Lessons Learned while Trying to Reproduce the OpenRF Experiment

Authorship: ***Mohamed Naoufal Mahfoudi***, Walid Dabbous ,Thierry Turletti and Thierry Parmentelat

Contact email: mohamed-naoufal.mahfoudi@inria.fr

General Outline

1. Context and Motivation
2. OpenRF: A Case Study
3. Reproducibility Challenges
4. Lessons Learned
5. Conclusion

Motivation

Learning

Learning with a hands-on experience

Extending

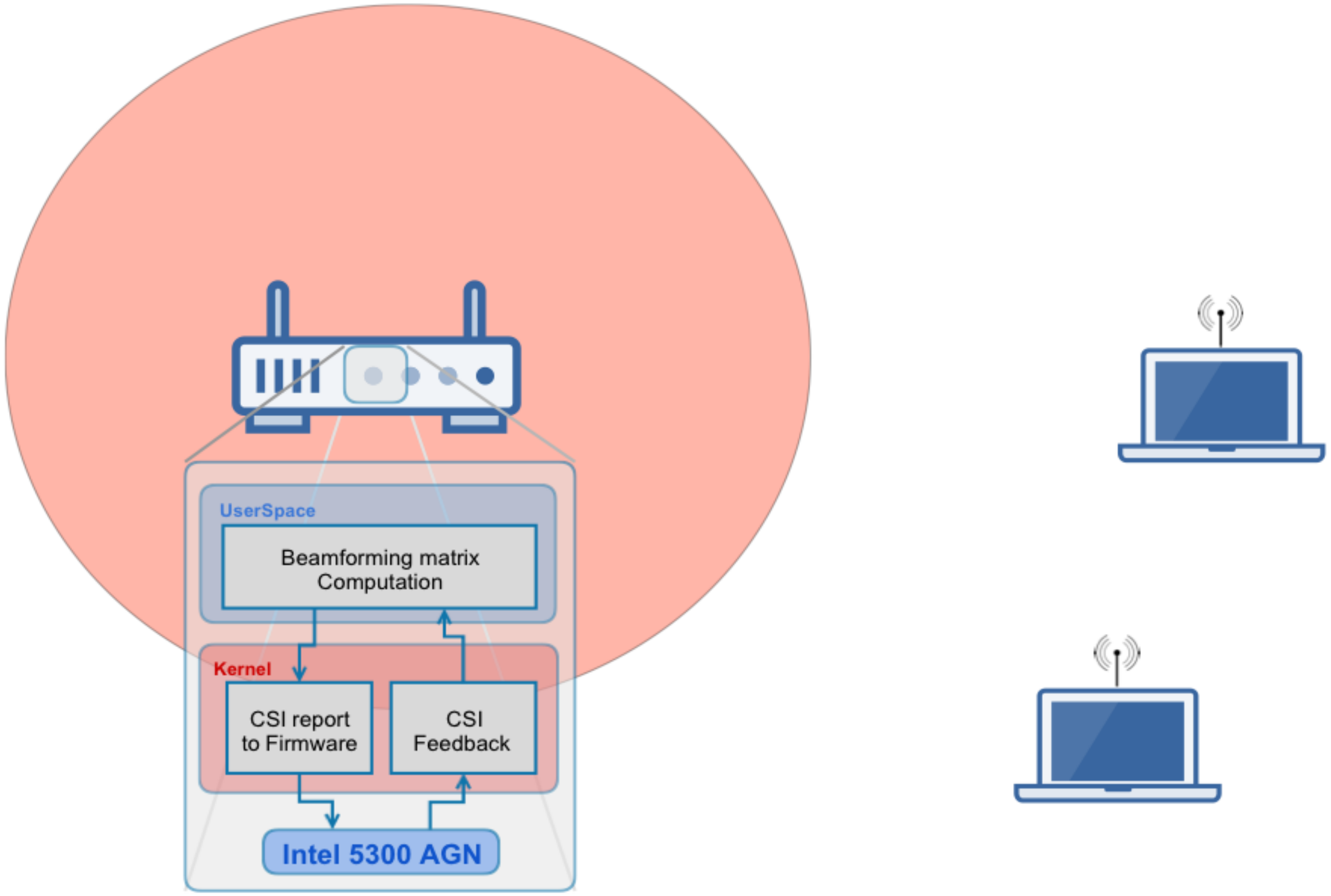
Extending scientific work

Reusing

Reusing tools for other scientific projects

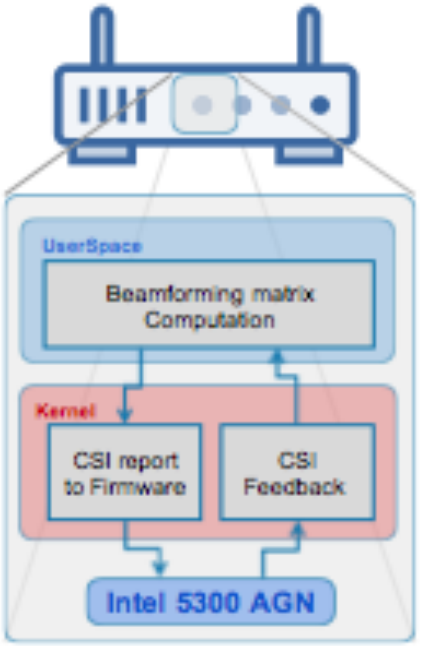
Beamforming

- Wireless channel reciprocity assumption
- Precoding Tx from CSI measurements upon the reception of a client packet



Beamforming

- Wireless channel reciprocity assumption
- Precoding Tx from CSI measurements upon the reception of a client packet



UserSpace

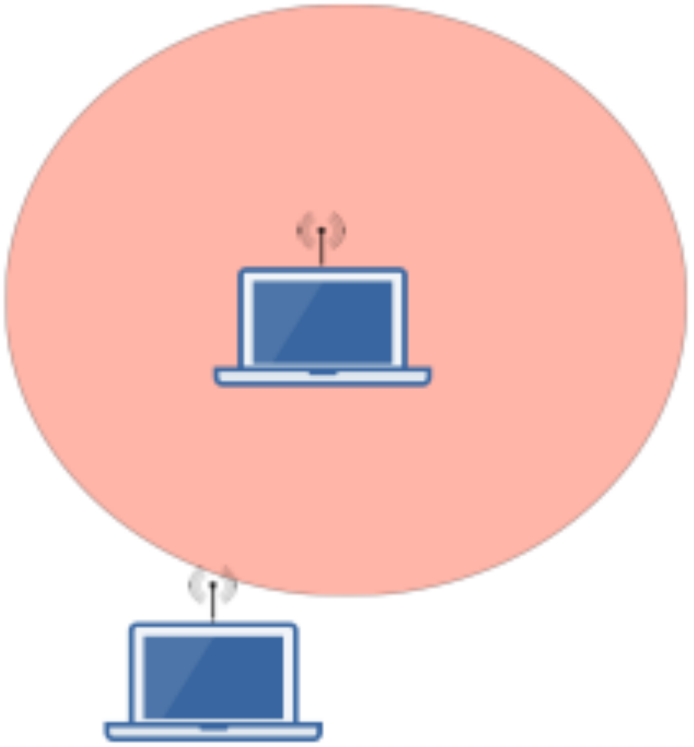
Beamforming matrix Computation

Kernel

CSI report to Firmware

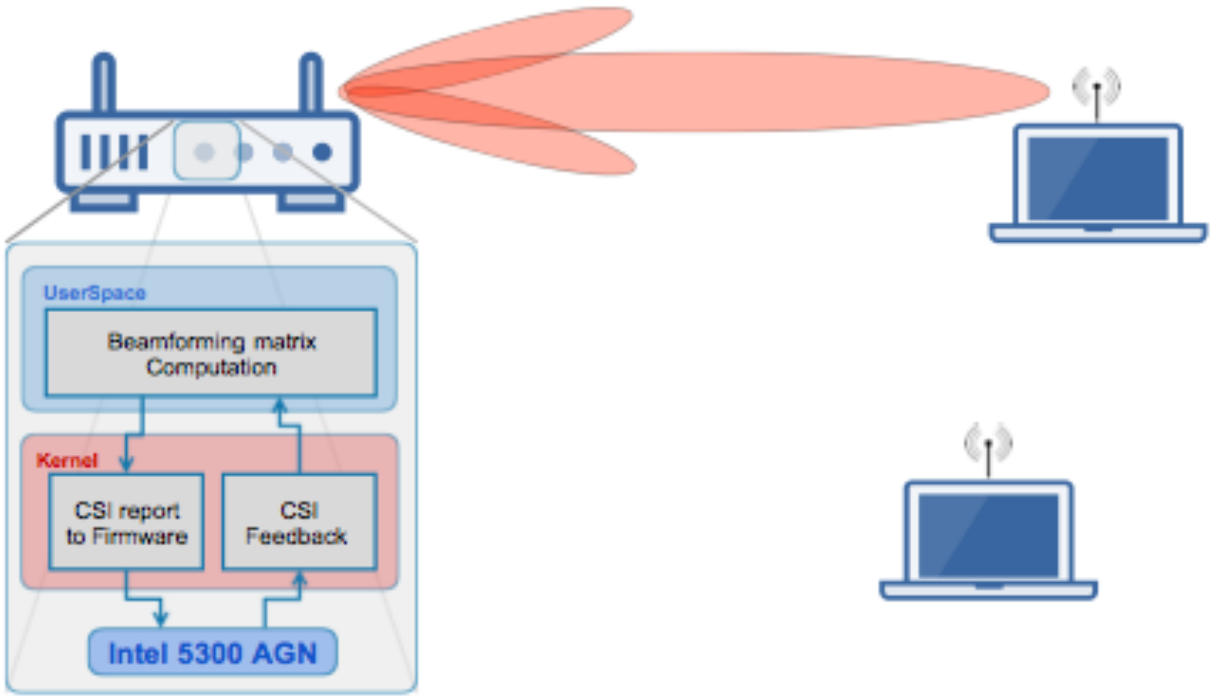
CSI Feedback

Intel 5300 AGN



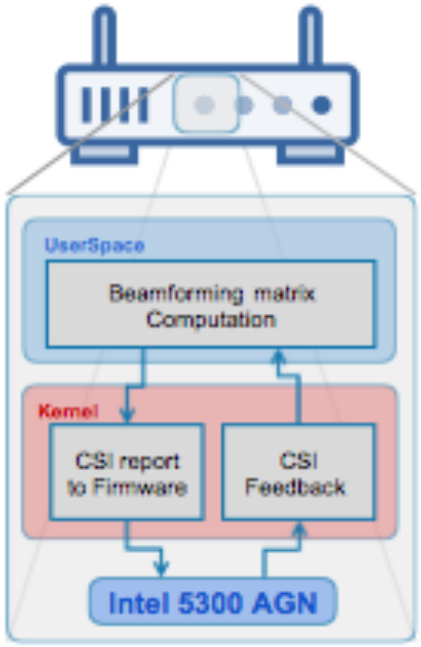
Beamforming

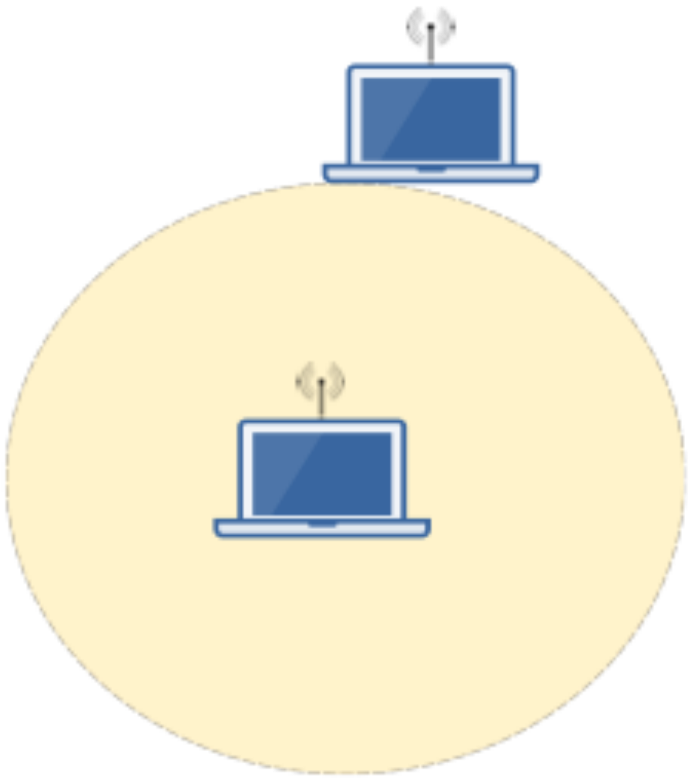
- Wireless channel reciprocity assumption
- Precoding Tx from CSI measurements upon the reception of a client packet



Beamforming

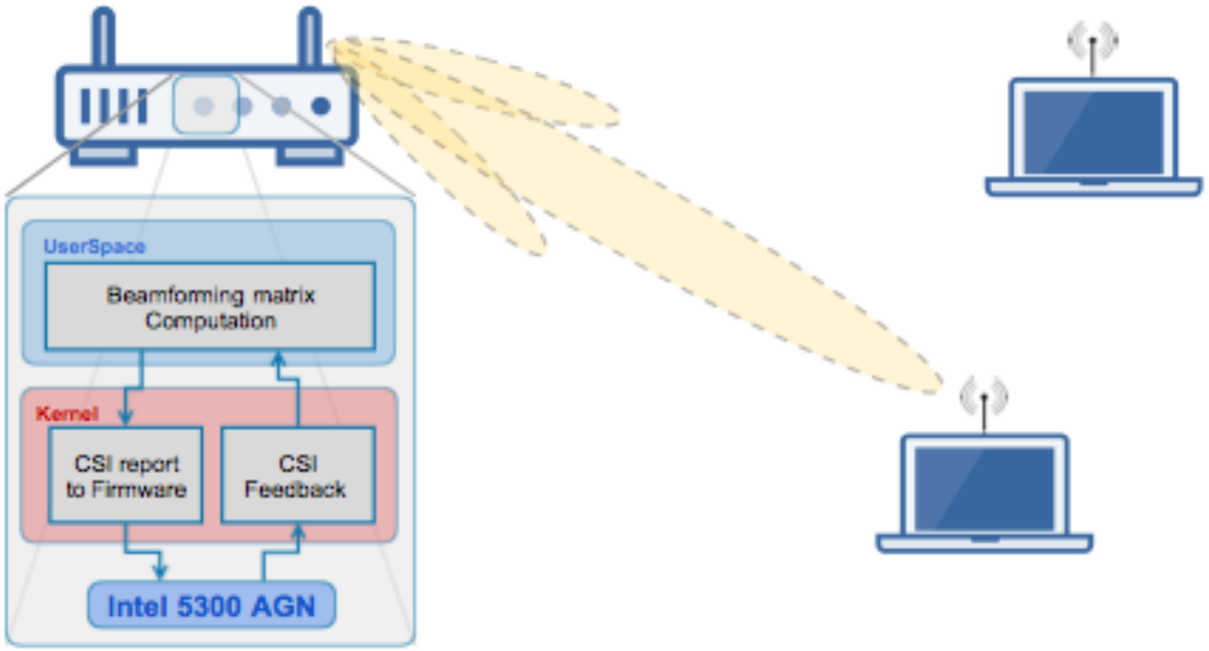
- Wireless channel reciprocity assumption
- Precoding Tx from CSI measurements upon the reception of a client packet





Beamforming

- Wireless channel reciprocity assumption
- Precoding Tx from CSI measurements upon the reception of a client packet



OpenRF[1]: A Case Study

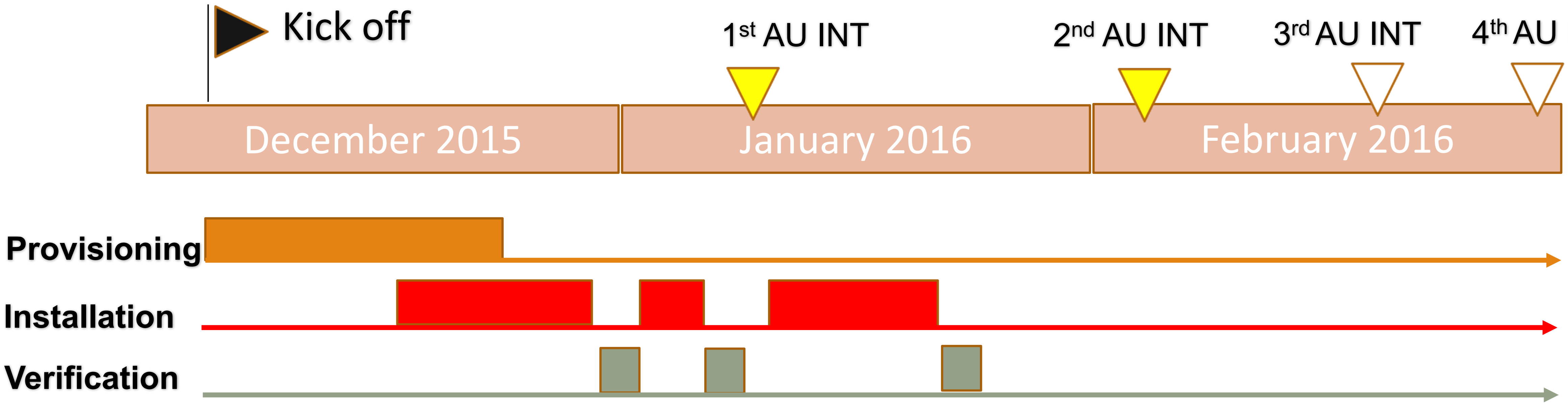
Learning	Extending	Reusing
Learning about the implementation of Beamforming in COTS Wi-Fi chips	Extending cross-layer optimization by physical layer manipulation	Reusing Acquired knowledge of the Intel CSI tool[2] and the Intel 5300 agn card for other related projects (Localization)

[1] Swarun Kumar and al. Bringing cross-layer mimo to today's wireless lans. SIGCOMM Comput. Commun. Rev., 43(4):387–398, August 2013.
[2] Daniel Chaim Halperin and al. Tool Release: Gathering 802.11N Traces with Channel State Information. ACM CCR, 41(1):53–53, Jan 2011.

Provided Material

- Web page containing:
  - OpenRF code release
  - Link to the ISO image used during the experiments (Ubuntu 10.04.4 LTS)
  - Installation instructions
  - A sample testing scenario with description of expected results

Project Timeline



RESSOURCES

- 45 Intel 533AN MMW Full / Half
- 3 x86 machines in office room
- 3 x86 machines in Anechoic chamber

INSTALLATION

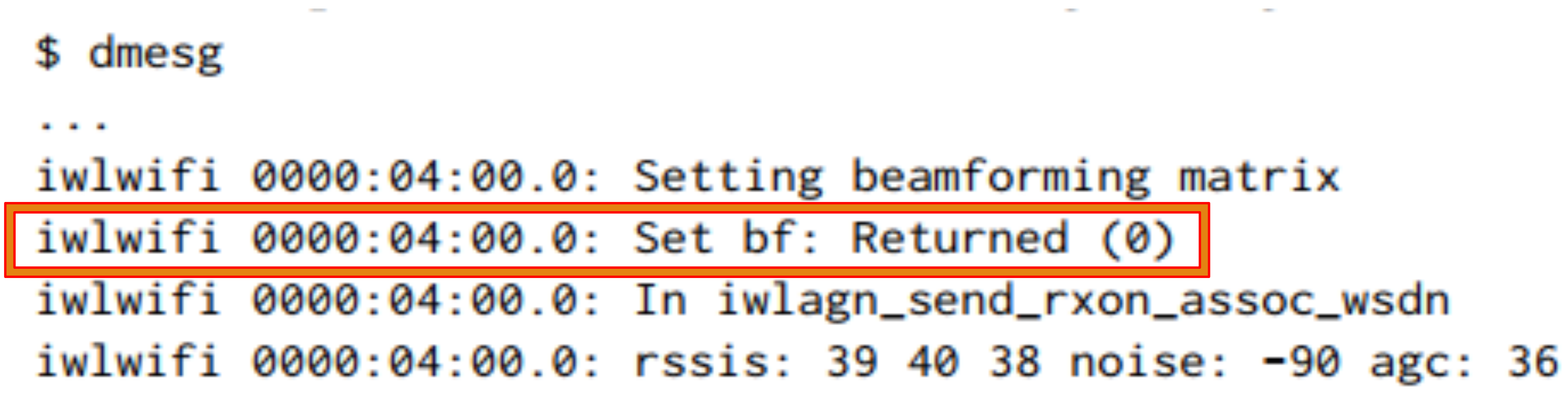
- Operating Systems:
  1. Ubuntu 15.04
  2. Ubuntu 12.04 LTS
  3. Ubuntu 10.04 LTS
- Kernel : 3.5.4-csitool+ kernel

VERIFICATION

- Checking Dmesg
- Plotting SNR from CSI
- PHY layer configuration

Initial Outcome

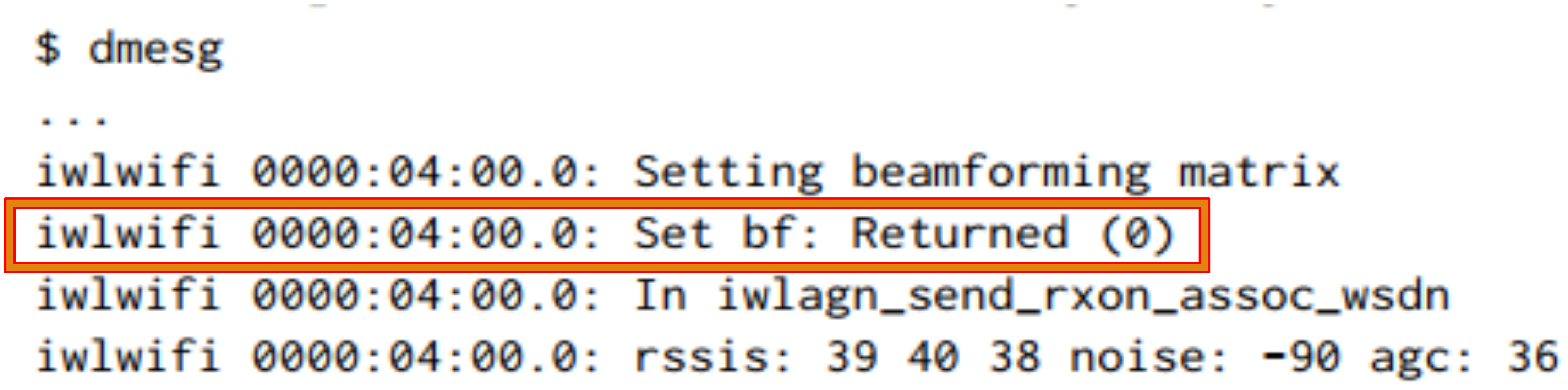
- Matrix injection and system response:
  - Driver message comes back with positive ACK
  - SNR plot for each subcarrier shows no difference between the before and after precoding



```
$ dmesg
...
iwlwifi 0000:04:00.0: Setting beamforming matrix
iwlwifi 0000:04:00.0: Set bf: Returned (0)
iwlwifi 0000:04:00.0: In iwlagn_send_rxon_assoc_wsdn
iwlwifi 0000:04:00.0: rssis: 39 40 38 noise: -90 agc: 36
```

Verification Procedure

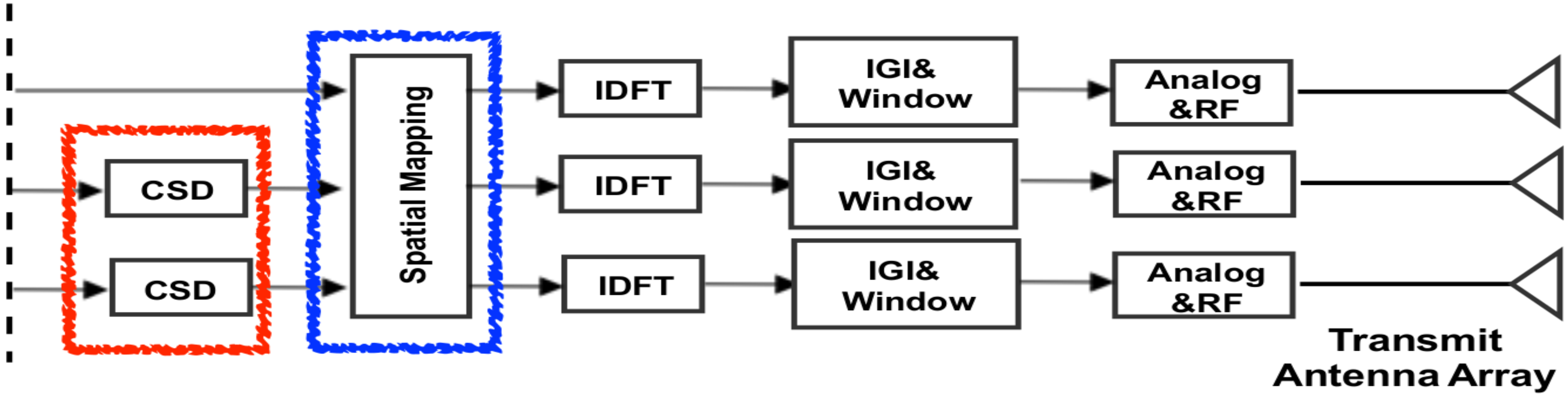
- Double check on every step from the matrix computation to the transmission of the CSI report to the Firmware
- Driver messages comes back clean with no Firmware crashes or injection rejection



```
$ dmesg
...
iwlwifi 0000:04:00.0: Setting beamforming matrix
iwlwifi 0000:04:00.0: Set bf: Returned (0)
iwlwifi 0000:04:00.0: In iwlagn_send_rxon_assoc_wsdn
iwlwifi 0000:04:00.0: rssis: 39 40 38 noise: -90 agc: 36
```

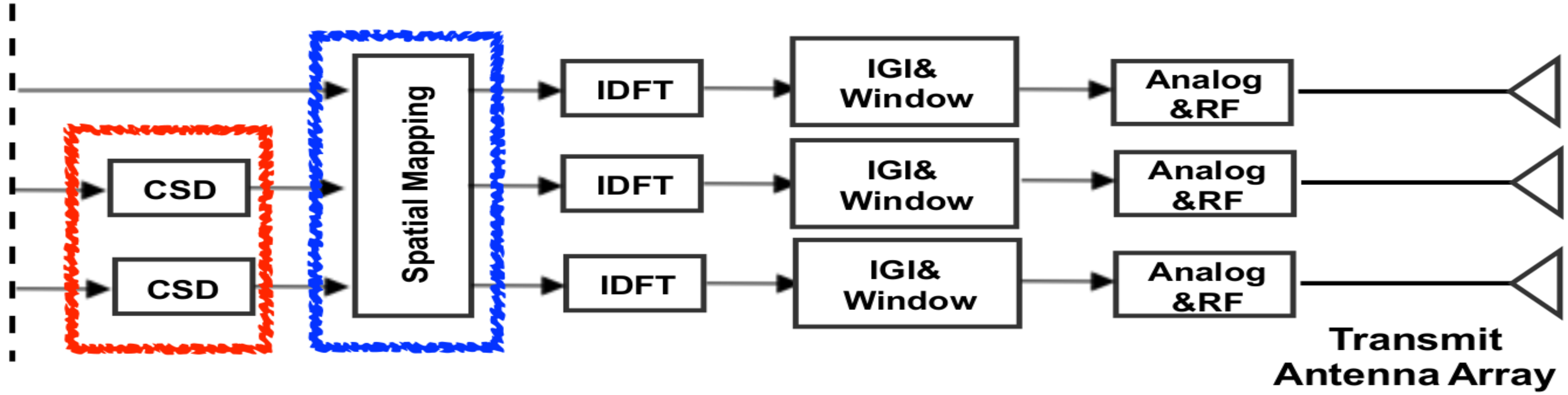
Verification Procedure

- Variation of Transmission scenarios:
  - Dynamic rate adaptation
  - Fixed number of space time stream (STS)
  - Fixed Modulation and Coding Scheme (MCS)



Interpretation

- The Spatial mapper is always set to the default Intel's indirect spatial mapping matrix
- Intel wireless cards vary from a manufacturer to another with some Engineering Samples (ES) still in the market



Lessons Learned

- It would preferable to:
  - Provide a detailed description of their experiment scenario
  - Publish the code and scripts used for their experiments

Lessons Learned

- It would be preferable to consider managing research projects as a Software development projects:
  - Similar reproducibility challenges
  - Source code management tools and notebooks are great for fostering a more reproducible research

Requirement Description

- System Description should focus on the critical details and particularities of their setup
- A shallow and lengthy description should be traded for a focused on critical requirements
- A system verification against multiple hardware should be conducted or at least the specific references should be mentioned

Code and Scripts

- The code should be:
  - Freely accessible
  - Possibly written in non propitiatory languages
  - Depending on accessible and properly maintained packages

Conclusion

- Reproducibility is tied to information accessibility
- Reproducibility should be accounted as a part of the research project’s activities