OpenAirCN-5G Project
Current Activities

Olivier CHOISY, b<>com
Michel TREFCON, b<>com
Luhan WANG, BUPT
Tien-Thinh NGUYEN, EURECOM
Outline

- **Introduction to OAI CN 5G project**
- **Introduction to 5G Core Network**
- **Current activities**
  - Meetings, discussion between partners
  - Code generation for interfaces
  - A prototype Implementation
- **Roadmap**
Introduction

- Open Air Interface Software Alliance
  - Core Network Software
    - Rel10/Rel14 Implementation
    - With ongoing development/maintenance

- Initiative to start a 5G Core Implementation
  - Objectives:
    - Initiative to provide an Open source implementation of 5G Core Network as specified by 3GPP
    - Build a community to perform this implementation

- How?
  - New Software implementation (5G Skeleton then NFs)
  - Partners: Eurecom, b<>com, BUPT, Blackent, ng5T
Introduction

- Preliminary work (Eurecom/BUPT) Q3-2017
  - HTTP/2, 5G SBA bus, ... ➔ initial prototype
- Initial workshop to start officially: 04/03/2018 (b<>com Paris)
  - Decision to start with a complete skeleton based on SBI/SBA 3GPP specifications (TS23.5xx, 29.5xx)
- 4 synchronization meetings with partners
- Trello project follow-up
- Progressive implication from partners expected
- Q3/Q4: initial SBA Interfaces implementation – on going
Outline

- Introduction to OAI CN 5G project
- **Introduction to 5G Core Network**
- Current activities
  - Meetings, discussion between partners
  - Code generation for interfaces
  - Prototype Implementation
- Roadmap
5G – Service Trends and Usage Scenarios

Enhanced Mobile Broadband
- Gibabytes in a second
- 3D video, UHD screens
- Work and play in the cloud
- Augmented reality
- Industry automation
- Mission critical appicaiton, e.g. e-health
- Self Driving Car

Smart Home/Building

Voice

Smart City

Future IMT

Massive Machine Type Communications

Ultra-reliable and Low Latency Communications

Gigabytes in a second

Network Slicing

Self Driving Car

Industry automation

eHealth

Work and play in the cloud

Augmented reality

3D video, UHD screens
Service-Oriented 5G Core Network

- Next Generation Network: To meet the needs of the range of services envisioned for 5G, with diverse performance requirements, across a wide variety of industries: *Flexible, Scalable, and Customizable*

- Service Based Architecture
  - support a modularized service, flexible and adaptable, with fast deployment cycles and updates for launching services on demand in the network
  - a set of NFs providing services to other authorized NFs to access their services
5G System Architecture

- Access and Mobility Management Function (AMF)
- Session Management Function (SMF)
- User plane function (UPF)
- Policy Control Function (PCF)
- Authentication Server Function (AUSF)
- Unified Data Management (UDM)
- Network Exposure function (NEF)
- NF Repository function (NRF)
- Network Slice Selection Function (NSSF)
- Service based interfaces (SBI): Namf, Nsmf, Nudm, Nnrf, Nnssf, Nausf, Nnrf, Nsmsf, Nudr, Npcf

5G System Architecture (3GPP TS23.501)
5G System Architecture

- 5G System Architecture in reference point representation (non-roaming scenario) (3GPP TS23.501)
Core Network: From 4G to 5G Networks

MME: Mobility Management Entity
SGW: Serving Gateway
PGW: Packet Data Network (PDN) Gateway
PCRF: Policy and Charging Rules Function
HSS: Home Subscriber Server
Service Based Architecture

- Network Function (NF) Service Framework (3GPP TS 23.501)
  - Interactions between NF Service Consumer and NF Service Producer: Request-Response, Subscribe-Notify
  - NF service authorization
  - NF service registration, de-registration
  - NF service discovery

- Interactions between NF Service Consumer and NF Service Producer
Service Based Architecture – NF Service Framework

- HTTP methods - Request/Response communication (CRUD)
  - Create/Read/Update/Delete a resource
Service Based Architecture – NF Service Framework

- HTTP methods - Subscribe/Notify communication
  - Notifications will be sent to NF Service Consumer when the change/event occurs at the Service Provider (Provider ~ Client, Consumer ~ Server)
  - Create/Modify/Delete subscriptions using HTTP POST, PUT/PATCH, DELETE

Create a subscription using HTTP POST

1. POST /xyz_subscriptions (XyzSubscription)
2. 201 Created (XyzSubscription)

Send a notification using HTTP POST

1. POST {callback_ref} (Notification)
2. 204 No Content

Modify a subscription using HTTP PUT

1. PUT .../{subscriptionId} (XyzSubscription)
2. 204 No Content

Delete a subscription using HTTP DELETE

1. DELETE .../{subscriptionId}
2. 204 No Content
Service Based Architecture – NF Service Framework

- NF service registration
  - NF instance informs the NRF the list of NF services that it supports e.g., during NF instance registration/activation

- NF service de-registration
  - Deregistration with the NRF due to specific events e.g., NF instance shutdown/disconnect or errors
Service Based Architecture – NF Service Framework

- NF service discovery (relying on NRF)
  - To be discovered by NRF, the requested NF instance needs to be registered in the NRF

- NF service status subscribe/notify
  - Subscribe to be notified of new registered/updated/deregistered NF instances
5GC Components - AMF

- Four services (3GPP TS23.501; 3GPP TS23.502; 3GPP TS29.518)
  - Namf_Communication service - Enables an NF consumer to communicate with the UE and/or the AN through the AMF
  - Namf_EventExposure service - Enables other NF consumers to subscribe or get notified of the mobility related events and statistics
  - Namf_MT service - Enables an NF consumer to make sure UE is reachable
  - Namf_Location service - Enables an NF consumer to request location information for a target UE

- NF Consumers (Table 5.2.2.1-1,3GPP TS23.502)

- Namf_Communication Service (3GPP TS29.518)
  - **Function**: Enables an NF to communicate with the UE through N1 NAS messages or with the AN (both UE and non UE specific)
  - **Operations**: UEContextTransfer; CreateUEContext; ReleaseUEContext; RegistrationCompleteNotify; N1MessageNotify; N1MessageSubscribe; N1MessageUnSubscribe; N1N2MessageTransfer; N1N2TransferFailureNotification; N2InfoSubscribe; N2InfoUnSubscribe; N2InfoNotify; EBIAssignment; AMFStatusChangeSubscribe; AMFStatusChangeUnSubscribe; AMFStatusChangeNotify
  - **Operation Parameters** (in & out) (clause 5.2.2.2, 3GPP TS23.502)
  - **Resource URLs and HTTP Method** (Table 6.1.3.1-1, 3GPP TS29.518)
  - **Usage of each operation** (clause 5.2.2, 3GPP TS29.518)

- Other AMF services’ detailed information also can be found in 3GPP TS29.518
5GC Components - AMF

- Different from EPC in interface protocol

  ➢ MME corresponds to AMF and SMF
  ➢ HSS corresponds UDM and AUSF
  ➢ Interface s6a turns into SBI

  ▪ Protocol stack changes from Diameter to HTTP/2
5GC Components – SMF, UDM, AUSF, PCF, NRF

- **SMF**
  - TS29.502 - Session Management Services, Stage 3, Rel 15.1.0, 2018-09
  - TS29.508 - Session Management Event Exposure Service, Stage 3, Rel 15.1.0, 2018-09

- **UDM**
  - TS29.503 - Unified Data Management Services, Stage 3, Rel 15.1.0, 2018-09

- **AUSF**
  - TS29.509 - Authentication Server Services, Stage 3, Rel 15.1.0, 2018-09

- **PCF**
  - TS29.507 - Access and Mobility Policy Control Service, Stage 3, Rel 15.1.0, 2018-09
  - TS29.512 - Session Management Policy Control Service, Stage 3, Rel 15.1.0, 2018-09

- **NRF**
  - TS29.510 - Network Function Repository Services, Stage 3, Rel 15.1.0, 2018-09
Outline

- Introduction to OAI CN 5G project
- Introduction to 5G Core Network
- **Current activities**
  - Meetings, discussion between partners
  - Code generation for interfaces
  - A prototype Implementation
- Roadmap
Current Activities

- Initiate some Project Coordination (trello) and regular meetings
  - Project Structure
  - Different Languages target: Cplusplus, Java

- Needs Partners to be involved in Design and implementation

- b<>com activities
  (shared on git: https://github.com/OPENAIRINTERFACE/openair5g-cn)
  - 3GPP Specification follow up
  - Identification of APIs and data models per NF
  - NFs Interface APIs and Code generation for interfaces
Outline

- Introduction to OAI CN 5G project
- Introduction to 5G Core Network

Current activities
  - Meetings, discussion between partners
  - Code generation for interfaces
  - A prototype Implementation

Roadmap
SBA Skeleton objective Activities

- Automatic source code generation
  - Evaluate tools,
  - define a process based from 3GPP Yaml
- 5GC API AMF and NRF focus
- Software environment
- Feedbacks
- Next steps
openapi-generator
automatic source code generation tool

OpenAPITools/openapi-generator open source project

- can be used
  - from CLI
  - as a maven or gradle plugin
- can generate source code for server and client sides
- 1 or 2 releases per month
- Template files can be customized
- Extensible: open for new generators
- Current used version is 3.3.3 started with version 3.1.0 (June 2018)
3GPP 5GC API

- Yaml files
  extracted from TS files and put under SCM

- AMF

  MT, Location, EventExposure and Communication
  - started with R15.0.0 and upgraded to R15.1.0
  - REST facades available with restrictions

- NRF

  NFManagement, NFDiscovery and OAuth2
  - NFManagement REST facade available with restrictions
  - NFDiscovery ongoing
  - OAuth2
Software Environment

- OpenAPITools/openapi-generator
  Current used version is 3.3.3 (started with 3.1.0)

- Java environment
  - java8
  - jaxrs/resteasy (server side)
  - maven 3.5.x
  - thorntail 2.2.0 (started with 2018.5.0)
  - uberjar file to be run on CLI
  - HTTP/2 enabled

- C++
  - Pistache framework (server side)
  - cpprest-sdk LIB (client side)
  - C++11, GNU g++ 5.4.0, cmake, make

- misc
  - swagger-ui : visualize and interact with the API’s resources (Web App)
Results

REST facade source code fully generated from yaml files.
- 1st implementations in Java & C++ serve responses with random values based on their types and value constraints (range constraints, regex)
- serving error responses (ProblemDetails) on not valid content receipt (malformed json body, invalid type values, constraint violation...)

Work to do is to add business logic on top of the generated skeleton.

Restrictions

Need to update 3GPP yaml files because of openapi generator restrictions
- $ref path must start with ./
- not fully OpenApi 3.0 compliant (anyOf, allOf, OneOf & callbacks)
- rewrite definitions to get around generator issues
Pros
- active community with many contributors and regular deliveries (monthly basis).
- lot of source code generators (Java, C++, Python, Go, ... etc)
- good documentation
- customizable
- extensible

Cons
- Currently not fully compliant against OpenAPI 3.0
  Doesn’t yet support anyOf/oneOf/allOf (work in progress in the open source community) & callbacks
- Issues may occur (from most serious to least serious)
  - at source code generation stage (exception)
  - at compilation time (generated source code doesn’t compile)
  - at runtime (expected feature may not work - e.g. bean validation for java/jaxrs/resteasy)
- OpenAPI compliance and issues can depend on generator family
3GPP TS Release “Migration”

3GPP specification changes impact the current source code base.

- From R15.0.0 to R15.1.0 (Sept release)
  - compilation failures
    - new interfaces to implement
    - removed attributes
  - test issues
    - value constraints updated
Next Steps

- **Java**
  - HTTP2 client
  - unit and integration tests

- **C++**
  - HTTP2
  - unit and integration tests

- **Software environment**
  - update of software components (OpenAPITools generator, thorntail, C++ libraries etc) benefits are bug fixes and new features.
  - need no regression tests with good coverage rate
Outline

- Introduction to OAI CN 5G project
- Introduction to 5G Core Network
- Current activities
  - Meetings, discussion between partners
  - Code generation for interfaces
  - A prototype Implementation
- Roadmap
A prototype implementation

- Related works
  - stage one - SBI implementation based on nghttp2.0 library in OAI
  - stage two - A simple 5G Core demo with python flask framework

- Capsulate Nghttp2.0 library to implement REST API
  - What’s nghttp2.0 library? - C implementation of HTTP/2 protocol
  - How to implement REST API?
    - Parse HTTP METHOD from “:method” header frame in function “on_header_callback”
    - Parse Resource URI from “:path” header frame in function “on_header_callback”
    - Parse data payload from from data frame in function “data_recv_callback”
    - REST API example - DELETE https://127.0.0.1:5001/namf-comm/v1/ue-contexts/1234
  - Work in OAI? - Integrate nghttp2.0 into OAI, separate MME into ESM and EMM
  - Evaluated the SBI performance(2018 ICC)

Title - “Performance Evaluation of Candidate Protocol Stack for Service-based Interfaces in 5G Core Network”
A Prototype Implementation
A simple 5GC demo developed with python flask framework

- **What’s the purpose?**
  - Develop a 5G Core framework, which shows how service-based 5G Core works
  - Implement simple NFs, which shows how service-based NFs work and what difference with EPC NFs
  - Simulate UE Registration Management Procedure and UE Triggered Service Request Procedure, which shows how 5G System Procedures work
  - This work shows how to follow 3GPP specifications to develop 5G NFs

- **Disadvantages?**
  - This demo just implements part of 5GC
    - ✓ parts of 5GC NFs - AMF, SMF, UDM, AUSF and UPF
    - ✓ A few system procedures - UE Registration Management Procedure and UE Triggered Service Request Procedure
  - Some don’t follow 3GPP specification
    - × N2, N3 and N4 protocol
    - × Simplified message parameters
    - × HTTP version is v1.1 - because python flask doesn’t support HTTP/2

- **Overview of the prototype**
  - What python flask framework look like? - one example from source code
  - Two procedures implementation - Registration & Service Request
  - Test scenario and explanation
  - Video
A Prototype Implementation Framework (python-flask)

➢ __init__.py - setting of listening port and default url prefix
➢ api folder - all service endpoints
➢ routes.py - Mapping of Resource

def create_app():
    app = Flask(__name__, static_folder='static')
    app.register_blueprint(
        v1.bp,
        url_prefix='/namf-comm/v1'
    )
    return app

def post(self, ueContextID):
    args = parser.parse_args()
    t = Thread(target = AMFNotifyToAN, args=(args,ueContextID,))
    t.start()
    return None, 200

class N1N2MSG(Resource):
    def __init__(self):
        super().__init__()

    def post(self, ueContextID):
        args = parser.parse_args()
        t = Thread(target = AMFNotifyToAN, args=(args,ueContextID,))
        t.start()
        return None, 200
A Prototype Implementation
Registration Management Procedure

1. Registration Request
2. AMF selection
3. Registration Request
4. Authentication/Security
5. Nudm_UECM_Registration
5b. Nudm_SDM_Get
5c. Nudm_SDM_Subscribe
6. Registration Accept
A Prototype Implementation
UE Triggered Service Request Procedure

1. Service Request
2. N2 message (Service Request)
3. Nsmf_PDUSession_UpdateSMContext Request
4a. N4 Session Establishment Request
4b. N4 Session Establishment Response
5. Nsmf_PDUSession_UpdateSMContext Response
6. Uplink data
A Prototype Implementation
Test Scenario

- **Implementation of NFs**
  - AMF - Connection & Registration management
  - SMF - PDU Session management
  - UDM - database management (use mysql database in HSS,EPC)
  - AUSF - Authentication center
  - UPF - configure UP
  - VM - simulated UE+AN

- **Deployment**
  - Five NFs are deployed in physical host
  - UE+AN are deployed in a VM

- **Message Path**
  - signaling message path *(red line)*
  - data flow *(green line)* (not be established initially)

- **Test**
  - After VM(UE+AN) finishes CM,RM and PDU Session management procedures
  - Then UPF *configure UP* for date flow (green line being established)
  - Finally, VM can *communicates with host* via green line data path
A prototype implementation

 Demo
Outline

- Introduction to OAI CN 5G project
- Introduction to 5G Core Network
- Current activities
  - Meetings, discussion between partners
  - Code generation for interfaces
  - A prototype Implementation

- Roadmap
ROADMAP

- Benefit from python based « Prototype implementation »
- Additional Partners to speed up implementation
- Structure Project
  - Git
  - Continuous Integration
  - Tests Tools
ROADMAP

- Core NFs interface available (Java, Cplusplus):
  - AMF (Java, cplusplus), NRF(java) : Q4 2018
  - SMF, AUSF (?) : Q1 2019
- Implementation of main functions of NRF : Q1 2019
- Implementation of a use case (simplified) as UE attach : Q2 2019
  - NRF registry from AMF
  - AMF attach with proxy to 4G HSS, SMF (?)

HELP WELCOME !
Current Status of 3GPP Specifications

5G System

- **TS23.501** - System Architecture for the 5G System, Stage 2, Rel 15.3.0, 2018-09
- **TS23.502** - Procedures for 5G System, Stage 2, Rel 15.3.0, 2018-09
- **TS29.500** - 5G System, Technical Realization of Service Based Architecture, Stage 3, Rel 15.1.0, 2018-09
- **TS29.501** - 5G System, Principles and Guidelines for Services Definition, Stage 3, Rel 15.1.0, 2018-09

5GC components

- **AMF**: TS29.518 - Access and Mobility Management Services, Stage 3, Rel 15.1.0, 2018-09: AMF Communication Service, AMF Event Exposure Service, AMF Location Service, AMF Mobile Termination Service
- **NRF**: TS29.510 - Network Function Repository Services, Stage 3, Rel 15.1.0, 2018-09: NRF NF Discovery Service, NRF NF Management Service
Current Status of 3GPP Specifications

- **SMF**
  - TS29.502 - Session Management Services, Stage 3, Rel 15.1.0, 2018-09: SMF
    - PDU Session Service
      - TS29.508 - Session Management Event Exposure Service, Stage 3, Rel 15.1.0, 2018-09
  - AUSF: TS29.509 - Authentication Server Services, Stage 3, Rel 15.1.0, 2018-09
  - PCF
    - TS29.507 - Access and Mobility Policy Control Service, Stage 3, Rel15.1.0, 2018-09
    - TS29.512 - Session Management Policy Control Service, Stage 3, Rel 15.1.0, 2018-09
    - TS29.571 - Common Data Types for Service Based Interfaces, Stage 3, Rel 15.1.0, 2018-09
Detailed NF information in specifications (SMF)

- Two services (3GPP TS23.501, 3GPP TS23.502)
  - **Nsmf_PDUSession** service - manages the PDU Sessions; uses the policy and charging rules received from the PCF
  - **Nsmf_EventExposure** service - exposes the events happening on the PDU Sessions to the consumer NFs

- NF Consumers (Table 5.2.8.1-1, 3GPP TS23.502)
- **Nsmf_PDUSession Service** (3GPP TS29.502)
  - **Function?** - Operates on the PDU Sessions (Creation / Deletion / Modification of AMF-SMF interactions for PDU Sessions)
  - **Operations**: Create; Update; Release; CreateSMContext; UpdateSMContext; ReleaseSMContext; SMContextStatusNotify; StatusNotify; Context
  - **Operation Parameters** (in & out) (clause 5.2.8.2, 3GPP TS23.502)
  - **Resource URLs and HTTP Method** (Table 6.1.3.1-1, 3GPP TS29.502)
  - **Usage of each operation** (clause 5.2.2, 3GPP TS29.502)

- **Nsmf_EventExposure service’s detailed information also can be found in 3GPP TS29.502**
Detailed NF information in specifications (SMF)

Different from EPC in interface protocol

- MME corresponds to AMF and SMF
- UPF can be understood as part of the SGW data plane
- Interface s11 turns into N4
- UDP Port number changes from 2123 to 8805 (detail see 3GPP TS29.244)

Protocol stack changes from GTP-C to GTP-U (detail see 3GPP TS29.244)
Detailed NF information in specifications (UDM)

Five services (3GPP TS23.501, 3GPP TS23.502, 3GPP TS29.503):

- **Nudm_SubscriberDataManagement** service - Allow NF consumer to retrieve user subscription data when necessary; Provide updated user subscriber data to the subscribed NF consumer
- **Nudm_UEContextManagement** service - Provide information related to UE’s transaction information; Allow NF to register and deregister its information for the serving UE in the UDM; Allow NF to update UE Context information in the UDM
- **Nudm_UEAuthentication** service - Provide updated authentication related subscriber data
- **Nudm_EventExposure** service - Allow NF consumer to subscribe to receive an event
- **Nudm_ParameterProvision** service - Provision information which can be used for the UE in 5GS

- NF Consumers (Table 5.2.3-1, 3GPP TS23.502)
- Nudm_SubscribeDataManagement Service (3GPP TS29.503)
  - Function?
  - Operations: Get; Subscribe; Unsubscribe; Notification; Info
  - Operation Parameters (in & out) (clause 5.2.3.3, 3GPP TS23.502)
  - Resource URLs and HTTP Method (Table 6.1.3.1-1, 3GPP TS29.503)
  - Usage of each operation (clause 5.2.2, 3GPP TS29.503)

- Other UDM services’ detailed information also can be found in 3GPP TS29.503
Detailed NF information in specifications (AUSF)

Two services (3GPP TS29.509)

- **Nausf_UEAuthentication** service - Provide **UE authentication** service to the requester NF
- **Nausf_SoRProtection** service - Provide SoRProtection service protecting the **Steering Information List** from being tampered with or removed by the VPLMN

- **Nausf_UEAuthentication Service** (3GPP TS29.509)
  - **Function***- Provide **UE authentication service** to the requester NF
  - **Operations** : Authenticate
  - **Resource URLs and HTTP Method** (Table 6.1.3.1-1, 3GPP TS29.509)
  - **Usage of each operation** (clause 5.2, 3GPP TS29.509)
  - Depending on the information provided by the AMF, AUSF enters in one of the following procedures:
    - ✔️ 5G-AKA
    - ✔️ EAP-based authentication

- **Nausf_SoRProtection service**'s detailed information also can be found in 3GPP TS29.509
Detailed NF information in specifications (UPF)

Features of the UPF (3GPP TS23.501)

- Traffic detection (e.g., classifying traffic of IP type, Ethernet type, or unstructured type)
- Traffic reporting (e.g., allowing SMF support for charging)
- QoS enforcement
- Traffic routing (e.g., for UL CL or IPv6 multi-homing)

SMF and UPF interactions (3GPP TS23.501)

- N4 Node level Procedures
- N4 Session management Procedures
- Parameters for N4 session management
  - N4 Session ID
  - Packet Detection Rules (PDR) that contain information to classify traffic (PDU(s)) arriving at the UPF
  - Forwarding Action Rules (FAR) that contain information on whether forwarding, dropping or buffering is to be applied to a traffic identified by PDR(s)
  - Usage Reporting Rules (URR) that contains information that defines how traffic identified by PDR(s) shall be accounted as well as how a certain measurement shall be reported
  - QoS Enforcement Rules (QER) that contain information related to QoS enforcement of traffic identified by PDR(s)