System Level Simulation of Relay Enhanced Cells

Prerequisites and Difficulties

28. Treffen der VDE/ITG-Fachgruppe 5.2.4
Mobilität in IP-basierten Netzen

Chair of Communication Networks

RWTH Aachen University

Maciej Mühleisen, Arif Otyakmaz

{mue, aoz}@comnets.rwth-aachen.de

Overview

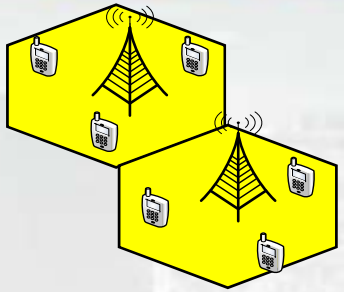
- System Level Simulation
- Channel Modeling
- Relay Capable Protocol Stack
- Multi-hop Scheduling
- Roadmap to Open Source

System Level Simulation

- Discrete Event Driven Simulation:
  - Events occur at certain points in simulation time

    => ex. traffic generator, ARQ timeout
  - Programming code is executed

    => Can generate new events for now or later
- (Wireless) Network Simulators
  - Scenario: Position & mobility of nodes, obstacles, channel physics
  - Composition: Protocol stack within each node
  - Functionality: Code handling events
  - Helpers: Event queue, random number generators, statistical evaluation, ...



HTML
TCP
IP
802.11 MAC
802.11g Phy

```
onTimeout(PDUNr)
{
  pduCopy = q.copy(PDUNr);
  send(pduCopy);
  setTimeout(PDUNr, to);
}
```

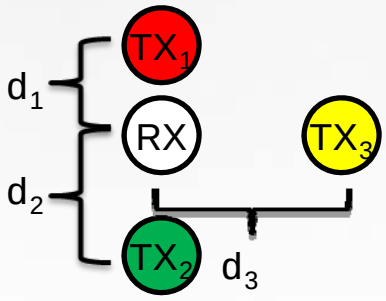
\+ Detailed results

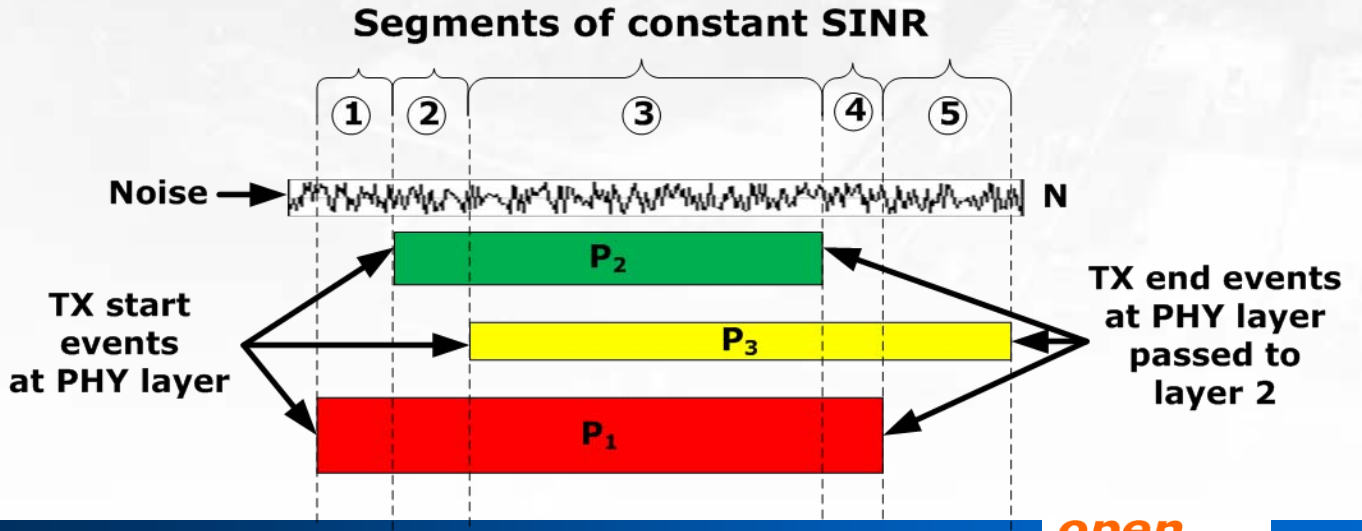
\- Implementation effort

\- Simulator runtime

Channel Modeling

- Modeling signal to interference and noise ratio (SINR)
  - Noise N: fixed / bandwidth dependant
  - Signal strength S:
    - Path loss (distance / propagation factor)
    - Shadowing (obstacles or random model)
    - Fast fading (random model)
  - Interference: Calculated like S for every simultaneous transmission
  - TX / RX antenna gains: From geometry
  - Weight each *segment of constant SINR* with its duration

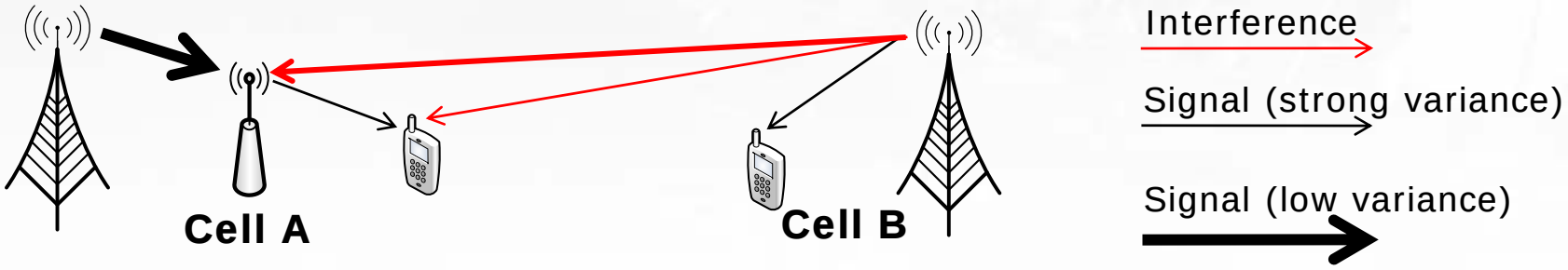




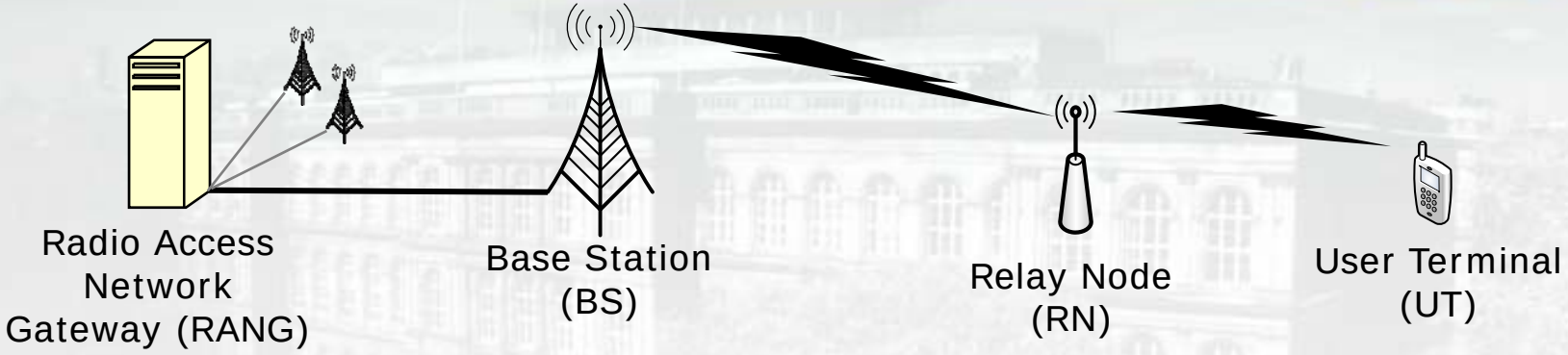
Channel Modeling cont.

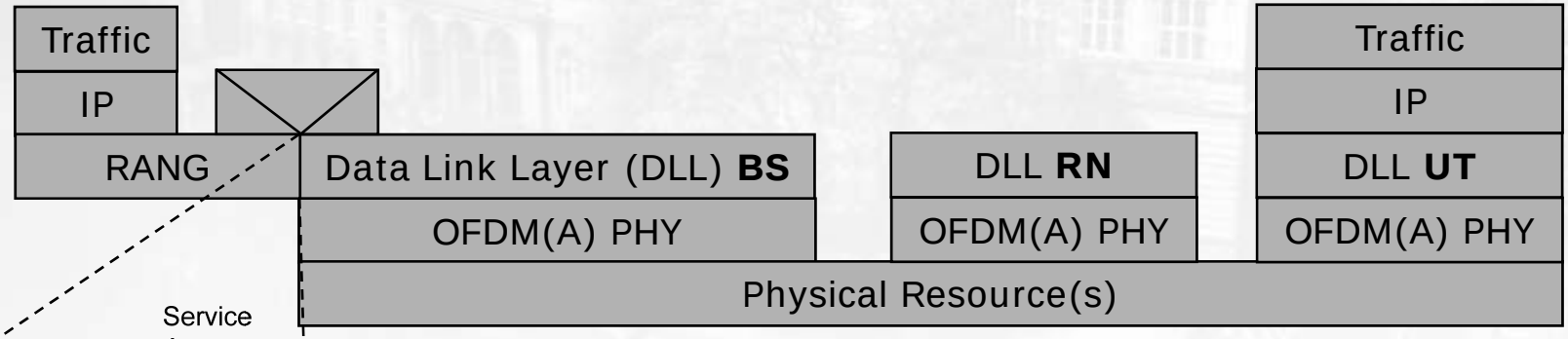
- Different channel models for different **node type pairs**
  - Relay Node (RN) positions are subject to network planning
  - They are most likely positioned line of sight to Base Station (BS)
    - Low (fixed) path loss / low propagation factor
    - Low / no shading
    - Low / no fast fading
  - Directional antennas and beamforming can be used BS<->RN
  - BS<->User Terminal (UT), RN<->UT and UT<->UT channels are more unpredictable
  - Channel between RN/BS and other cell RN/BS is more unpredictable

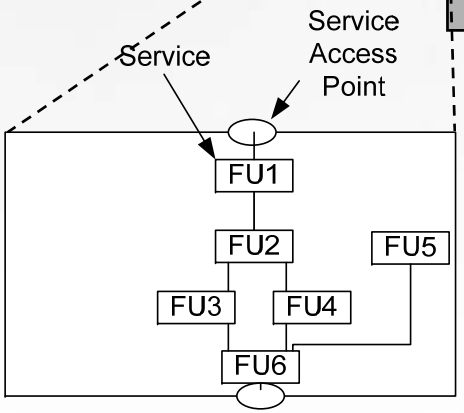
=> Higher data rates for BS<->RN channel + lower error rate



Relay Enabled Protocol Stack



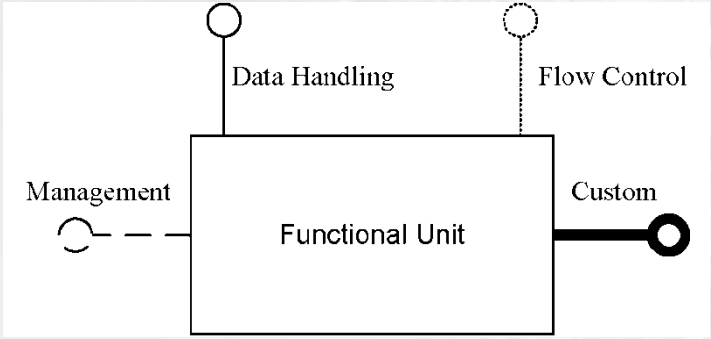


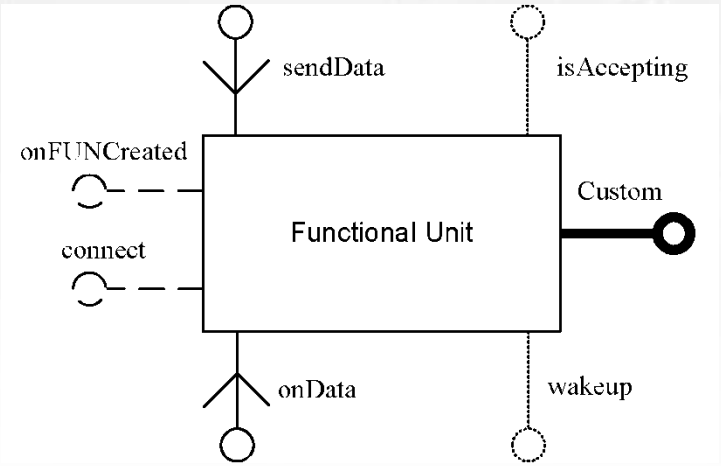


- Mostly general purpose protocol layers
- Technology and node type specific DLL
- Each layer consists of multiple Functional Units (FUs)

- RANG allows evaluation per cluster (multiple cells)

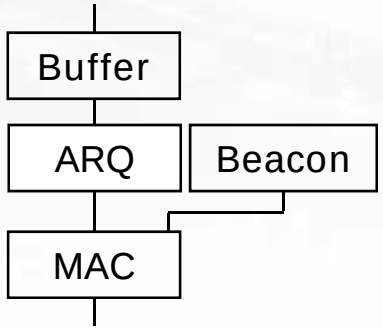
Relay Enabled Protocol Stack cont.





Functional Units (FUs):

- Goal: Rapid protocol implementation
- Perform subtask of the layer (Buffer, ARQ, MAC, segmentation, concatenation, CRC...)
- Are connected to form FU Network (FUN)
- Pass PDUs up and down the stack
- Can modify, buffer, drop or inject new PDUs
- Flow control decides when to pass PDUs
  - isAccepting: Can I pass the PDU down?
  - wakeup: My state has changed, I can accept now

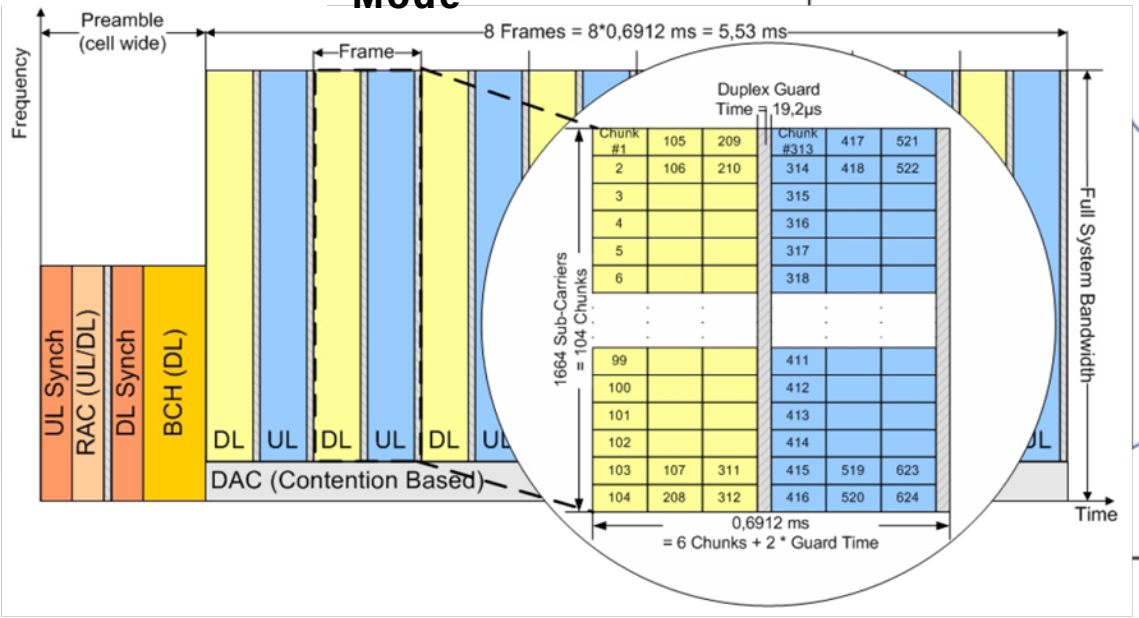


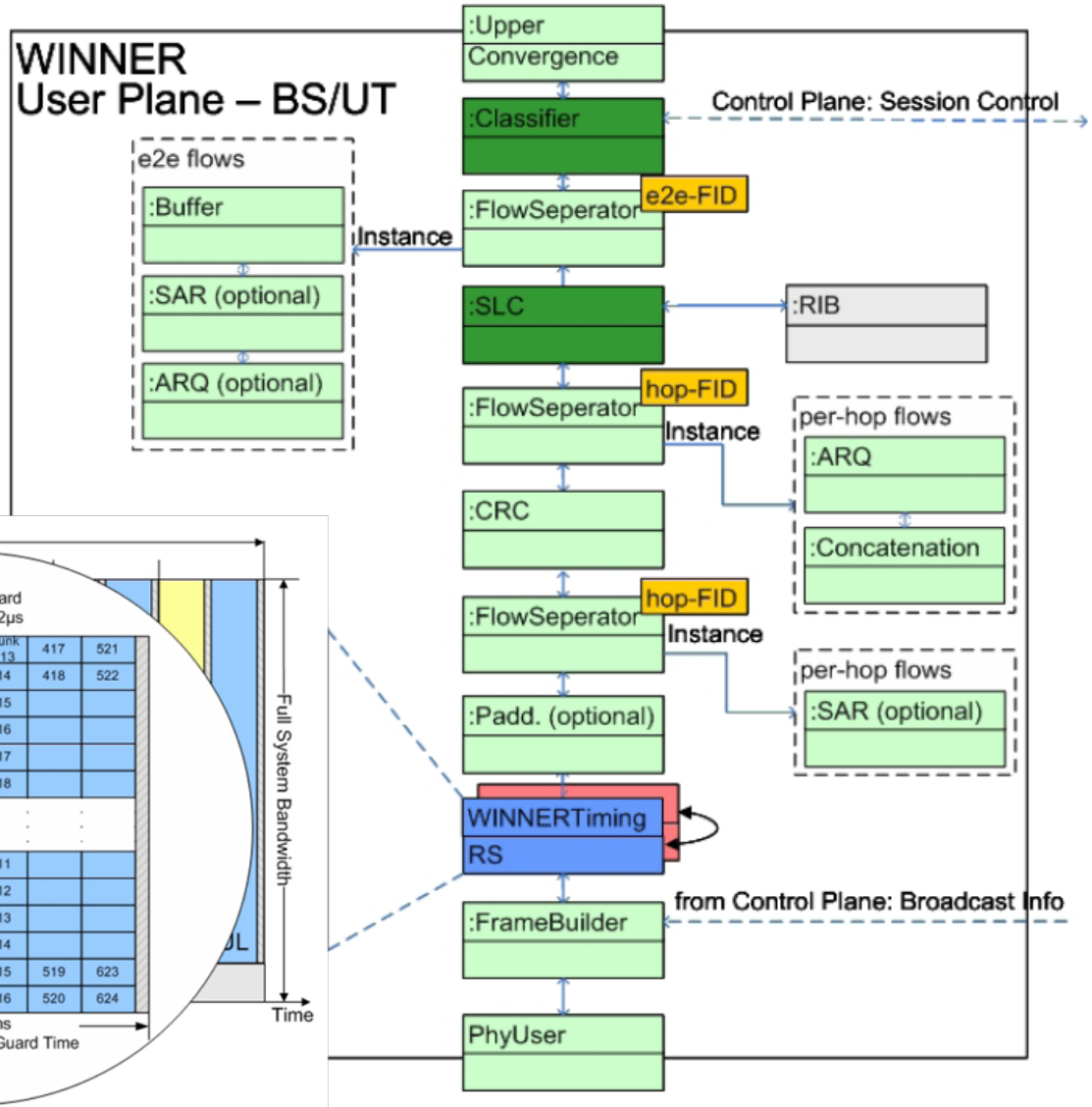
Relay Enabled Protocol Stack cont.

General purpose FU

Wireless World Initiative New Radio (WINNER)

Time Division Duplex (TDD) Mode



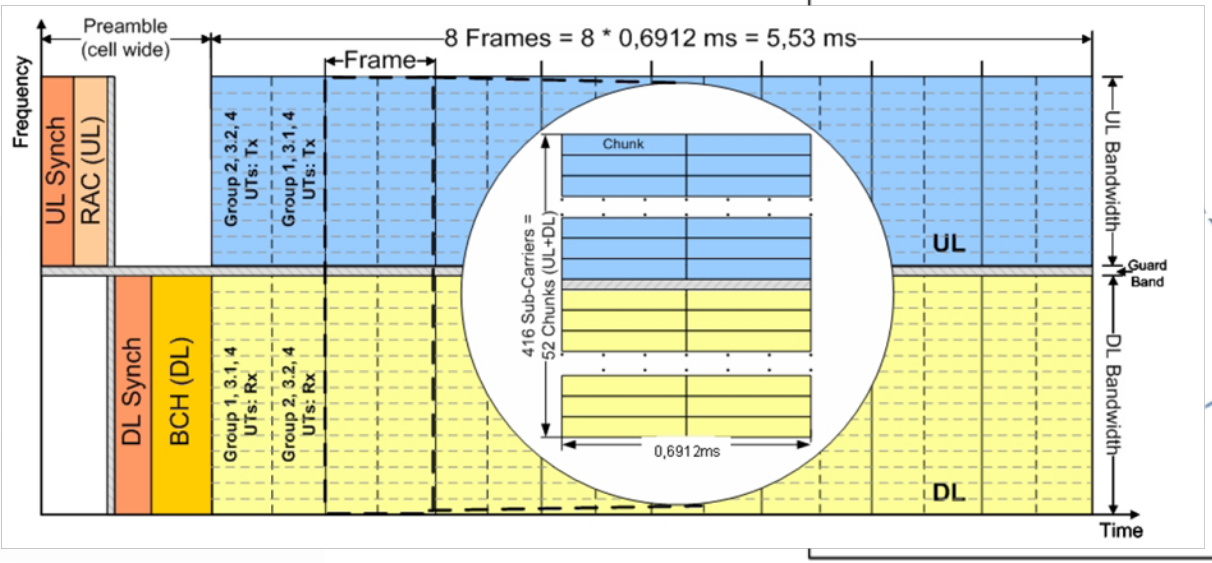


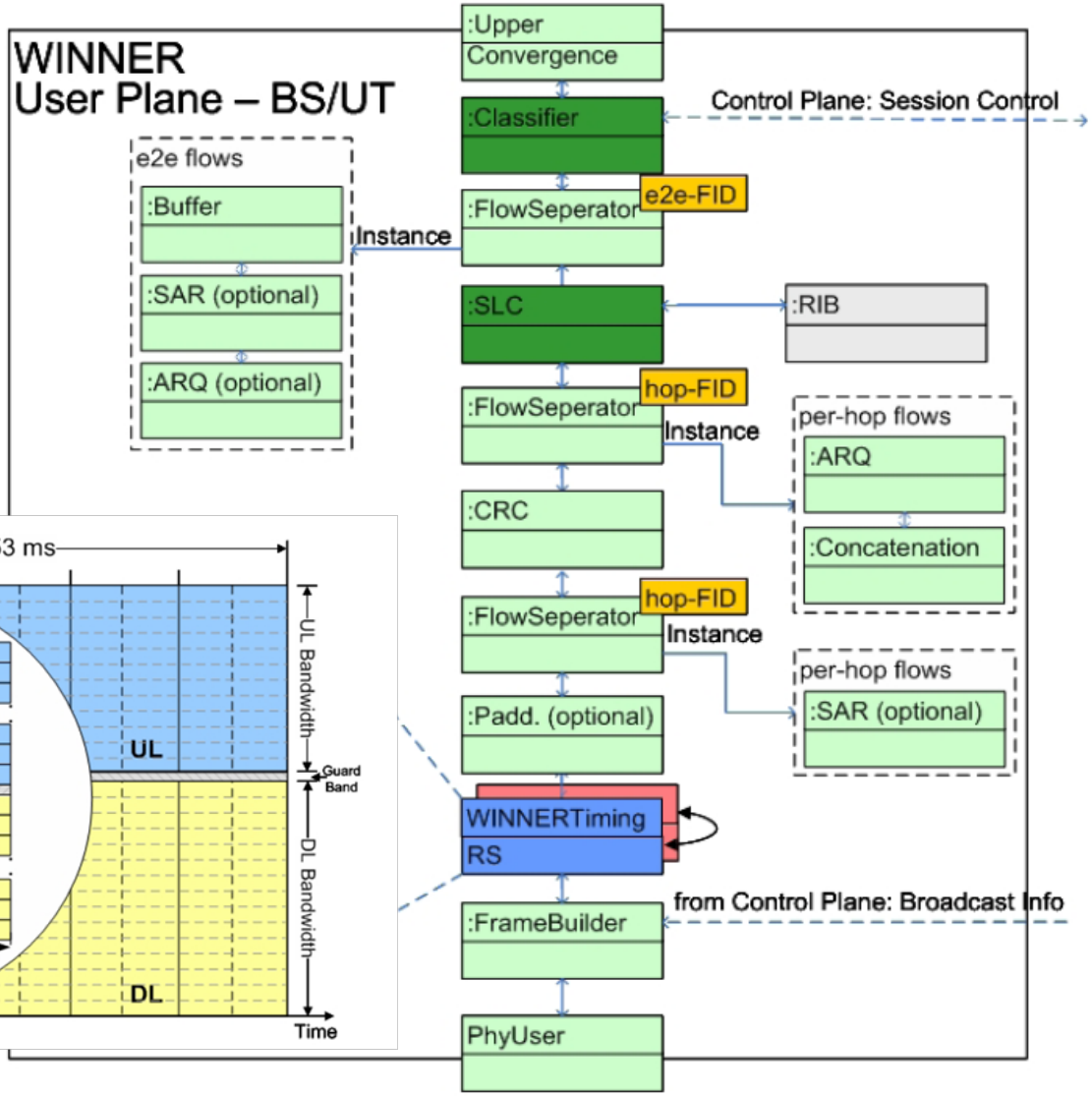
Relay Enabled Protocol Stack cont.

General purpose FU

Wireless World Initiative New Radio (WINNER)

Frequency Division Duplex (FDD) Mode

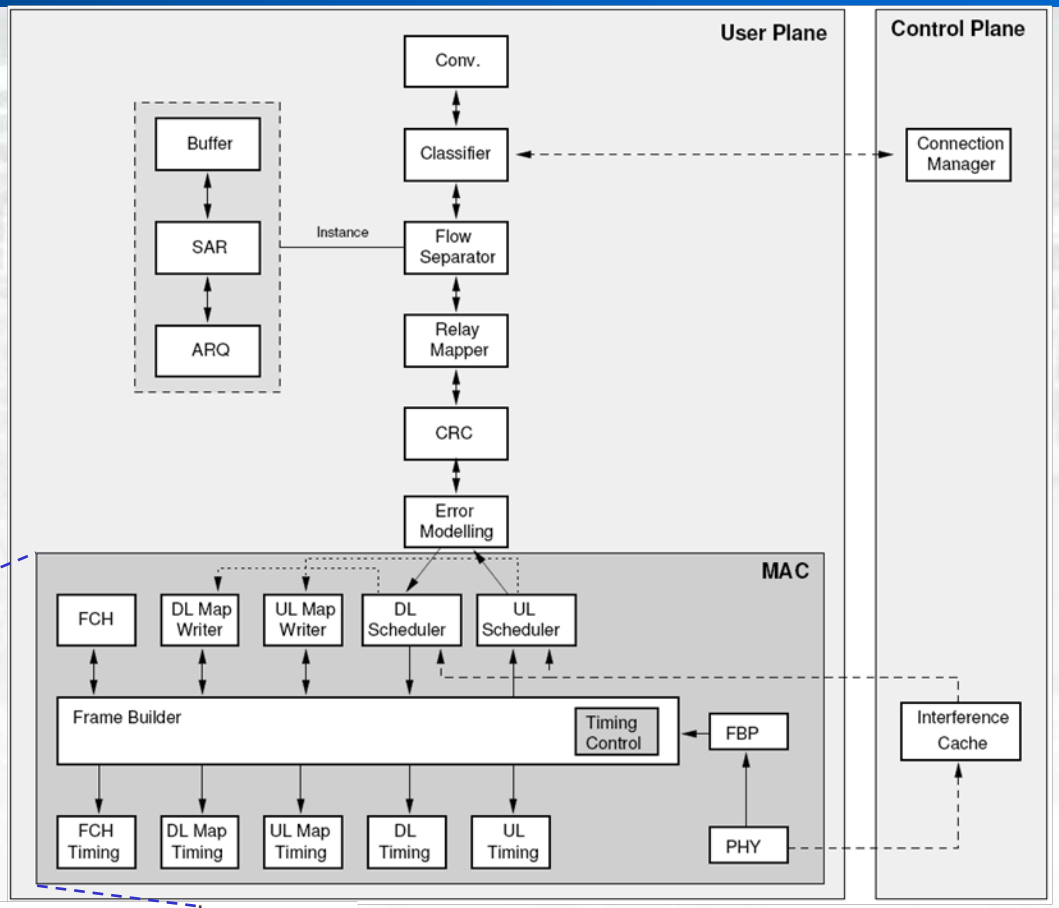


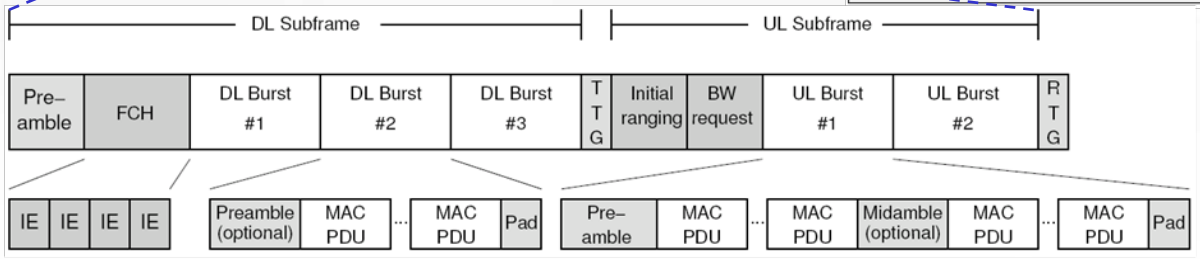


Relay Enabled Protocol Stack cont.

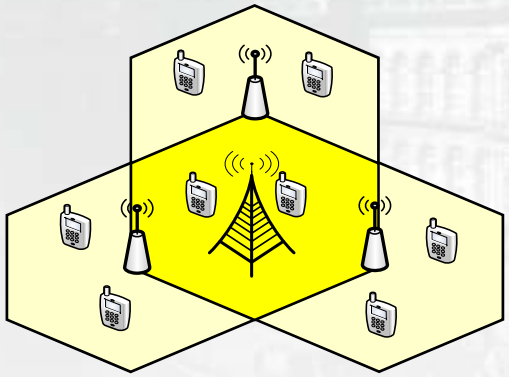
IEEE 802.16 WiMAX

TDD OFDM

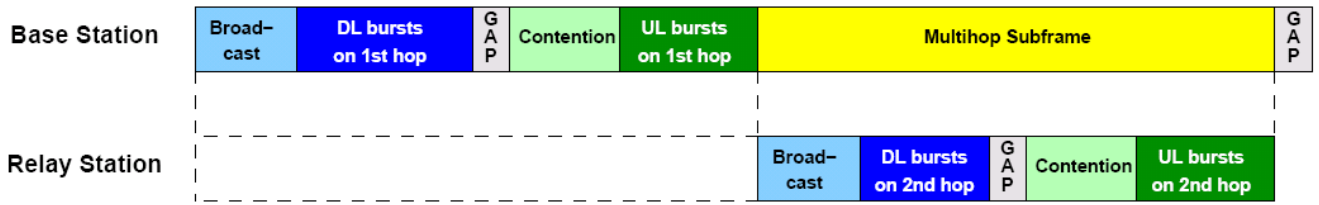




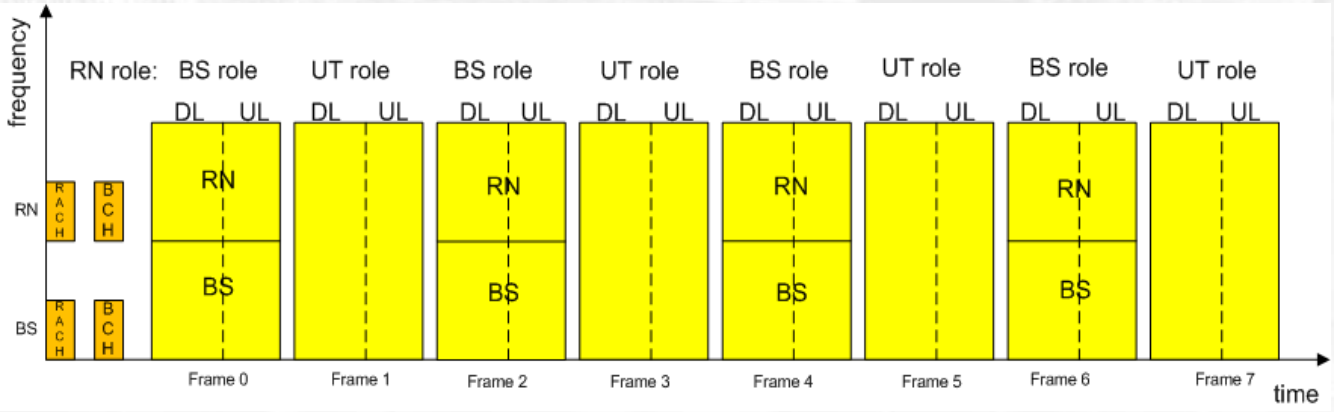
Relay Enabled Protocol Stack cont.



WiMAX Relay Concept



WINNER TDD Relay Concept

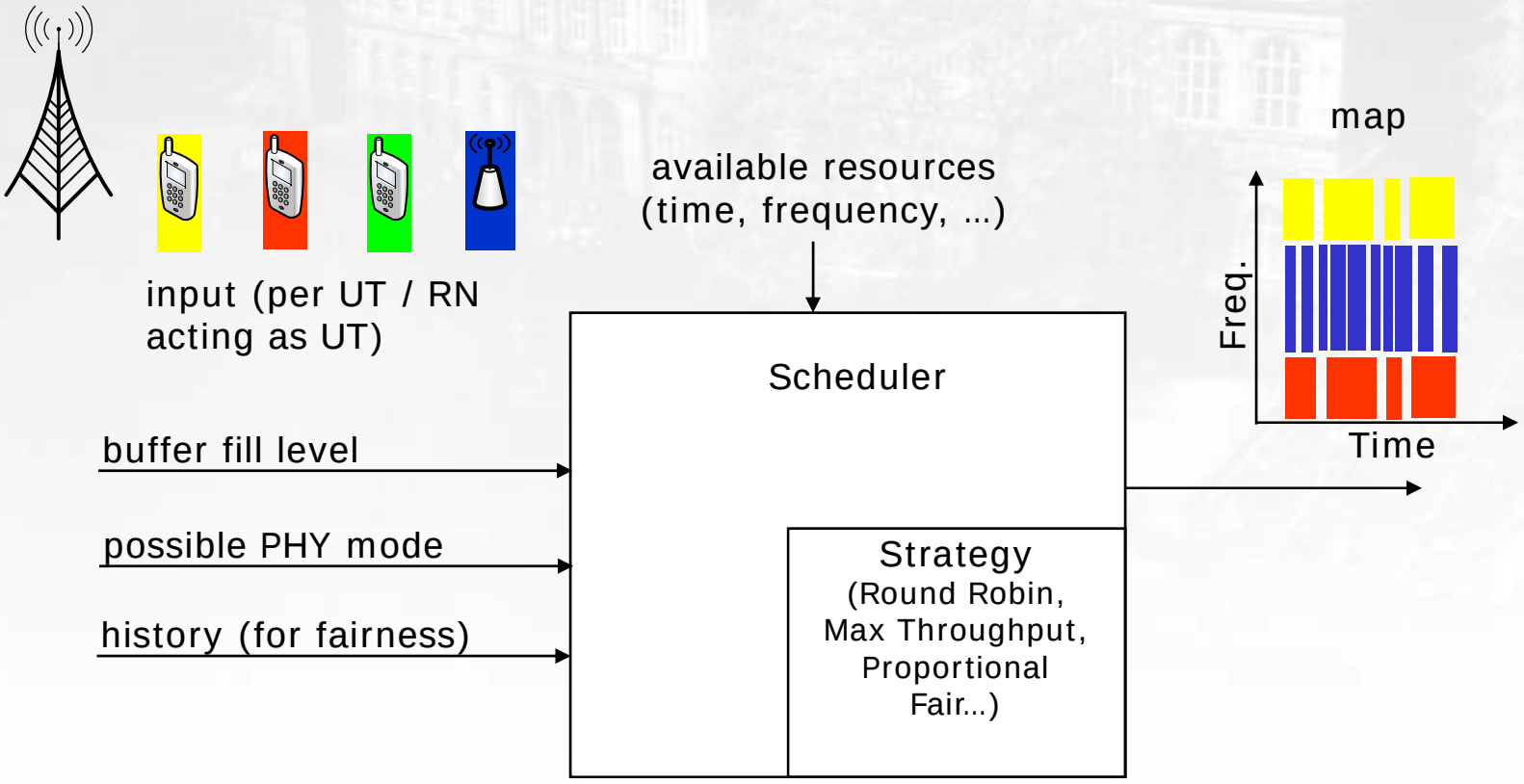


Resource Partitioning:

- 1st hop BS<->RN: RN is scheduled like normal (very busy) UT
- How to split resources between BS and RN operating as BS?
- Can we adjust it dynamically?
- Multiple RNs: Further partitioning vs. spatial reuse

Scheduling

- Downlink (DL) and uplink (UL) scheduling
- Done by BS / RN acting as BS
- Scheduler creates UL and DL schedule (map) and broadcasts it
- RN needs more resources in BS schedule

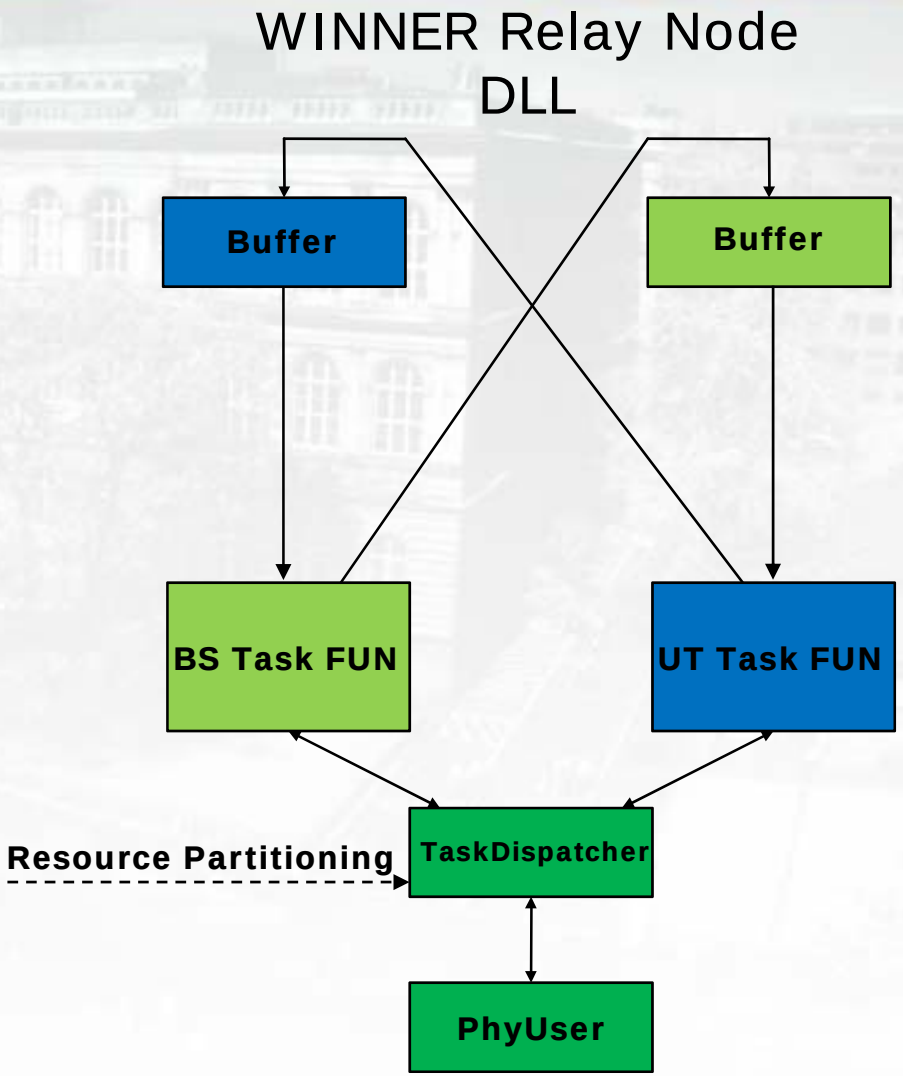


Scheduling cont.

Relay Node FUN:

- Two FUNs: 1 for UT, 1 for BS task
- *TaskDispatcher* switches between tasks
- PDUs need to be buffered
- UT task
  - Receive map from BS
  - Send PDUs in assigned resource
  - Turn on receiver as indicated by map
- BS task
  - Schedule resources for UTs
  - Estimate channel to/from each UT
  - Broadcast map
- Problem: BS scheduling 1st hop does not know channel on 2nd hop

=> Possible buffer overflow in RN



Scheduling cont.

What is the impact of this problem?

- Resources wasted on 1st hop
- High buffer levels at RN cause high delays
- Maximal possible throughput is underestimated
- Spectral efficiency is underestimated

Possible solutions:

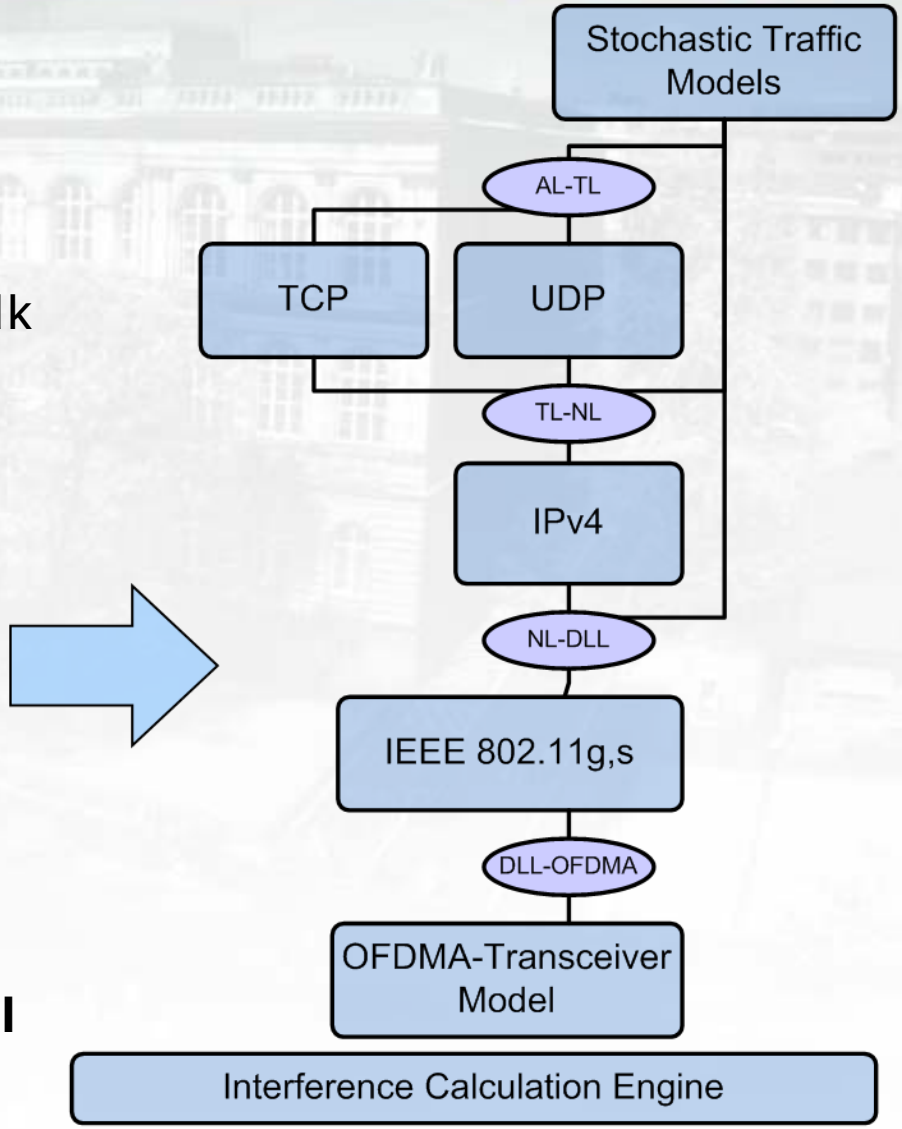
- Uplink: RN knows how much resources it gets from BS, can schedule its UTs accordingly
- Dynamic resource partitioning: Match 1st and 2nd hop resources / capacities
- Signaling: Inform BS about channel at 2nd hop
- Flow control:
  - Use some sliding window mechanism (ex. ARQ)
  - XON / XOFF

⇒ Open research topic

Roadmap to Open Source

Open Source Wireless Network Simulator (openWNS)

- Available on Launchpad
  - www.openwns.org
  - www.launchpad.net/openwns-sdk
- **Lesser** General Public License (LGPL)
- Framework, traffic generators and simple transport layer are already published
- Published by the end of 2008:
  - Full IEEE 802.11g/n/**s** simulator
- **IEEE 802.3, IEEE 802.16, IEEE 802.21 & IST-WINNER / LTE will follow later**



Thank you for your attention!

Contact:
mue@comnets.rwth-aachen.de
info@openwns.org