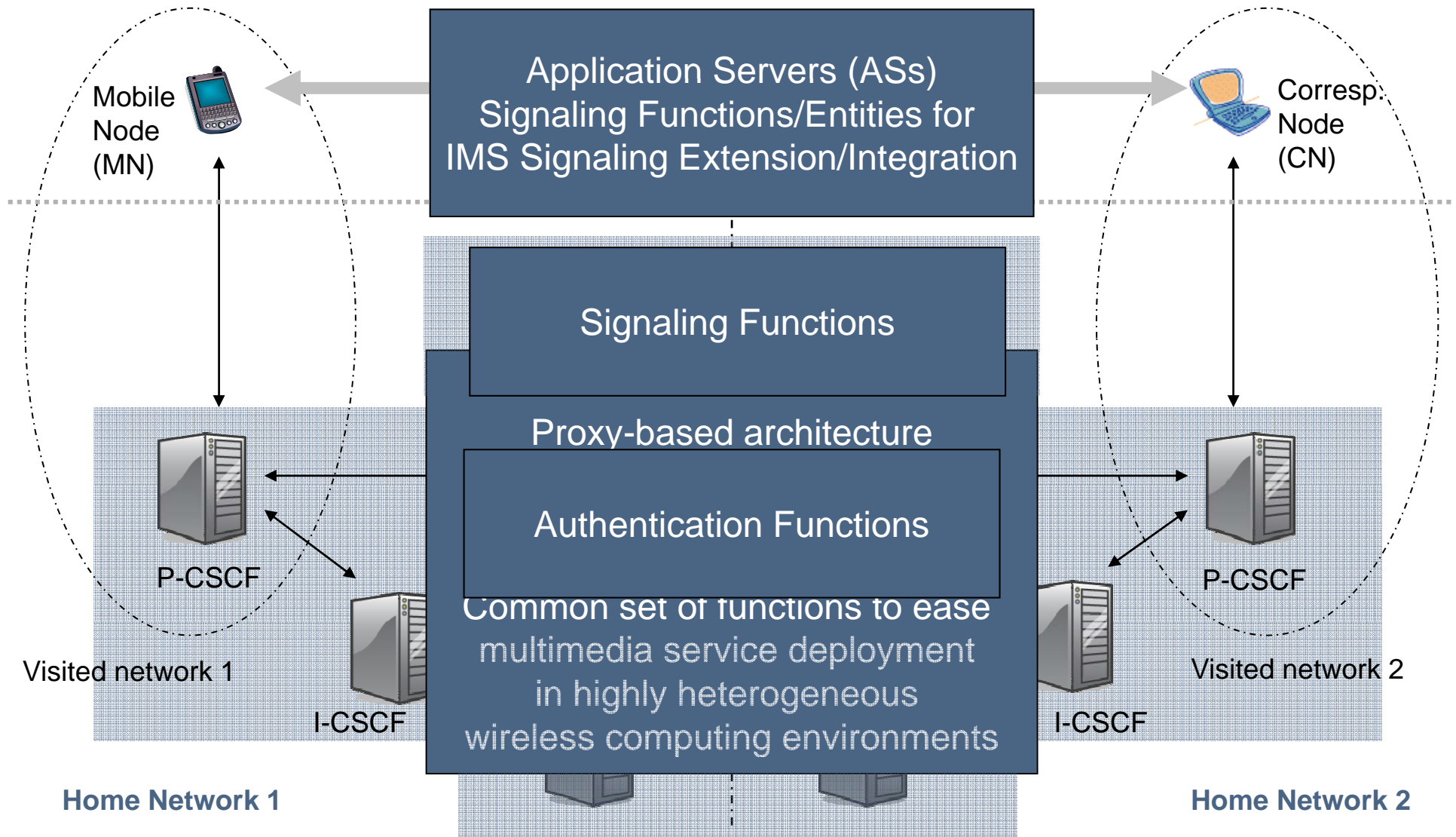


# Application scenario: multimedia services in the Wireless Internet





# IMS – IP Multimedia Subsystem

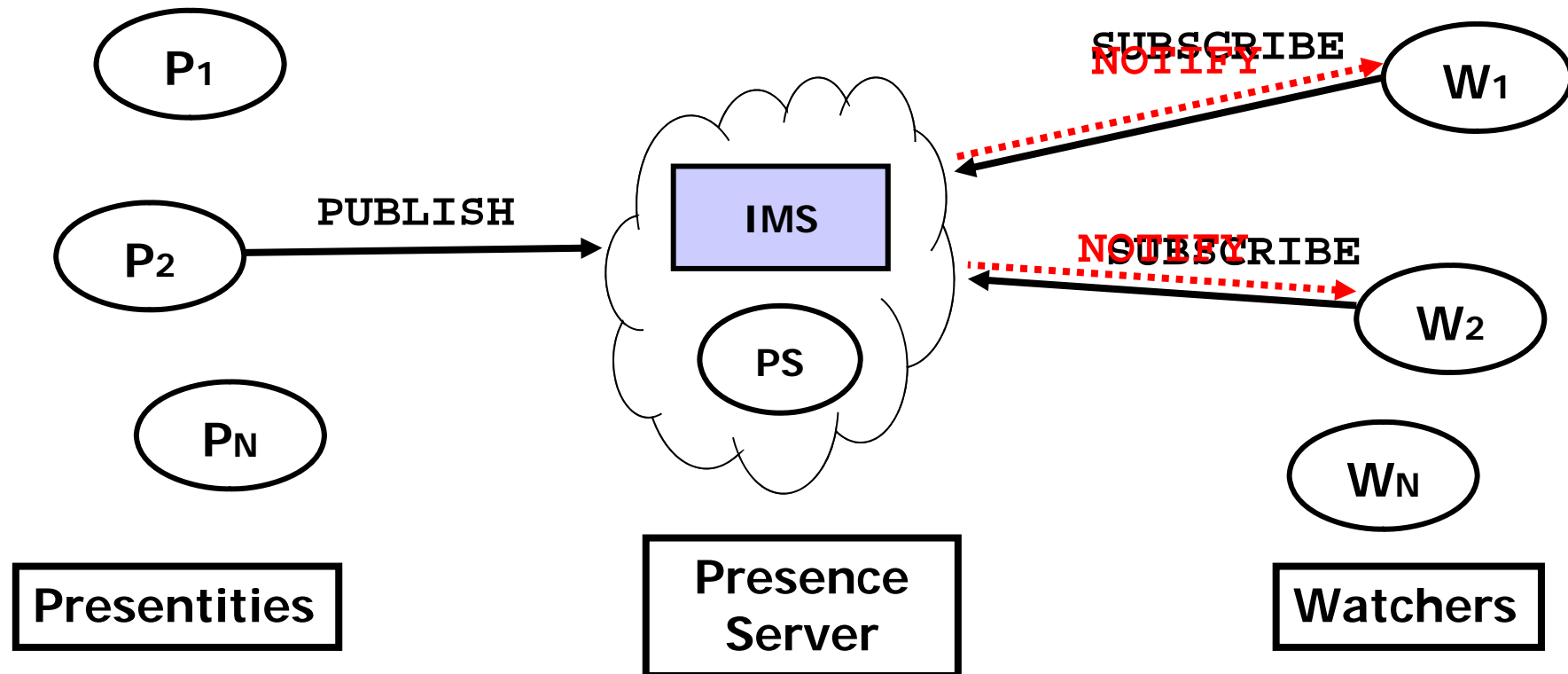


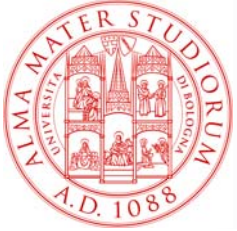


# IMS Presence Service

**Presence Service (PS)** permits users and hw/sw components, called **presentities** ( $P_i$ ), to convey their ability and willingness to communicate with subscribed **watchers** ( $W_j$ )

IMS standardizes PS as a specific AS → **IMS PS**





# Application-level middleware for multimedia services

- ***Session management and continuity***
- ***Context-aware*** power management ***middleware***
  - **updates** low level parameters (wireless communications availability, battery level, ...)
    - ***wireless access prediction and mobile node energy monitoring***
  - **executes** application-level specific **energy-saving decisions** and **session signaling actions**
    - ***dynamic wireless interface switch on and automatic session re-direction/handoff***
  - **integrates** seamlessly with existing infrastructures
    - ***full compliancy with IMS standard***



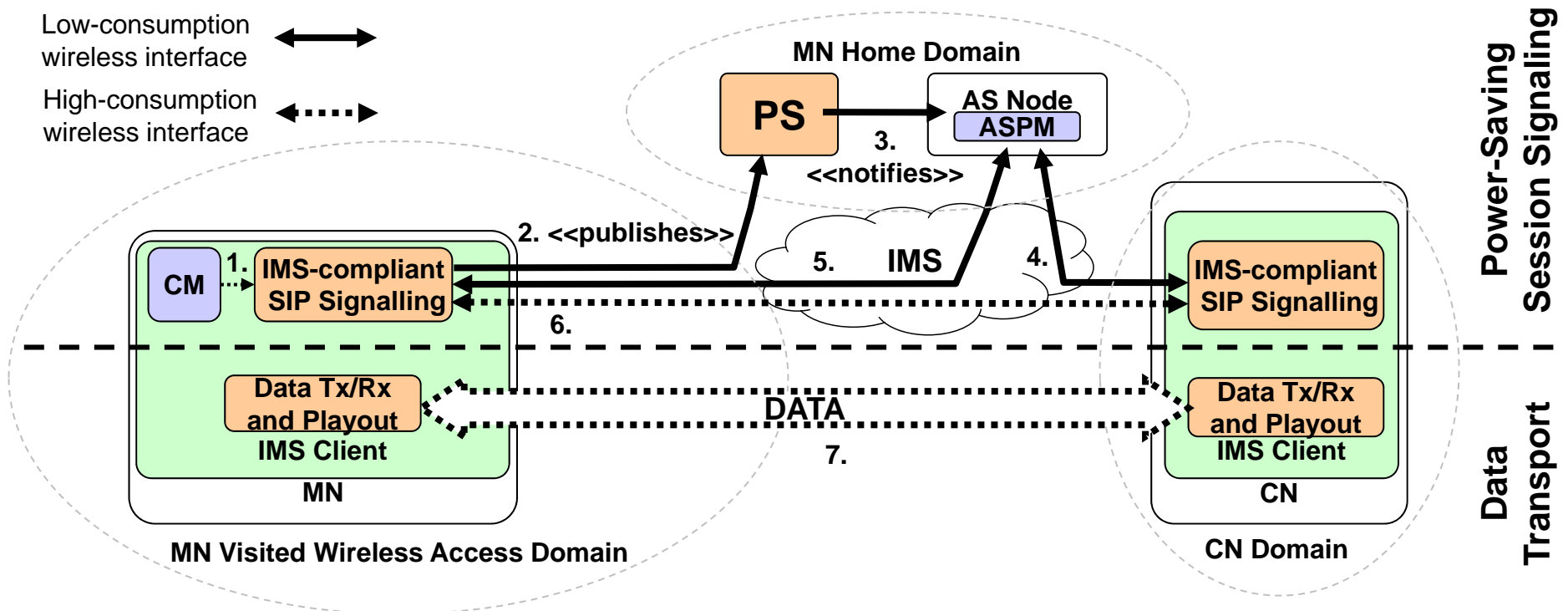
# IHMAS active middleware

## IMS-compliant Handoff Management Application Server

- Multimedia session continuity
  - **Session signaling enhancements** for power management
- Active session signaling and media data paths
  - Dynamic session signaling and re-direction **to exploit high energy-consumption and low transmission-cost wireless interfaces**
- IMS-compliant solution
  - Session management entity realized as a **novel IMS AS**
  - AS performs management operations
    - **Context-aware energy-saving decision making and orchestration**
- Application-level approach
  - **Seamless integration** with existing services
    - **IMS PS** to deliver **context updates** about mobile node



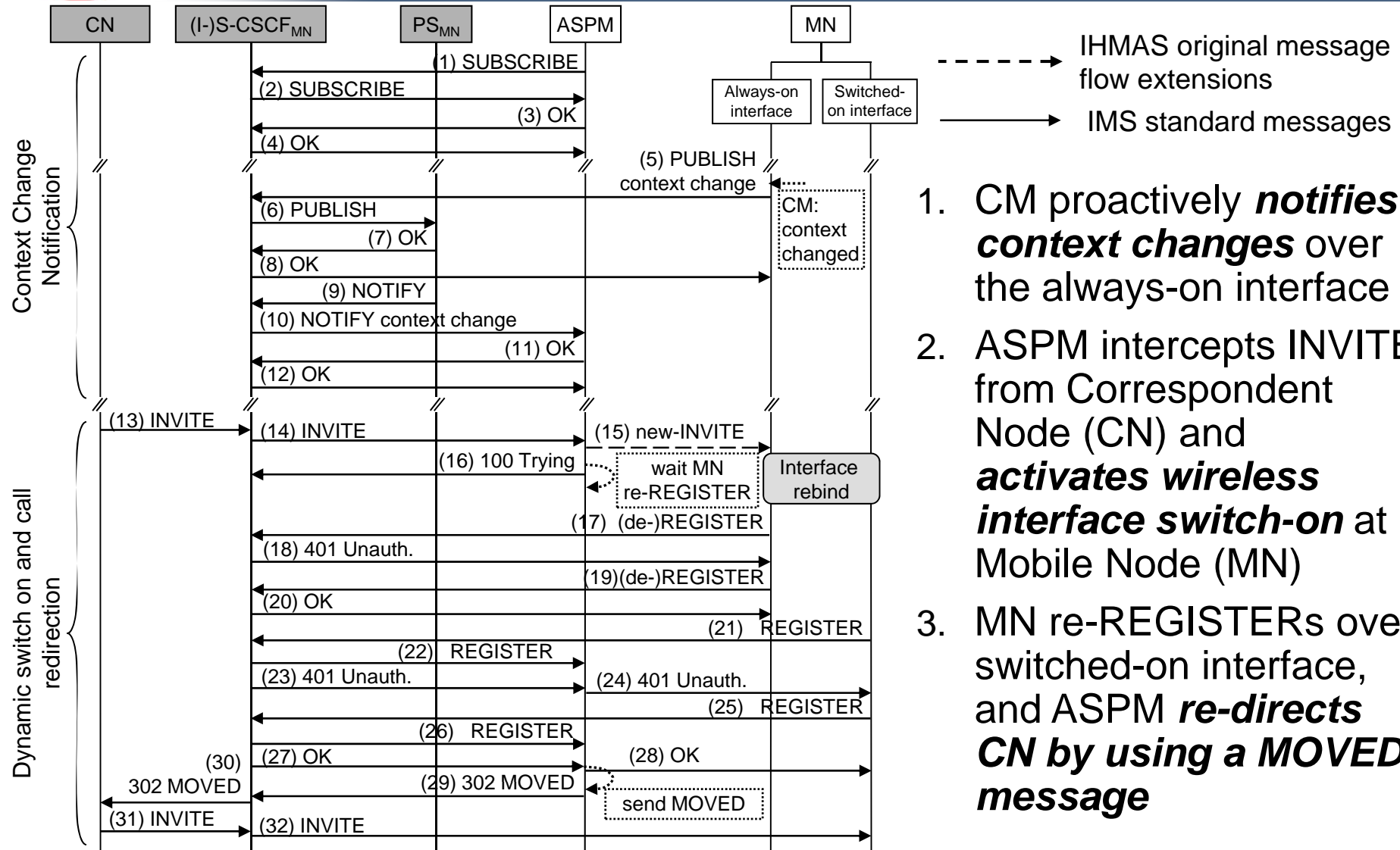
# IHMAS Power Management Facility: Distributed Architecture



- Context Monitor – **CM** (one for client): implements lightweight and completely decentralized context monitoring via local access to client wireless devices
- AS for Power Management – **ASPM** (one for IMS domain): realizes our IMS energy-saving optimization
- IMS PS – **PS** (one for access locality): facilitates CM-ASPM interactions



# IHMAS Power Management Facility: Modified Invitation Protocol



1. CM proactively **notifies context changes** over the always-on interface
2. ASPM intercepts INVITE from Correspondent Node (CN) and **activates wireless interface switch-on** at Mobile Node (MN)
3. MN re-REGISTERS over switched-on interface, and ASPM **re-directs CN by using a MOVED message**



# Implementation Insights: AS for Power Management

```
private void processInvite(Request request) {  
1. SessionDescription sdp =  
   sipUtils.getSessionDescription(request);  
2. sd=addPowerManagementAttribute(request, sd);  
   request.removeContent();  
   try {  
     request.setContent(sd,  
       headerFactory.  
         createContentTypeHeader("application", "sdp"));  
   } catch (ParseException e) { e.printStackTrace(); }  
   try {  
3. sipProvider.sendRequest(request);  
   } catch (SipException e) { e.printStackTrace(); }  
}
```

- ASPM implementation is based on **JAIN SIP stack**
- On INVITE message, ASPM:
  1. Extracts Session Description Protocol (SDP) part of the message
  2. Adds in the optional field (“a:” field) the MAC of the wireless interface to switch on
  3. Sends it to MN

```
INVITE sip:alice@open-ims.test SIP/2.0
```

```
...  
Content-Type: application/sdp  
Content-Length: 414  
  
v=0  
o=- 0 0 IN IP4 192.168.3.11  
s=IMS Call  
c=IN IP4 192.168.3.11  
t=0 0  
m=audio 10281 RTP/AVP 3 0 14 101  
b=AS:64  
a=rtpmap:3 GSM/8000  
a=rtpmap:0 PCMU/8000  
a=rtpmap:14 MPA/90000  
a=rtpmap:101 telephone-event/8000  
a=fmtp:101 0-11  
a=curr:qos local none  
a=curr:qos remote none  
a=des:qos mandatory local sendrecv  
a=des:qos mandatory remote sendrecv  
a=switchoninterface:00:04:23:5E:48:DE 192.168.125.2
```

- **Modified INVITE** message
  - SDP part of the INVITE message as modified by ASPM
  - **Note our application parameter a:... field at the end of the message**



# Implementation Insights: IMS Client

```
void ims_process_incoming_invite(eXosip_event *je)
{
    ...
    sdp_message_t * sdp_message;
    eXosip_lock();
    sdp_message=eXosip_get_sdp_info(je->request);
    eXosip_unlock();
    switchOnAddress=extractPowerManAttribute(sdp_message);
    if(newInvite) ims_send_de_register();
    else { ... /* standard session invite management */ }
}

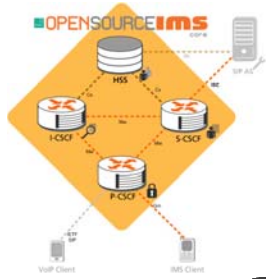
void ims_send_re_register()
{
    int port=5060, pid, status;
    pid=fork();
    if(pid==0) { // child
        if(!is_bye) { execl("../scripts/switchOnInterface.sh",
            "switchOnInterface.sh", switchOnAddress, (char *)0 ); }
        else { execl("../scripts/switchOffInterface.sh",
            "switchOffInterface.sh", switchOnAddress, (char *)0 ); }
    } else { // parent
        wait(&status);
        if(!is_bye) { while( eXosip_listen_addr(IPPROTO_UDP, switchOnAddress,
            port, AF_INET, 0) != 0 ) port++; }
        else { while( eXosip_listen_addr(IPPROTO_UDP, alwaysOnAddress,
            port, AF_INET, 0) != 0 ) port++; }

        ims_send_register();
    }
}
```

- Based on **UCT IMS Client**
- New-INVITE processing
  - **Extracts** from the “a:...” field **the MAC of the wireless interface to switch on**
  - Sends de-REGISTER
- Re-registration
  - **Switches on the wireless interface**
  - **Sends a REGISTER** over the switched-on wireless interface



# Implementation Details



ASPM

## UCT IMS Client

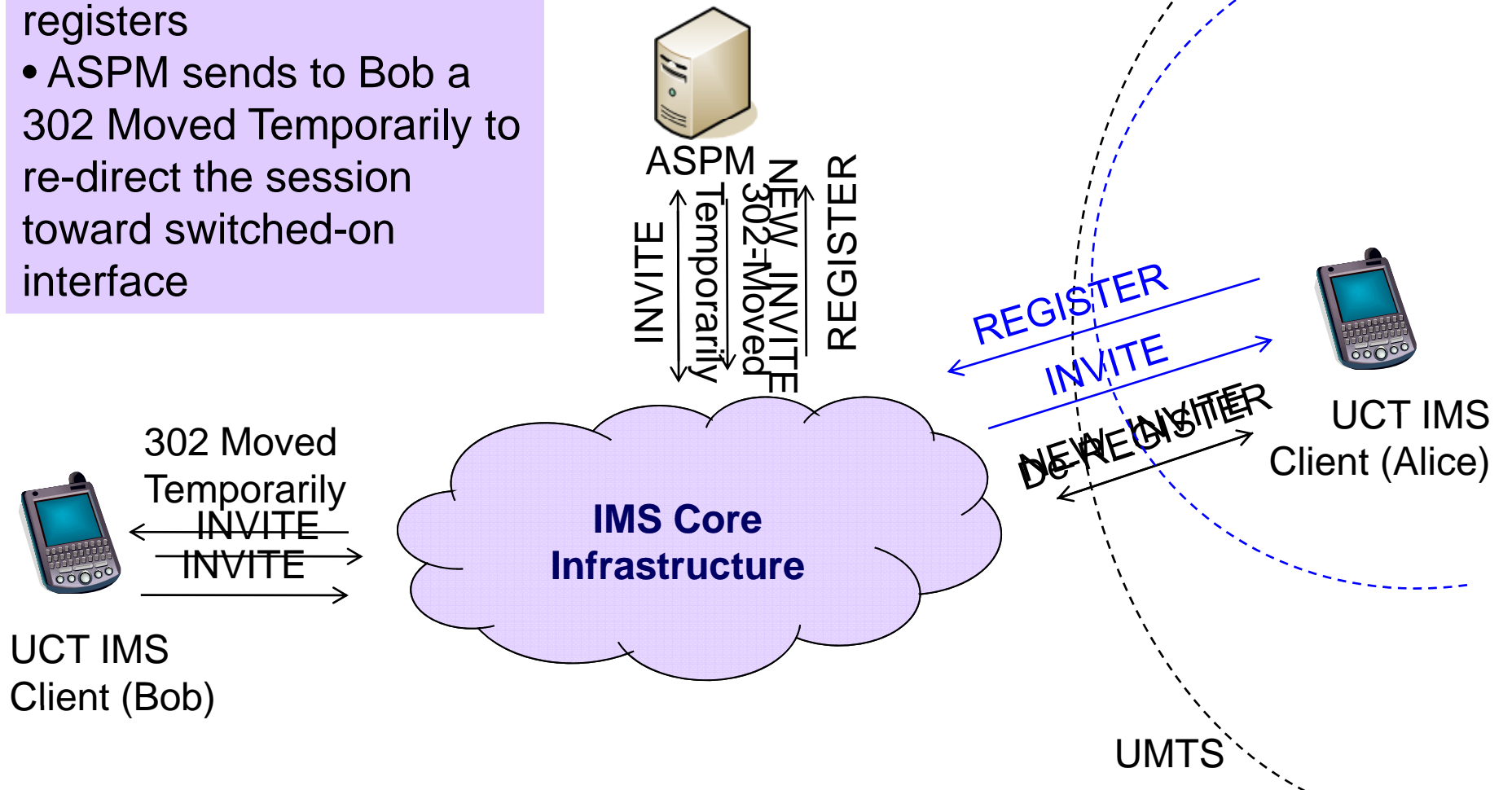
- IMS core components
  - OpenIMSCore (Fokus)
  - initial Filter Criteria (iFC) for IMS message re-routing
- ASPM
  - Java NIST JainSIP implementation of the SIP stack
  - OpenIMSCore properly configured to interpose ASPM in the signaling path
- University of Cape Town (UCT) IMS Client
  - CM: `iwconfig` and `hcitool` (Linux), NDIS (WIN)
  - CM integration with UCT IMS Client
- Deployment environment
  - Client: Linux laptops with 3G UMTS adaptor and IEEE 802.11b Cisco card
  - P-/I-/S-CSCF run on PCs: 2 CPUs 1,80GHz, 2048MB RAM, Linux Ubuntu
  - Wireless infrastructures:
    - Wi-Fi: Cisco Aironet 1100 AP

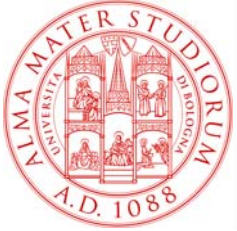


# Experimental testbed

- Alice switches on the WLAN interface and re-registers
- ASPM sends to Bob a 302 Moved Temporarily to re-direct the session toward switched-on interface

Adds our power saving "a:" field





# Experimental Results (1): some elements about the testbed

## Experimental testbed:

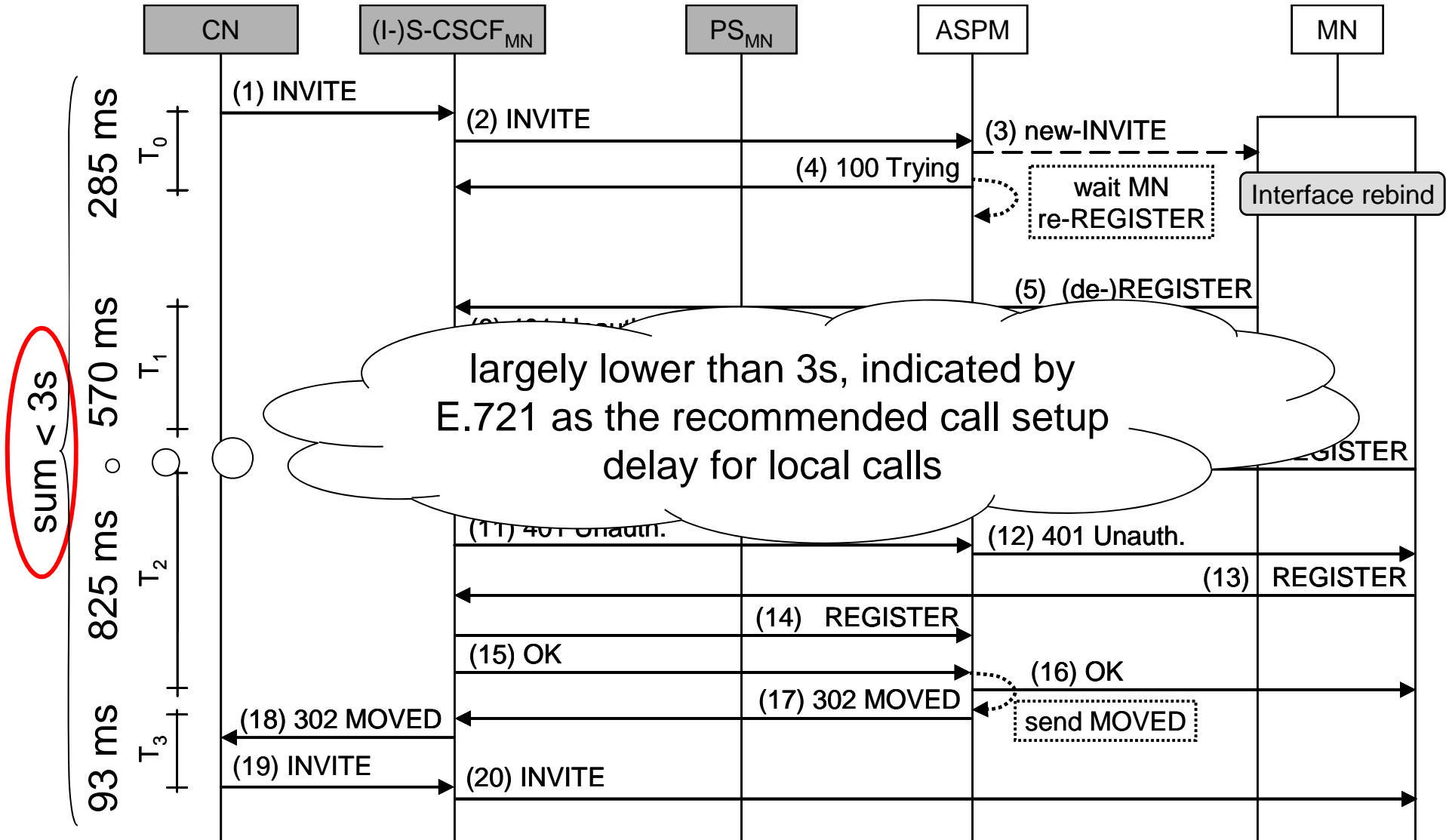
- ❑ Linux boxes for the IMS infrastructure (1.8GHz CPU and 2GB RAM)
- ❑ **Nokia N95** as reference client device (one of the first smart phones with WiFi)
- ❑ **VoIP service with GSM-encoded** audio, frame rate = 50 frames/s
- ❑ Reported results are averaged over 1000 session initiation cases

## Session Initiation Delay Analysis:

- T0: initiation phase
- T1: mobile node de-registration
- T2: mobile node registration for WiFi
- T3: MOVED notification



# Experimental Results (2): session initiation delay



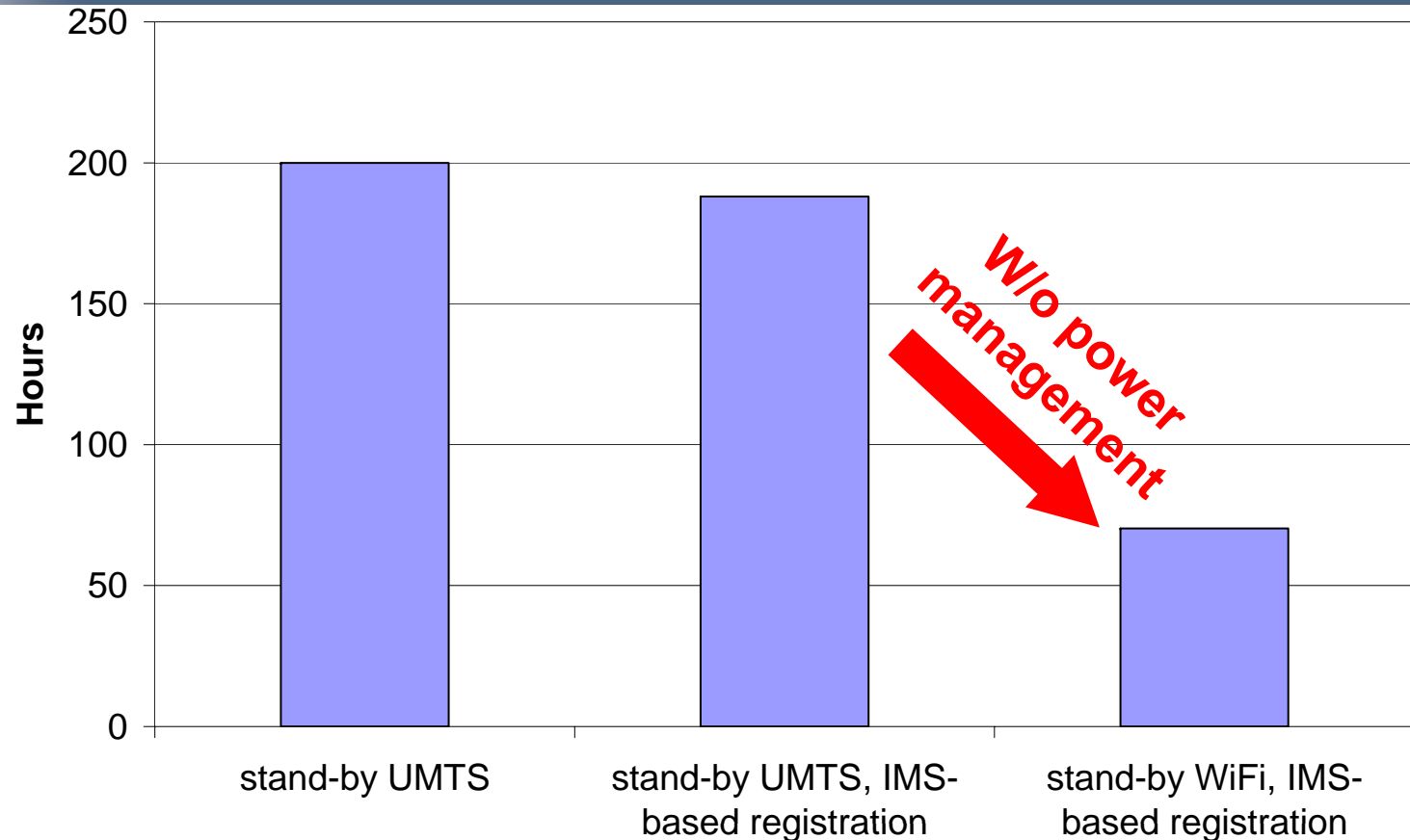


# Experimental Results (3): Stand-by time

## **Nokia N95**

WiFi and  
UMTS  
standby

These standby  
costs are  
completely  
eliminated by  
our IHMAS  
power  
management  
technique

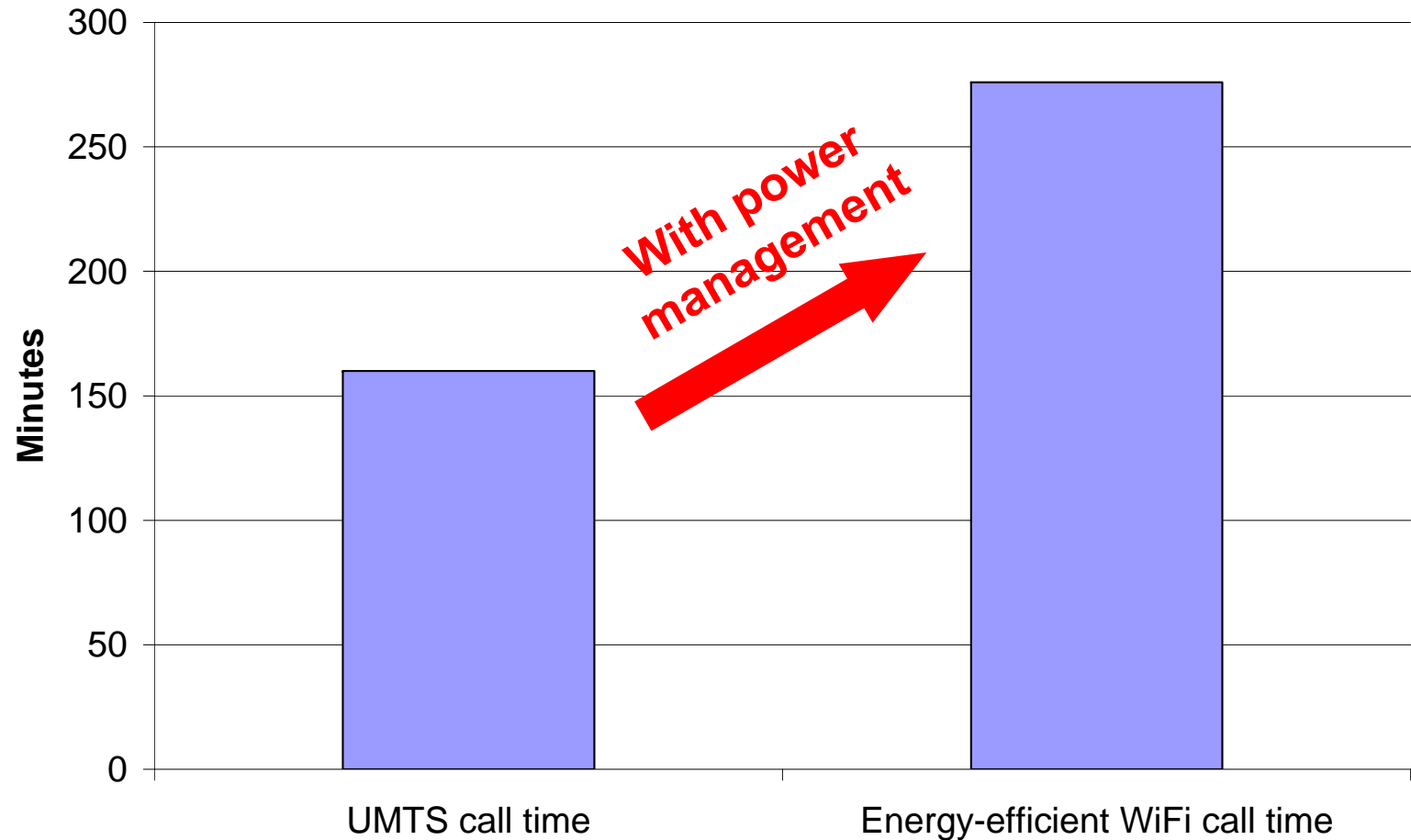


- Analytical results evaluated for N95 (also based on Arjona experience and measurements)
  - Battery: 950mAH and charged at 3.7V
  - WiFi average additional consumption (always-on): 0.05W

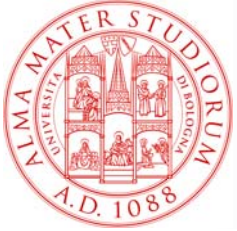


# Experimental Results (4): Call time

In addition,  
**significant  
advantages  
on call time**



- Energy-efficient WiFi call time is longer than the talk time duration specified by Nokia for UMTS



# Conclusions and ongoing work

## ■ Conclusions

- **Feasibility** of IMS-based standard approaches also for **context-aware power management**
- Practical guide on how to exploit IMS ASs to support application-level management functions
- Energy-saving techniques can **relevantly increase battery lifetime** when using high-consumption and low-cost wireless interfaces
- **Session invitation delays** introduced by the IHMAS facility for power management are **compatible even with strict VoIP reqs**

## ■ Ongoing work

- Scalability issues of IMS PS (see our article on IEEE Wireless Comm in June)
- J2ME version of CM and IMS client
- Extensive measurements of energy consumption in wide-scale deployment environments



# IHMAS project Web site and contacts

- Prototype code: <http://lia.deis.unibo.it/Research/IHMAS>
- Contacts: Luca Foschini ([lfoschini@deis.unibo.it](mailto:lfoschini@deis.unibo.it))



Any questions?

Thanks for your attention!