DESIGN AND IMPLEMENTATION OF A MAC-PHY-API FOR BETTER WAVEFORM PORTABILITY

Robert Koch, Martin Kohlmann, Ewald Hedrich, Jochen Martin-Creuzburg
Fraunhofer Institute for Integrated Circuits IIS

Christoph Barz, Henning Rogge, Marc Adrat
Fraunhofer Institute for Communication, Information Processing and Ergonomics FKIE

Fraunhofer

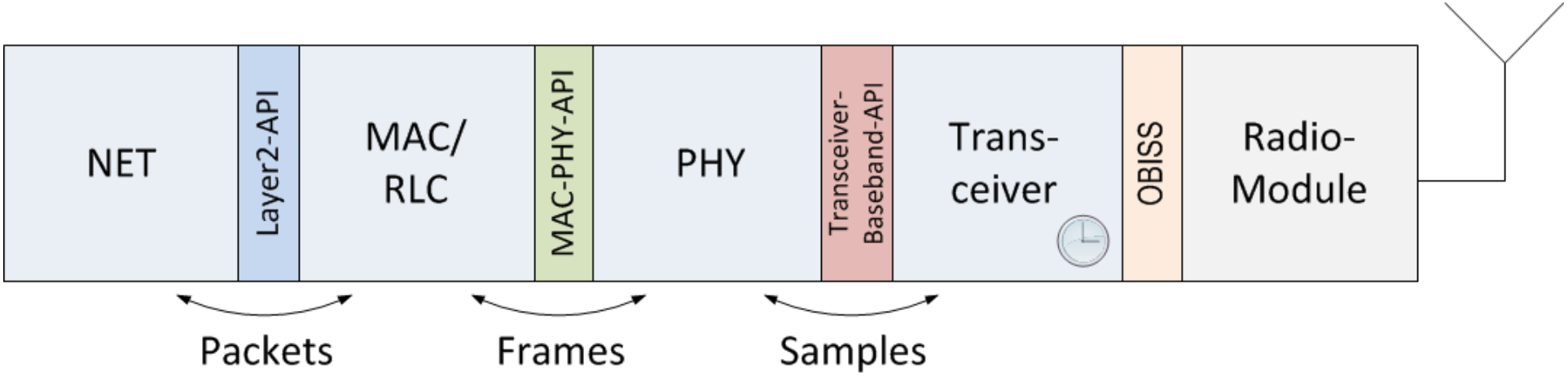
MOTIVATION

- Carefully designed interfaces between the key functional areas (ISO/OSI layers) of a waveform can significantly improve, e.g.,
  - waveform development, implementation, test, and portability
- In a *Software Defined Radio* (SDR) these interfaces are provided by
  - *Application Programming Interfaces* (APIs)
- The *Wireless Innovation Forum* (WInnF) facilitates progress on APIs
  - by promoting standardization and harmonization
  - in an international *Standards Developing Organization* (SDO)
  - by recognizing *Efficient Porting of Waveform Applications* as #1 of the *Top 10 Most Wanted Innovations List*

Motivation for APIs

„Classical radio“

- Stack of Software Defined Radio layers
  - “Analogue” radio module
  - “Digital” baseband processing unit
    - Waveform layers: PHY, MAC & Link, Network

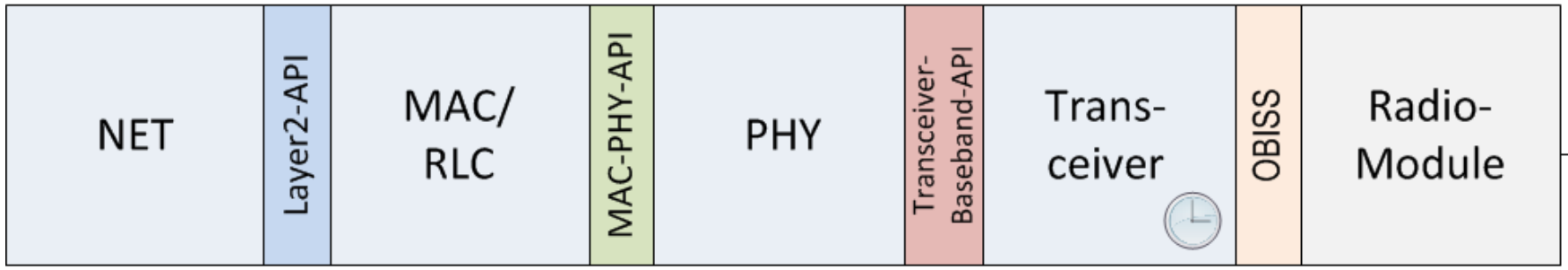


Overview – Current Software Defined Radio APIs

IETF Dynamic Link Exchange Protocol (DLEP), Internet Draft (draft-ietf-manet-dlep-24)

SDRF-08-I-0014-V0.0.0 Specification of the IQ Baseband Interface, Version 1.0, 26. January 2009

WINNF-15-I-0094 OBISS Interface Specification, Version 2.0, 25. November 2015



???

Specification needs still to be developed

SDRF-08-S-0008-V1.0.0 Transceiver Facility Specification, Version 1.0, 28. January 2009

WINNF-08-S-0008-V1.3.30 Transceiver Facility Specification, Draft Version, 28. September 2016 (work in progress; Version 2.0 to be released soon)

Fraunhofer

Flexible IP-Waveform (FLIP-WF) – A National R&T Study

Fraunhofer Institutes IIS & FKIE in Co-operation with BAAINBw & WTD81

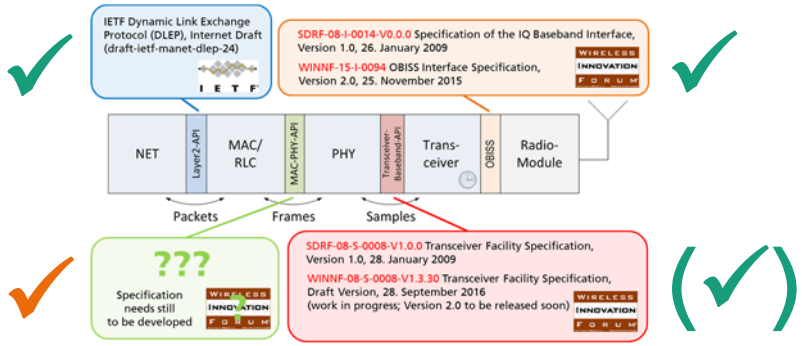
Bundeswehr
Wir. Dienen. Deutschland.
BAAINBw
WTD81

- Key Objective
  - Qualification for Fast Prototyping with flexible IP-Waveforms on Software Defined Radio (SDR) platforms

- Motivation / Rationale
  - research on a novel Flexible IP-Waveform (FLIP-WF) for future Tactical Environments
  - early Identification of new Technologies enabling a cost and time efficient Assessment in Practice
  - provision of new Operational and Technical Capabilities to the Warfighter (in Coalition Operations)
  - modular, scalable, and reconfigurable Design with modern *Cross Layer Interfaces*

Short Introduction to our National R&T Study – Rapid Prototyping with a Flexible IP-capable Waveform

- PHY
  - OFDM, 25-100 kHz BW
  - Tactical VHF / UHF bands
- MAC
  - CSMA with reserved slots for CNR-PTT-style voice transmission
- Link layer
  - Transparent Bridging
  - IPv4 & v6 capable (layer 3)
- (optional) MANET
  - Dynamic Routing with OLSRv2
  - Dynamic Link Exchange Protocol



Fraunhofer

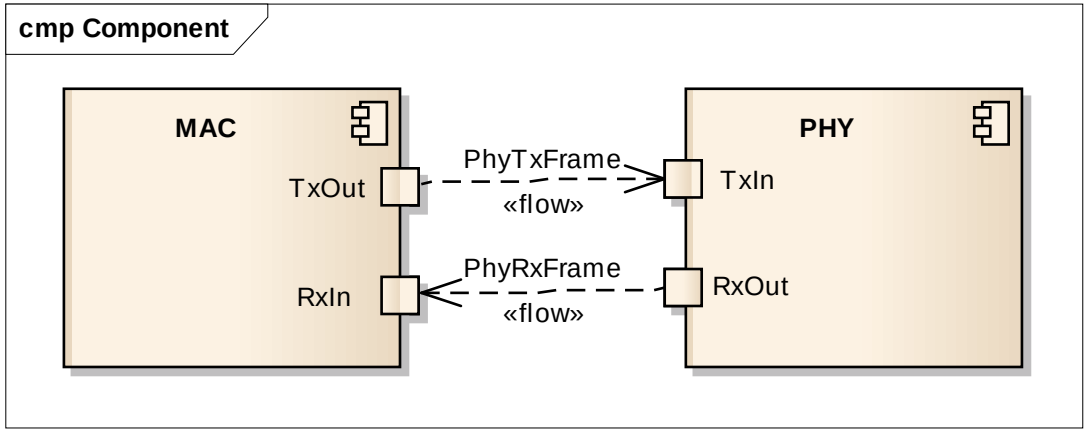
Benefits of a PHY-MAC-API

- For the national R&T Study:
  - the spatial separation of the development teams for the
    - PHY Layer (at FhG IIS in Erlangen)
    - MAC/NET Layers (at FhG FKIE in Wachtberg)

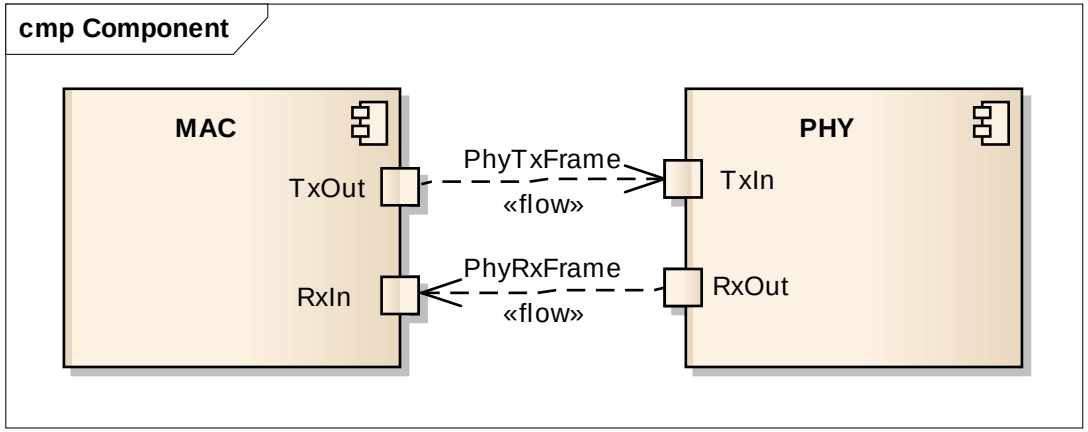
    required a clear and precisely defined PHY-MAC-API
- In general:
  - typically the implementation boundary between GPP and DSP/FPGA on a heterogeneous SDR platform
  - separates sample driven PHY from timing driven bursts on MAC
  - API hides all implementation details from the complementing layer
  - … and lots more …

API Architectural Details – Data Model Transmit

- Transmit indications
  - SEND_FRAME
  - GET_TIME_INFO
    - Response: TIME_INFO
  - GET_PHY_PARAMETER
    - Response: PHY_PARAMETER
  - SEND_FRAME_LOOP (for testing purposes)
  - SET_TIMER

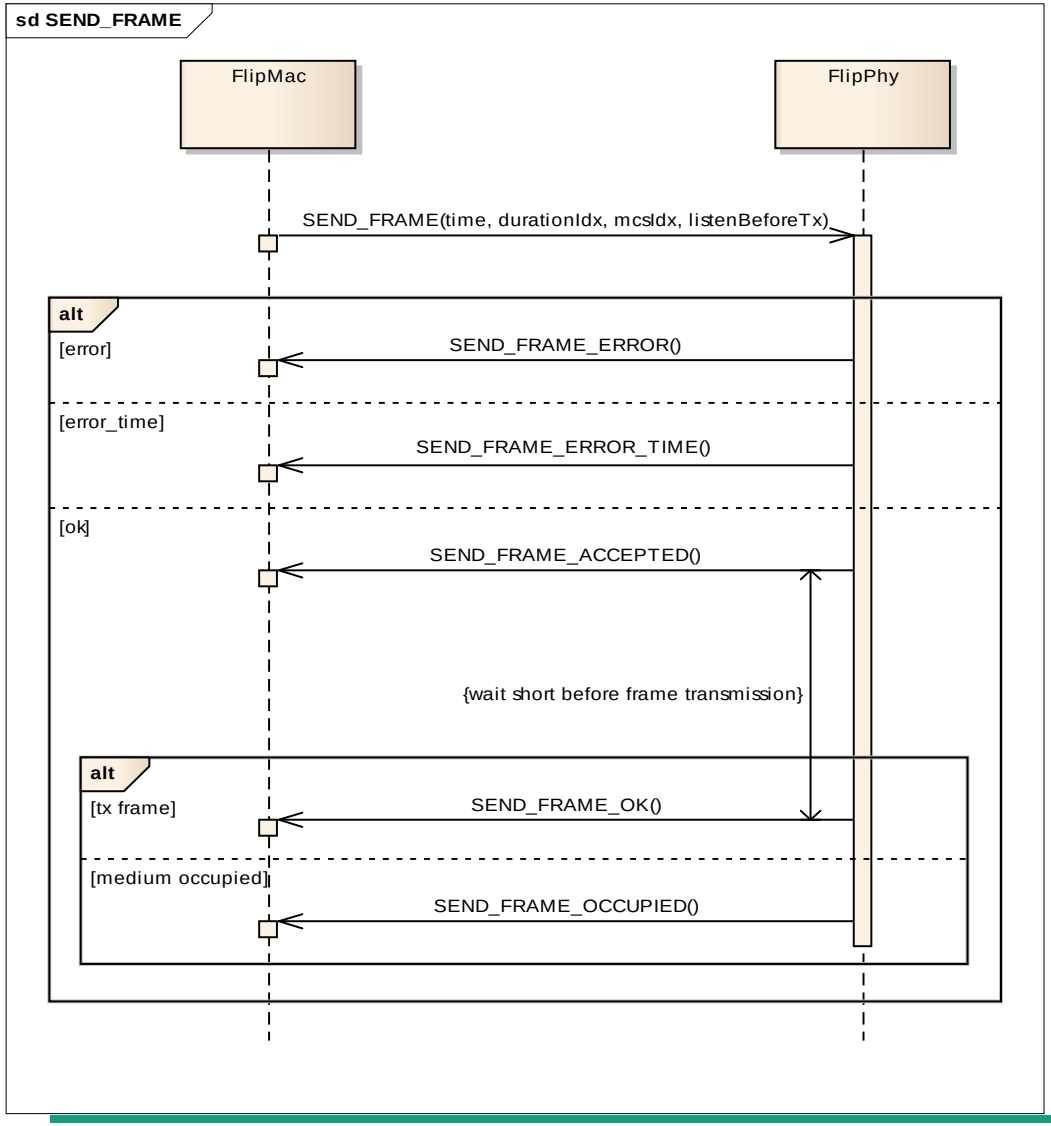


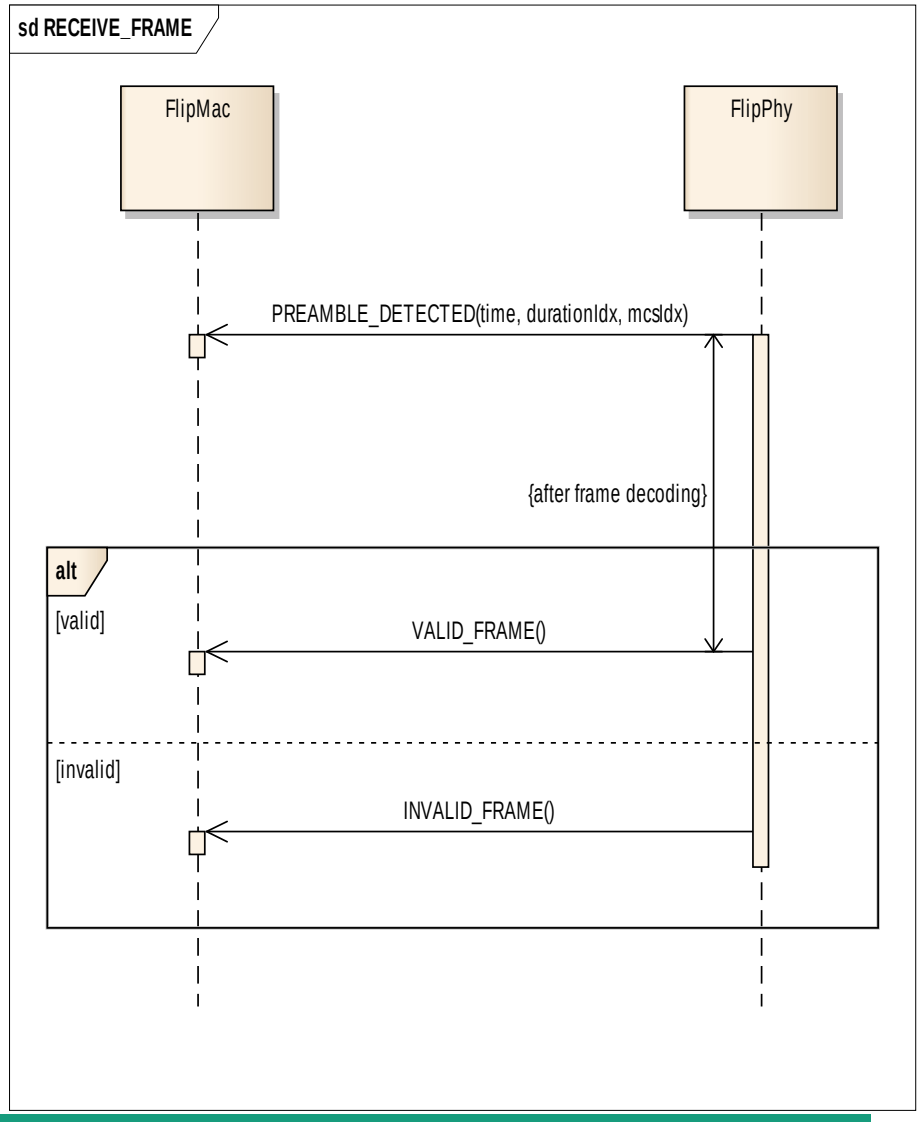
API Architectural Details – Data Model Receive



- Receive indications
  - PREAMBLE_DETECTED
  - VALID_FRAME
  - INVALID_FRAME
  - SEND_FRAME_ERROR_TIME
  - SEND_FRAME_ERROR
  - SEND_FRAME_ACCEPTED
  - SEND_FRAME_OK
  - SEND_FRAME_OCCUPIED
  - and others: TIMER_EVENT, TIME_INFO, PHY_PARAMETER

API Architectural Details – Timing Sequences

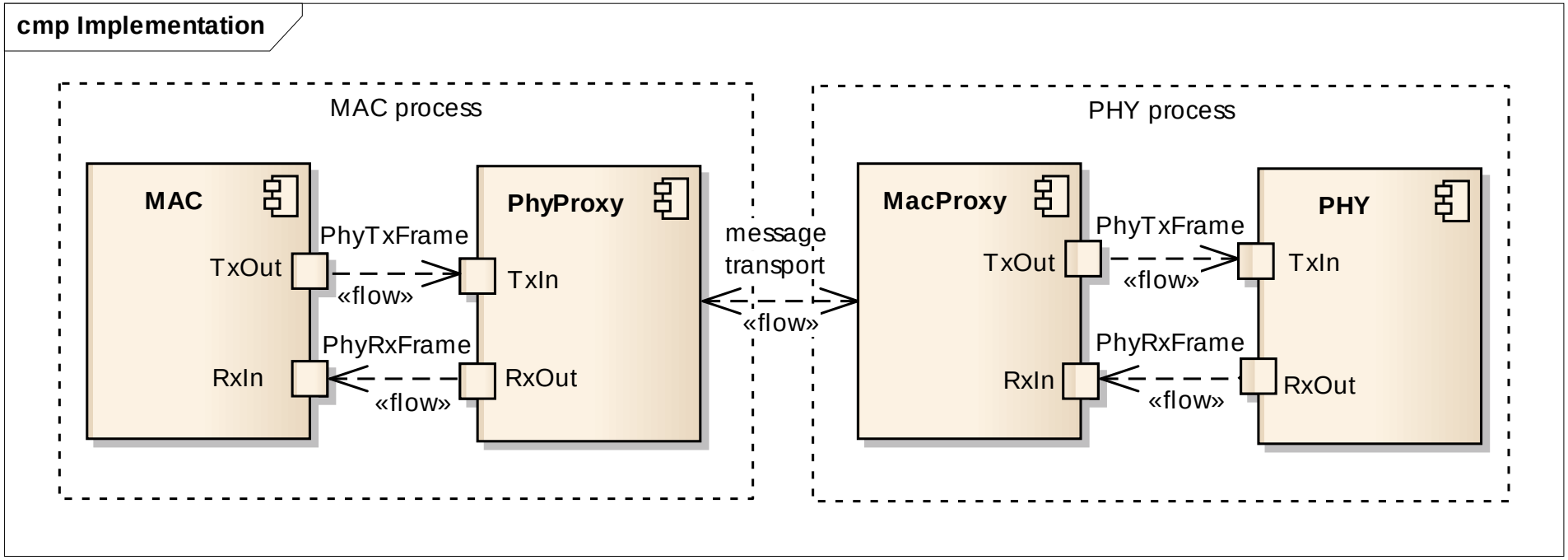




Fraunhofer

API Implementation Details

- Current implementation in C and Python
- Inter Process Communication and message transport via UDP sockets
- Object serialization with Google protocol buffers v2



Fraunhofer

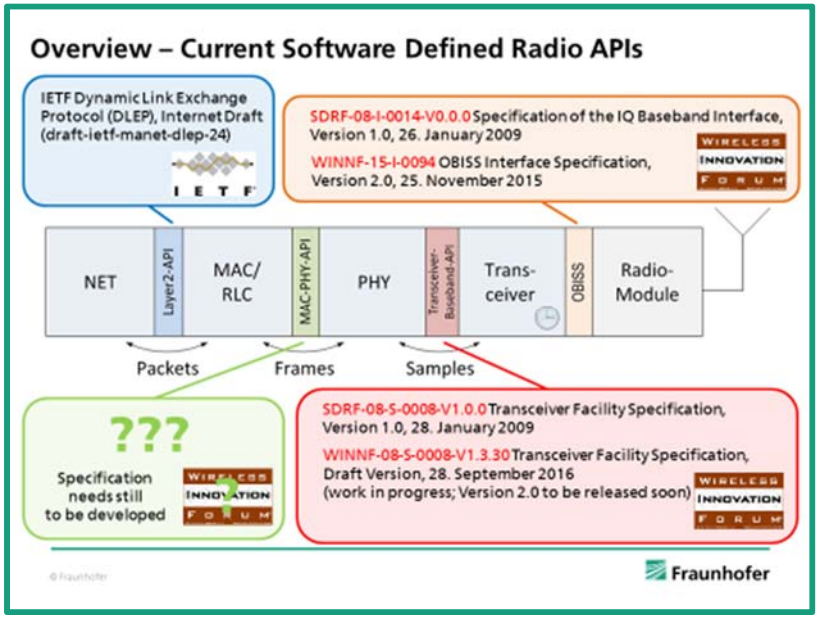
Demonstration at NATO CWIX 2016

- Planned participation at NATO CWIX 2017 with new waveform feature set

Fraunhofer

Way Ahead

- Our MAC-PHY-API definition will be updated and revised iteratively within our current national R&T study (FLIP-WF)
- With our iterative implementation and demonstration approach, we are able to give the API some time to “reach maturity”
- Most important:
  - we implement and test on real hardware
  - not just academic studies
- Further results can be provided again at the *Wireless Innovation Forum*



Fraunhofer