DT Access 4.0 Architecture Description

Version 5.0

DT and Radisys Confidential

DT Access 4.0 Architecture Description
Version 5.0

Contents

Figures

Release History

This table lists the history of changes in successive revisions to this document:

Table 1: Release History

Version	Date	Author (s)	Status	Description
0.1	Nov 01, 2021	Shaun Missett	Draft	ToC Outline
0.2	Nov 29, 2021	Shaun Missett	Draft	Working on Contents
V2.0	Dec 14, 2021	Shaun Missett	RYSS Initial Version	Initial Version released to DT
V2.1	Dec 15, 2021	Shaun Missett	RYSS Initial Version	Updated the E2E Section of FTTB. Modified some of the Figures to render properly in the “Reading” View. Small additions to various NFR sections, adding ‘standard statements’ Added descriptive comments to some sections with only Figures.Updated Release history to include Release to DT.
V2.2	Dec 17, 2021	Shaun Missett	RYSS Initial Version	Updated the Diagnosis section to include the new implementation by Radisys using POST versus GET in NEMO.
V2.3	Jan 14, 2021	Shaun Missett Srikanth Yandapalli Amit Ghosh	Second version Draft with input from DT on Version 1	Reformat to add high level architecture for A4 FTTH/FTTB and ECP/POD Networking. Reduce the detailed design information.
V2.3.3	Jan 31, 2022	Uwe Schlichting	RYSS Internal Version	Add Fabric Section
V2.7	June 31, 2022	Shaun Missett	RYSS Internal Version	Add Section on changes to Concept A4@DPU Add Message sequence extension for M-SEQ-003 DPU specific
V2.9	Pre-MS3, 2023	Shaun Missett, Uwe Schlichting	RYSS Internal Version	Update section 11. Update with response to T. Haag comments on Fabric section of 2.7. Update with changes responding to comments from Radisys Team. Remove Message sequence extension for M-SEQ-003 DPU specific. Link to Telekom-Wiki.
V3.0	Pre-MS3, 2023	Uwe Schlichting	RYSS Internal Version	Update section 11. With TSA Performance and SLA Target Table.
V3.1	MS3, 2023	Shaun Missett	Submitted to DT	Fix Figure Numbering Add Section 13 for Remote Device
V3.2	Pre MS4, 2023	Uwe Schlichting	RYSS Internal Version	Details on RD
V4.1	Pre MS5, 2024	Uwe Schlichting	RYSS Internal Version	Cleaning and preparing
V4.2	Pre MS5, 2024	Shaun Missett Uwe Schlichting Radu Cosnita Ashok Rawat Hans-Jörg Kolbe Robert Soukup Avinash Sakalabhaktula	RYSS Internal Version	Updated Links from Telekom-Wiki to TELEKOM-WIKI Updated figures (better look and feel) Layout and quality New Chapter 2 3.4.6.2 Service Edge Data plane 3.4.8 Fabric Routing 3.4.8 Service-Edge QoS 3.4.9. POD Management Networking

				3.5 FTTC added 4.2 Updated 4.2.3.11 EVPL Synchronization 4.4.1 RD Service Activation 4.4.2 A4 Management Networking 5 A4 POD Diagnosis Framework A4 POD Telemetry Framework 7 A4 Edge Cloud Platform
V4.3	Pre MS5, 2024	Keisha review		Final layout and cleaning
V4.4	Pre MS5, 2024	Final team review and additions		Final contributions
V5.0	MS5 Q2-2024	Uwe Schlichting	Submitted to DT	For feedback from DT until MS6 submission

Planned for next version with MS6 endo24:

- *DT review and feedback*
- *General decision how deep or low level we want to go – Uwe/Shaun*
  - *Seek feedback from DT*
- *Include an internal architecture diagram for local as well as central EMS - HJK*
- *Can we have a viewgraph on "inside the LEMS and cEMS)? – HJK*
- *Identify a chapter # for LCM (NOS/BaseOS/K8S, Infra, Apps) – Uwe*
- *Flows in 3.9: HJHK can describe in detail, including the mapping to BBF. - HJK*
  - *Will not be done for the moment, we do it once BBF TR474 is published*
- *Uwe: Are we missing a dedicated overview chapter for NEMS and CeML?*
- *Shaun/Uwe: Clocking and Timing update on 1588 and SyncE requirements and design*
- *Uwe: Chapter 3.4.6.2 Service Edge Data plane – New/Ready for review*
  - *Except Wholesale/Wholebuy N:1: Next version*
- *POD Networking is redundant*
- *Adjust Gross/Kleinschreibung*

1 Preface

1.1 Objective

This document provides a high-level architectural overview of the Access 4.0 Solution. It shall serve newcomers to get an overview as well as experienced project members to deep-dive into certain aspects. While this single document cannot cover details, one goal is to provide the reader the means to study selected items further. The Radisys team appreciates any feedback on this document that we can incorporate in further revisions.

1.2 Audience

This document assumes that the readers of this document are the A4 product development team, Management, Quality Assurance Department, Test and Marketing Teams.

1.3 References

Title	Date	Version	Publisher/Link
Concept DPU v5.1	Sept.20,2021	V5.1	D.T. DT Project A4 Concept Paper DPU@A4 V5.1.
Diagnosis Concept and Diagnosis Type Catalogue	Sept.30,2021	--	D.T.
BBF TR-101: Migration to Ethernet-Based Broadband Aggregation	2011/17	Issue 2	Broadband Forum: https://www.broadband-forum.org/technical/download/TR-101_Issue-2.pdf
BBF TR-178: Multi-service Broadband Network Architecture and Nodal Requirements	2017/09	Issue 2	Broadband Forum: https://www.broadband-forum.org/technical/download/TR-178_Issue-2.pdf
BBF TR-167: GPON-fed TR-101 Ethernet Access Node	2017/09	Issue 3	Broadband Forum https://www.broadband-forum.org/technical/download/TR-167_Issue-3.pdf
BBF TR-156: Using GPON Access in the Context of TR-101	2017/11	Issue 4	Broadband Forum: https://www.broadband-forum.org/technical/download/TR-156_Issue-4.pdf
BBF TR-459: Control and User Plane Separation for a Disaggregated BNG	2020/06	Issue 1	Broadband Forum: https://www.broadband-forum.org/technical/download/TR-459.pdf
BBF TR-301i2: Architecture and Requirements for Fiber to the Distribution Point	2017/03	Issue 2	Broadband Forum: https://www.broadband-forum.org/technical/download/TR-301_Issue-2.pdf

BBF TR-384: Cloud Central Office (CloudCo) Reference Architectural Framework	2018/01	Issue 1	Broadband Forum: https://www.broadband-forum.org/technical/download/TR-384.pdf
BBF TR-355: YANG Modules for FTTdp Management	2016/07	Issue 1	Broadband Forum: https://www.broadband-forum.org/technical/download/TR-355_Issue-1.pdf
BBF TR-385: ITU-T PON YANG Modules	2020/10	Issue 2	Broadband Forum: https://www.broadband-forum.org/technical/download/TR-385_Issue-2.pdf
BBF TR-459 Issue 2: Multi-Service Disaggregated BNG with CUPS. Reference Architecture, Deployment Models, interface, and Protocol Specifications	2023	Issue 2	https://www.broadband-forum.org/pdfs/tr-459-2-0-0.pdf
BBF WT-474: Subscriber Session Steering	Work in progress	Draft v14	Working text, will be published in 2025
Open BNG Operator Position Paper	2020	n/a	https://www.telekom.com/resource/blob/609420/d8e83cbf8032f6875b9231a869613dea/dl-200930-whitepaper-openbng-data.pdf
ONF SEBA Reference design 2.0	2021/03	Issue 2	ONF: https://opennetworking.org/wp-content/uploads/2021/03/ONF-Reference-Design-SEBAv2.0FINAL3.pdf
A4 – PAO and pfs “A ride through PAO-land and the interworking with PFS”	Nov 30, 2023	n/a	Online Training, available of Telekom-Wiki/Telekom Wiki H.J. Kolbe, D. Sahoo, R.N. Acharya
IEEE 1588v2	2008	Issue 2	IEEE
T-REC-G.988 _201908-Amd2	2019/08	Amd2	ITU-T
ITU-T G.8262.1: Timing characteristics of enhanced synchronous equipment slave clock	2019/01	1.0	ITU-T
ITU-T G.8275.1: Precision time protocol telecom profile for phase/time synchronization with full timing support from the network	2020/03	3.0	ITU-T
ITU-T G.8273.2: Timing	2020/10	4	ITU-T

characteristics of telecom boundary clocks and telecom time slave clocks for use with full timing support from the network			
Who dares wins! How access transformation can fast-track evolution of operator production platforms.	2019	n/a	https://www.adlittle.com/de-de/node/23381

1.4 Definitions, Acronyms, and Abbreviations

Term	Definition
ACL	Access List
AC/DC	Alternating Current/Direct Current
AFI/SAFI	(Subsequent) Address Family Indicator (BGP)
AN	Access Node
ANP-TAG	Access Node Port-Tag (tag added by access node like MSAN as second VLAN tag, therefore, could be called “Outer” tag, in standard documents often named S-TAG)
ASDF	Access Session Detection Function (c.f. BBF WT 474)
ASG	Aggregation and Service Gateway
AS	Autonomous System (BGP)
ASN	AS Number (BGP)
A4	Access 4.0
A10NSP	Handover point to Network Service Provider over A10 reference point according to BBF (wholesale model, BBF TR 101)
BBF	Broadband Forum
BBS	Business Support Systems
BGP	Border Gateway Protocol (eBGP: exterior BGP/iBGP: interior BGP)
BGP-LU	BGP Labeled Unicast
BNG	Broadband Network Gateway
BOM	Bill of Material
BOR	Bottom of Rack (Switch)
CAPEX	Capital Expenditures
CeML	Central Management Layer
CID	Circuit ID (HW port)
CMB	ConnectModular Broadband
CP	Control Plane
CPE	Customer Premises Equipment
CoA	Change of Authorization
COCO	Company Connect (DT)
CO8000	8000 DT Central Offices
DBNG	Disaggregated BNG (dBNG at DT: distributed BNG)
DCIP	DT L3 Business Access (Deutschland LAN IP-Connect)
DCN	DT Management Network (Distributed Control Network)
DHCP	Dynamic Host Control Protocol

DPU	Distribution Point Unit
DSL	Digital Subscriber Line (SDSL: Synchronous DSL/VDSL: Very High Speed DSL
DT	Deutsche Telekom
eEEC	enhanced synchronous Ethernet equipment clock
EMS	Element Management System
EPL	Ethernet Private Line
EVC	Ethernet Virtual Circuit
EVPL	Ethernet Virtual Private Line
FSoL	First Sign of Life
FTTC	Fiber to the curve
FTTH	Fiber to the Home
FTTB	Fiber to the Building
GEM	GPON Encapsulation Method
GPON	Gigabit Passive Optical Network
G.fast	ITU_T Standard G-Series for “fast access to subscriber terminals” (DSL based)
IGP	Internet Gateway Protocol
IP2	DT Backbone (IP/MPLS)
IP	Internet Protocol
IPSec	IP Security (Encryption)
IPTV	IP Television
IPv4	Internet Protocol version 4
IPv6	Internet Protocol version 6
ISIS	Intermediate System to Intermediate System (such as ISIS)
IT	Information Technology
LAC	L2TP access concentrator
LDP	Label Distribution Protocol (MPLS)
LI	Lawful Intercept
Line ID	Line ID (NGA-Forum), identifies the access line (per U-Interface) uses TLV 0x02 for agent remote ID; (Access-Loop-Remote-ID)
LER	Label Edge Router (also PE Router)
LNS	L2TP network server
LSR	Label Switch Router (also P-Router)
LoML	Local Management Layer
L2BSA	Layer 2 Whole-Sale
L3BSA	Layer 3 Whole-Sale
L2TP	Layer 2 Tunneling Protocal
MAC	Media Access Control

MED	Multi Exit Discriminator (BGP)
MSAN	Multi-Service Access Node
MPLS	Multiprotocol Label Switching
MTU	Maximum Transfer Unit
NE	Network Element
NEG	Network Element Group
NEL	Network Element Link
NEMS	Network Element Management System
NFV	Network Function Virtualization
NEP	Network Element Port
NSP	Network Service Profile
NTP	Network Time Protocol
NTR	Network Timing Reference
OLT	Optical Line Termination
OAM	Operation and Maintenance
ONF	Open Network Forum
ONT	Optical Network Termination
ONU	Optical Network Unit
OPEX	Operational Expenditures
OS&R	Operations Support and Readiness
OSS	Operations Support Systems
PFCP	Packet Forwarding Control Protocol
PADI	PPPoE Active Discovery Initiation
PADO	PPPoE Active Discovery Offer
PADT	PPPoE Active Discovery Terminate
PAO	POD Access Orchestrator
PFS	“Plattformsteuerung” (DT Radius Server Platform)
P-GW	Packet Data Network Gateway
POD	Point Of Delivery
POD_LM	PoD Local Management
POD_PAO	PoD Access Orchestrator
POTS	Plain Old Telephone Service
PPP	Point-to-Point Protocol
PPPoE	PPP over Ethernet
PW(E)	Pseudo Wire (Emulation)
QoS	Quality of Service

RBFS	Rtbrick File System
RD	Remote Device (DT specific CPE)
RFC	Request For Comment
RID	Remote ID (Line ID)
RaBaPol	RADIUS Based Policy Control
RADIUS	Remote Authentication Dial-In User Service.
RMK/SMK	Service/Resource Management for RD
RR	Route-Reflector
SDH	Synchronous Digital Hierarchy
SDN	Software Defined Networking
SE	Service Edge
SEBA	SDN-Enabled Broadband Access
SG	Service Gateway
S-GW	Serving Gateway
SID	Segment (Routing) ID
SNMP	Simple Management Network Protocol
SR	Segment Routing
SSH	Secure Shell
SUSE	Gesellschaft für Software und System Entwicklung
SyncE	Synchronous Ethernet
ToD	Time of Day
TP	Termination Point
TSA	Technology Specific Agreement
TSF	Traffic Steering Function (c.f. BBF WT 474)
T-TAG	Termination Tag (in case of double VLAN tagged frame could be called “Inner” tag, in standard documents often named C-TAG)
UNI/NNI	User-Network or Network-Network Interface
UP	User Plane
UPSF	User Plane Selection Function (c.f. BBF WT 474)
VIP	Virtual IP address
VLAN	Virtual Local Area Network
VPWS	Virtual Private Wire Service (MPLS)
VRF	Virtual Routing Function (MPLS)
VRRP	Virtual Router Redundancy Protocol
XGS PON	10 GE Passive Optical Network
Z900	900 DT Central Offices

2 Setting the scene

2.1 Architectural Overview of DT Deployed Solution

Currently DT offers FTTC/FTTH/Business Access as shown in the picture below. As an extension, FTTB is in early deployments. FTTH is ramping up. The access network today is BNG-centric according to a BBF TR-101/TR-178 model using edge BNGs that directly connect to MSANs.

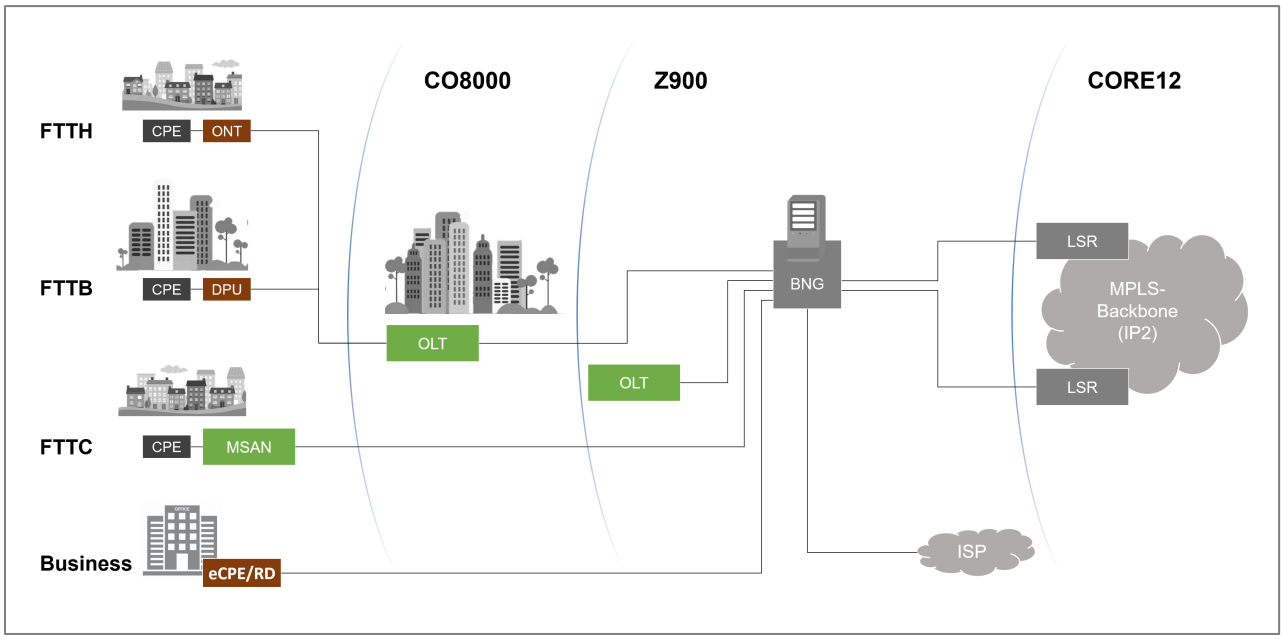

Figure 1 DT Current Access Deployment

Existing deployments are mainly copper using Multi-Service Access Nodes (MSANs) mainly in street cabinets (~200K) serving Retail, Business and Wholesale customers. Business Services are served by Remote Devices (RD) with point-to-point business connections and MSANs offering xDSL. POTS access is also deployed via MSANs infrastructure and POTS line cards are used in the same street cabinets. For POTS, the BNG serves as (IP) backhaul towards the DT VoIP platform. All access variants and customer segments (Retail/Wholesale/Business) are realized on the BNGs which are used to terminate user (subscriber) sessions and provide the physical handover points (A10NSP) to wholesale providers. There are a total of about 2.5k BNGs in approximately 900 locations making the network difficult to manage and evolve in a cost-effective way. Especially the need to terminate all kinds of subscriber on the BNG line cards directly attached to the MSANs and the static A10NSP handover points prohibit more flexible deployments.

2.2 Disaggregation, Cloud Native and Whitebox Hardware

The ‘pain-points’ of the existing service deployment models are derived from the dependencies on specific ‘vendors’, their roadmaps and the costs associated with hardware and ‘vendor lock’ in e.g., developing new features in the network. There is a need for DT to be in a position where new solutions can be quickly prototyped, tested, and introduced in addition to flexibly adding new ‘whitebox’ hardware platforms.

The “who dares wins” white paper outlines a path to evolving networks and software systems towards a more flexible and independent architecture.

Thus, in response to these ‘pain-points’, DT developed the ‘A4’ solution (Access 4.0) which is based on lessons learned from the years of service deployment experience, best practices emerging in the areas of disaggregation, SDN, NFV, modular IT architectures and closed-loop IT automation from the cloud. Access 4.0 also incorporates BNG disaggregation principles as defined by the Broadband Forum (BBF) TR-384 and TR-459 and is aligned with Open Networking Foundation (ONF) and SDN principles as exemplified by the SEBA/VOLTHA programs.

The A4 solution re-defines the access-network which becomes disaggregated, automated and partly virtualized. This re-architecture of the access-networks allows for rapid prototyping of new solutions without the need for complete reliance on specific vendors and the ability to quickly introduce new technologies and hardware platforms and decouples software and hardware lifecycles providing modularity and flexibility by exchangeability.

The goal of this disaggregation is to reap the benefits associated with an open-source / community-based ecosystem that brings economies of scale, rapid innovation with interoperable software modules, reduced Capital Expenditure (CAPEX), and Operational Expenditure (OPEX), as well as gain agility for new service introduction.

The ONF SEBA Reference design 2.0 describes the high-level target architecture as shown in the following diagram:

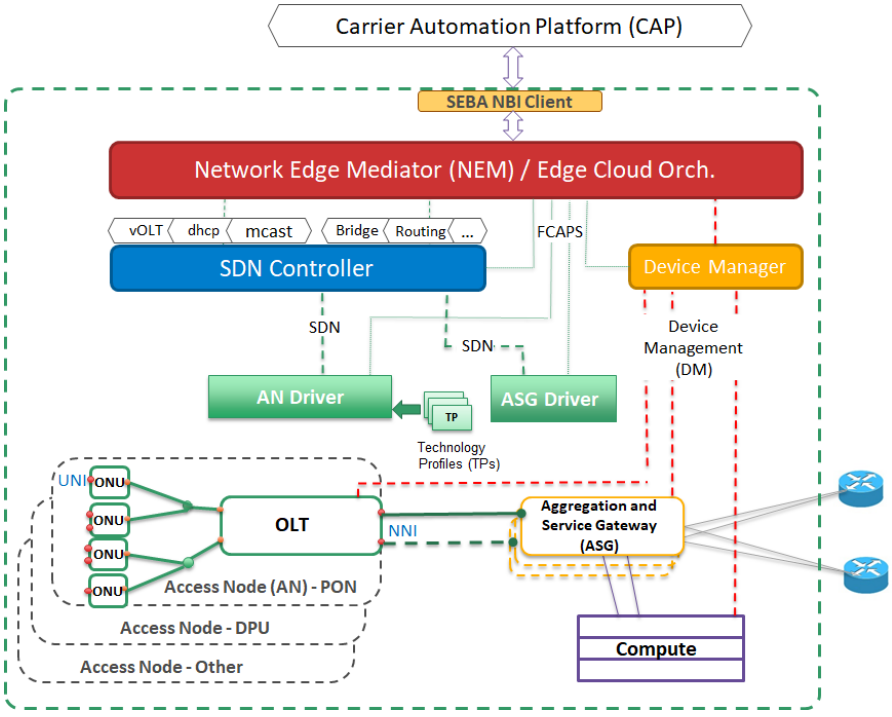

Figure 2 ONF SEBA Reference Design Architectural Diagram

The previous Figure describes Access Nodes connected to an Aggregation and Service Gateway (ASG) layer defined as: one or several switches/routers in a connected topology that supports options for layer 2 aggregation, layer 3 service aggregation, Service Edge/BNG (and/or S/P-GW) functions and supports composing an SDN-controlled Leaf-Spine Fabric. Each customer port (FTTH, B, C) is addressed by one dedicated tunnel (e.g. pseudowire) ID, so each port can be routed within the Fabric to an arbitrary endpoint using a method later developed by BBF as “Subscriber Session Steering” (BBF WT-474). The Fabric provides a mesh for the localized ANs to access the network, and a management network between the compute functions, the ANs, AN drivers and ASG drivers.

SEBA further comprises the open source VOLTHA framework to manage and control white box OLTs as well as the NEMS (Network Management) layer. In meantime, the VOLTHA project had been transferred into the Linux Foundation.

While the SEBA architecture is still covering the fundamentals of the Access 4.0 system, in selected areas like session steering and northbound interfaces, the BBF’s Cloud CO framework has in meantime significantly advanced. Alignment with the Access 4.0 architecture is provided by DT and Radisys teams working jointly at the BBF community.

2.3 Disaggregated BNG Control and User Plane Separation

As stated above, BBF has picked up significant work relevant to Access 4.0. Driven by the OpenBNG white paper, the disaggregation of the BNG had been addressed by TR-459. In TR-459, the BNG is separated into control plane (BNG-CP) and user plane (BNG-UP) entities that can scale separately, which is mainly advantageous in case the BNG function is deployed in data centers on general purpose servers. TR-459 mandates three interfaces between control in user plane. A management interface, a control plane interface (based on PFCP) and an interface to re-direct packets arriving at the BNG user plane. The latter is similar to concepts from the OpenFlow protocol where “first signs of life” (FSoLs) can be sent to a detached controller that can then make decisions on policies to be applied and enforced.

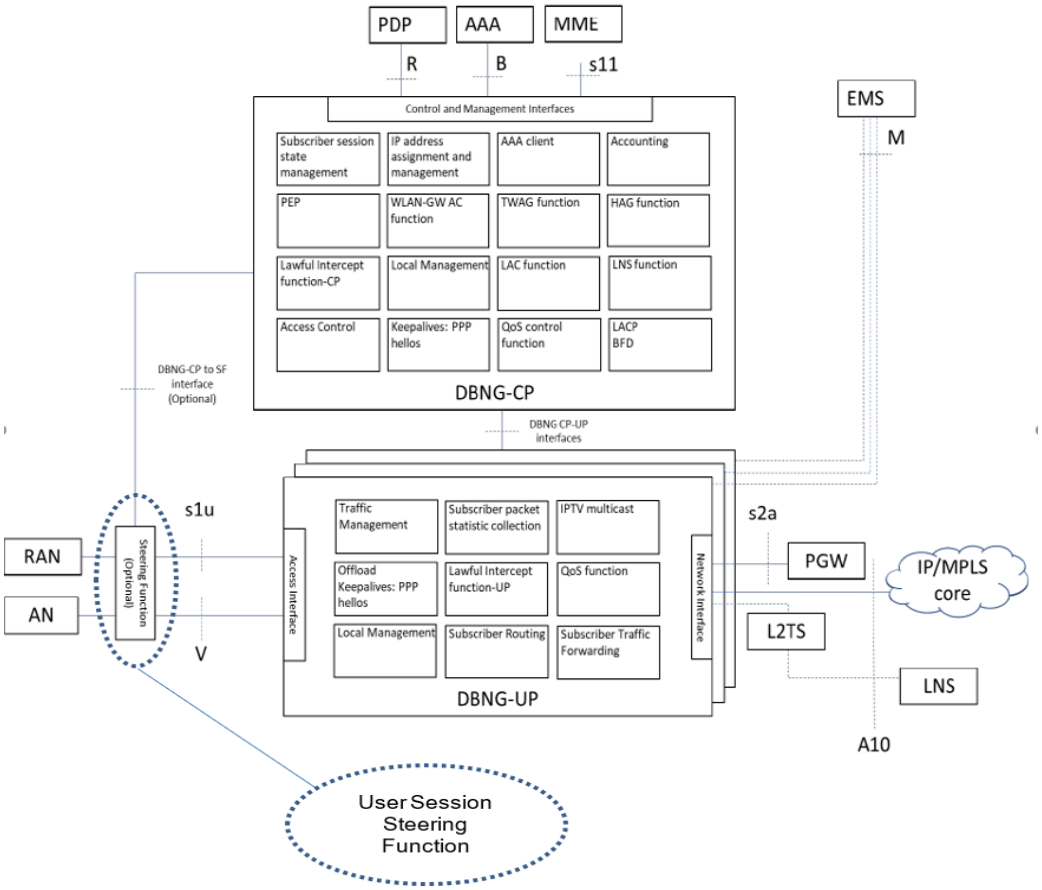

Figure 3 Control- and User-Plane separation (BBF TR-459 Figure 2)

This disaggregated model with separation of the BNG control and user planes allows for a single control plane to interface with many DBNG user planes leading to the additional optional ability of the control plane to use SDN principles to configure the Access Node and connecting Service Edge Switches to steer user sessions to specific user planes as shown in the BBF TR-459 (Figure 3 and Figure 4).

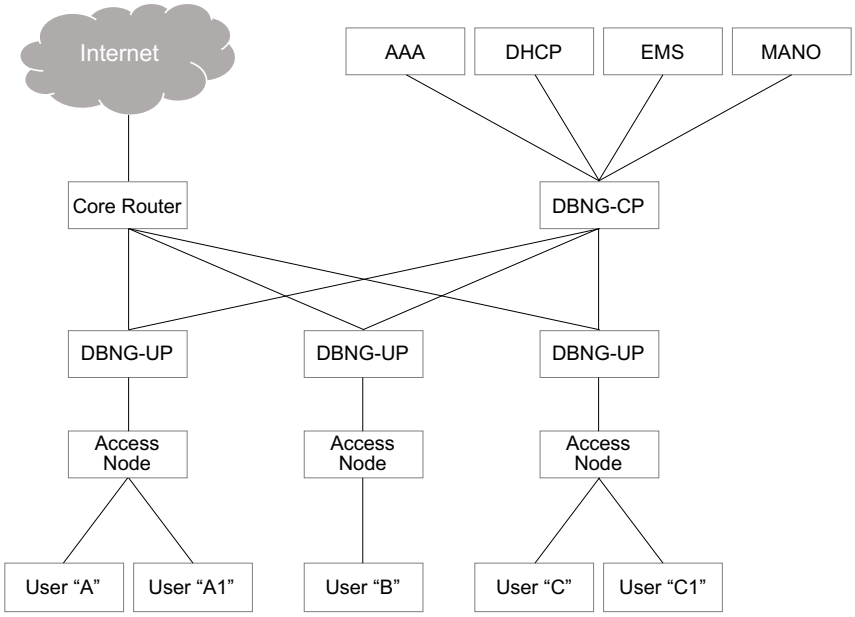

Figure 4 DBNG with Single Control Plane and multiple DBNG-UP (according to BBF TR456)

The disaggregated BNG by its nature allows to cover a certain level of redundant user planes since the control plane is enabled to move active subscriber sessions from one user plane instance to another one. This function uses a grouping capability for the existing sessions. In order to provide a fully functional session handover including the access and aggregation network, it has become very obvious that a function to steer the sessions to the desired new user plane instance will be needed. Thus, the BBF is in the course of developing the subscriber session steering framework (WT-474). Redundancy is one of the key use cases for session steering.

The other big target for control plane user plane separation has only been partly achieved: Interoperability between different vendors for BNG CP and BNG UP. The standard (even in issue 3) still leaves specifics open and does not cover all session types needed (e.g. L2 wholesale is still missing and may be addressed in TR-459i4).

The Access 4.0 architecture does as of today not require the control plane user plane split as the BNG control plane is always co-located with the physical device providing the user plane function. This may change in further evolutions of Access 4.0, especially when considering to add x86-based service edges to the A4 PODs (topic currently under investigation).

The key disaggregated BNG features that Access 4.0 is using are: Hardware independence as the BNG runs on white boxes and the separation of the session termination (SE, Service Edge) from the routing function. This maps to the OpenBNG “Cloudified, functionally separated” option.

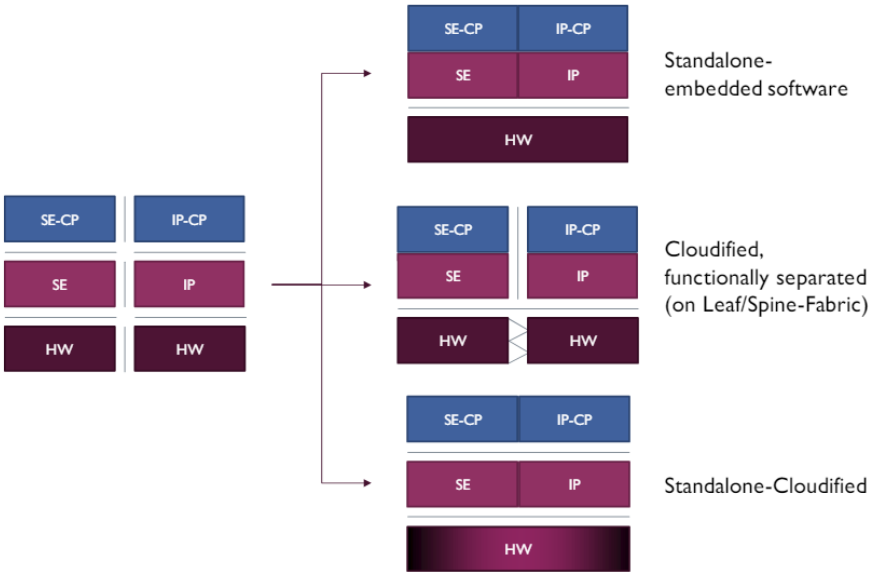

Figure 5 OpenBNG Options (from OpenBNG white paper)

In terms of the SEBA architecture, the A4 BNG is implemented in the ASG layer.

Coming back to subscriber session steering: As already depicted in the section on ONF SEBA, there is a need to control the path of individual subscriber sessions towards the desired entity that provides the session termination (the SE=Service Edge or SG=Service Gateway).

BBF WT-474 defines use cases, architecture and requirements around dynamic subscriber session steering.

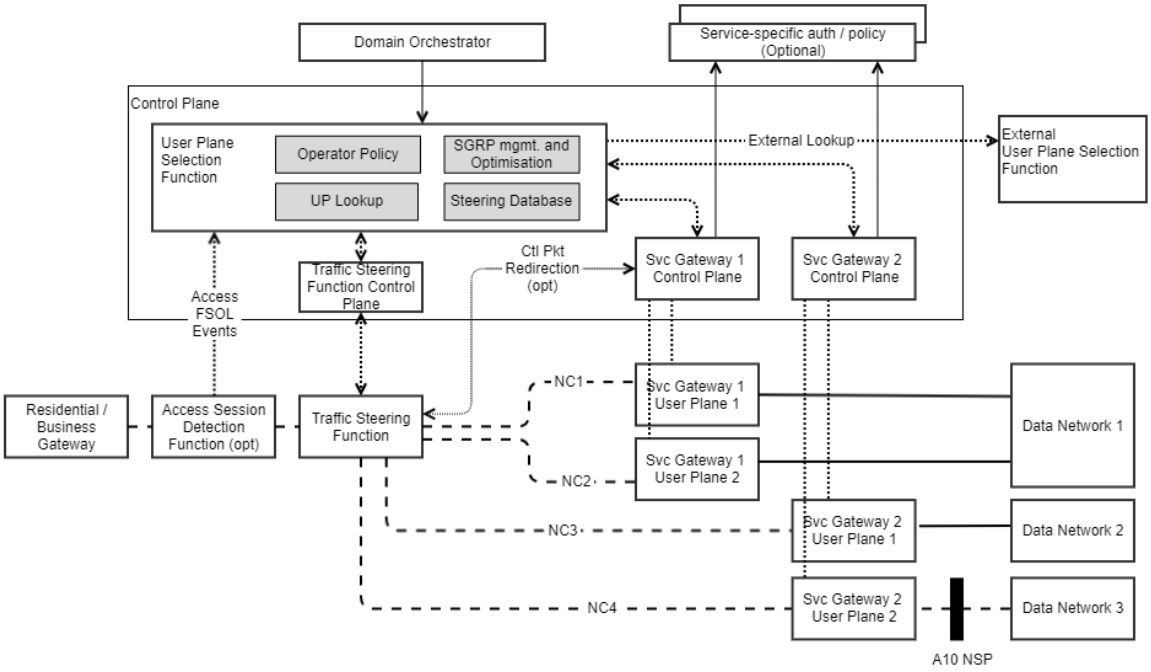

Figure 6 BBF Subscriber Session Steering Framework (from WT-474 draft, not yet published)

The key function is the UPSF (User Plane Selection Function) that – upon three types of triggers (a1, a2, a3) – creates and manages session tunnels to the BNG (=SG). The UPSF instructs the TSF (traffic steering function) to configure the TSF user plane to provide the steering of the sessions on the network layer.

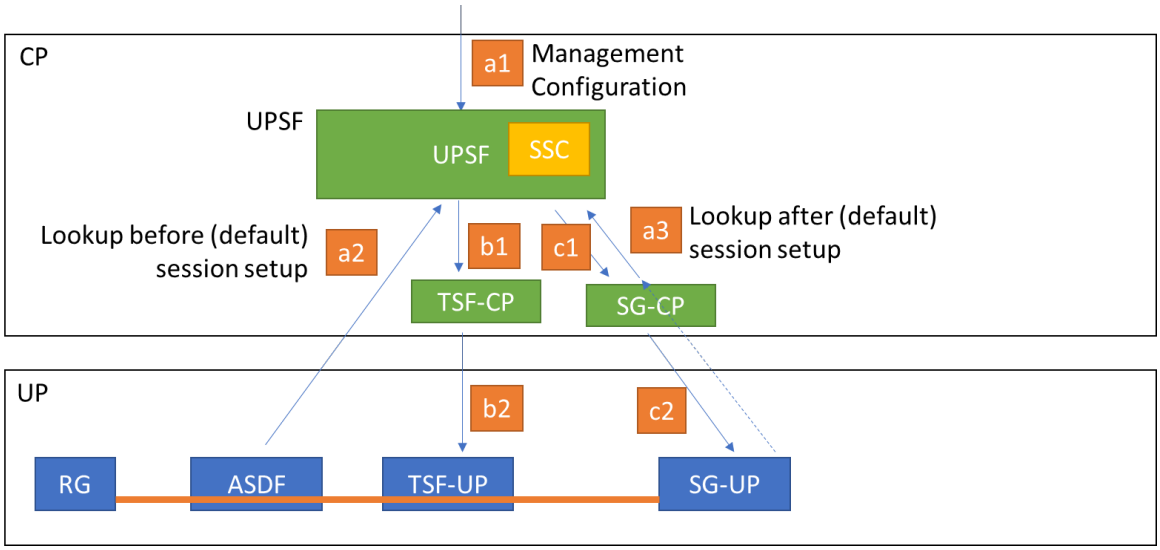

Figure 7 UPSF and related functions in WT-474 latest draft

BBF WT-474 defines three cases for session steering:

1. **Case a1 Static Session Steering**: traffic for a subscriber is configured such that it is always steered to a specific destination SG-UP (Service Gateway-User Plane)
2. **Case a2 Access Network Triggered Session Steering**: access network triggered steering covers the cases when the decision on where a subscriber session should be steered, and the corresponding implementation of that decision, is triggered by an event from the access network. These trigger events might include physical layer changes such as access port coming up, or a packet event. Examples for such events are PON via ranging or DSL showtime but any other mechanism that follows the principle providing a first sign of life (FSoL) can apply.
3. **Case a3: Service Gateway Triggered Session Steering**: Service Gateway triggered steering covers the cases where the decision as to where a newly connecting subscriber session should be steered, and the corresponding implementation of that decision, is triggered by an event from the Service Gateway control plane.

Access 4.0 primarily implements a model based on option a2 above and uses an **Access Session Detection Function [ASDF]:** The ASDF is used in cases where automated access session detection for L2 path steering and forwarding is required. It is a component that detects when a CPE is attached to the AN, through events such as DSL or PON line synchronization, or by identifying an event such as a packet being received on the access line after a long inactivity. Such events can be considered a First Sign of Life (FSoL).

The ASDF signals the detected attachment event to the Control Plane, after annotating the notification with information related to the attachment position of the subscriber on the AN, such as Line-Id, and with other logical access-side information, such as VLAN ID, GEM, ONT Serial Number, CPE MAC address, etc.

Session Steering based on FSoL is shown in WT-474 as below:

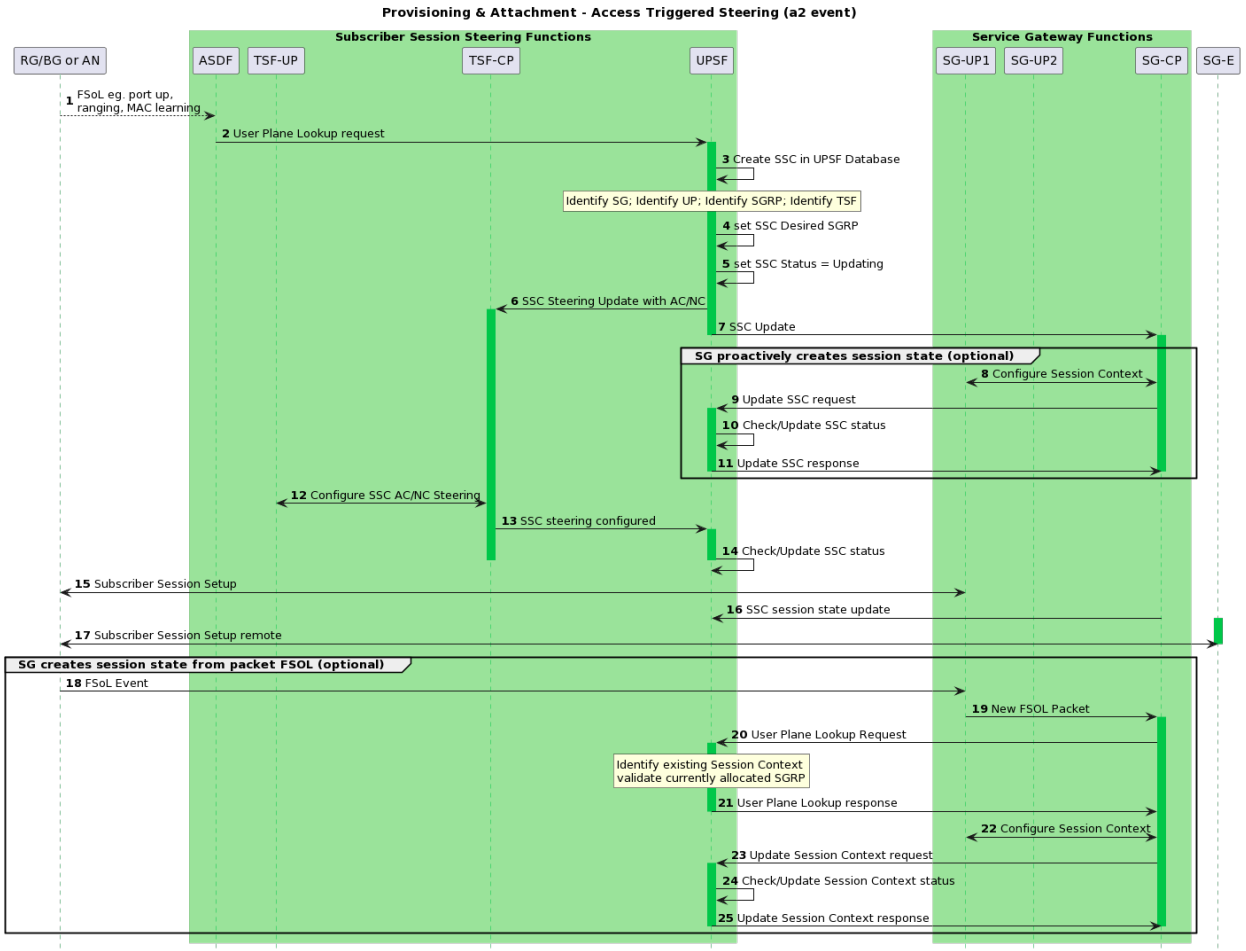

Figure 8 Session Steering flow based on FSoL in WT-474 latest draft

Once the Session Steering Function has created the Subscriber Session Context (SSC) it will set the Session Status to be “Empty”. The Session Steering Control Plane will connect the subscriber with the Service Edge and the actual Subscriber session can start with control messages from the subscriber access side to the Service Edge. The Sessions can terminate in locally available DBNG User Planes (called SG-UP here) as well as in external BNG (called SG-E). Both cases exist in Access 4.0. Local termination for DT’s subscribers and remote (SG-E) termination for Layer 2 Wholesale.

2.4 Access 4.0 Key Guiding Principles

Access 4.0 is designed along the following key guiding principles:

1. Designed for change; meaning the modularity in SW and HW provides the flexibility in life cycle of SW and HW components.

2. Decouple technology from OSS/BSS: Means that the POD and its components decouples OSS/BSS life cycle as well, which means it eliminates dependencies between NT/IT

3. Decouple SW/HW life cycle and provide modularity and flexibility by exchangeability.

4. NEMO/A4-EMS simply hands over the data only, which means each POD holds its own inventory

5. Technology-specific items are isolated at device controllers (“driver-level”)

6. Two Stage Service Activation:

   i. Empty Session Establishment: triggered by ‘First Sign of Life’ (FSoL); Select Service Edge and set up AN and Fabric Path
   ii. Session Establishment: Policy resolved, Authentication, Services Configured (Note L2BSA Services require a cross connect and MPLS label encapsulation across the Fabric and AAA services are provided by the Wholesaler).

7. Event driven design: either events in terms of message requests from external OSS/BSS/User interfaces or events in terms of ‘business events’ which generate actions in response.

8. Separation of the fulfilment and planning concerns; The OSS-IT creates planned resources which satisfy specific network functions as logical resources independent of the actual physical device which fulfils that role.

9. The ‘Las Vegas Principle’ which describes the requirement that the A4 POD manages itself and only reports what is need by external systems. All network element configuration is generated by the A4 POD using templates for specific service profiles driven by ‘Business Events’.

10. Open source/white-box based.

11. Zero Touch Provisioning

2.5 The global product Connect Modular Broadband (CMB) and Access 4.0 (A4)

Access 4.0, originally the name of a project or initiative, has gradually also become the name of the solution for Deutsche Telekom. The initiative Access 4.0 started in May 2016. Over time several technical prototypes and intermediate solutions were created. Today’s "final” version 1.0, if you will, is the one DT takes to rollout. But as we write this document, DT works already on further improvements with for instance a new topology (e.g., POD Topology V2). In the further course of the text, A4 is used as the name for DT’s solution.

In 2023, the CMB product was launched. Initially, CMB was a mere copy of A4. But Radisys took it to market as-is to ‘test the waters’. The feedback from potential customers was very helpful to better understand the actual market demand that goes beyond DT's requirements. Yet since CMB is currently unprecedented in the market, it took and still takes a lot of efforts and explanation because there is no other product offering a similar scope of e2e functionality. A4 / CMB paths the way to new grounds. The short summary of the

customer feedback, though, is that A4 and CMB are different in certain areas. For instance, customers would like to leverage their own telco cloud also for the POD servers. Moreover, A4 puts the entire Fabric in one or two racks next to one another. But potential CMB customers have a spatially distributed topology, where Leaf and spine or even the spines themselves are several kilometers afar. And this in turn requires an in-band connectivity also for leafs and so on.

However, the “beauty” of A4 and CMB is that it all is built on microservices and in a modular fashion. We are able to benefit from advances in both A4 and CMB. Mid-term we want CMB to be the master branch, where A4 is an incarnation of CMB.

3 Access 4.0 Architectural Model, Evolved Deployment Model

The implementation of a new disaggregated and virtualized access-network solution is embodied in the DT Access 4.0 (A4) architecture. The A4 Access topology currently remains the same (however Access 4.0 is topology independent) and mini data centers will be built (usually with 20k to 100k subscribers) representing one of the 900 locations in the DT network as shown below in Figure 9. The goal is to apply principles of SDN/NFV and disaggregation to all Access Nodes (AN) where possible e.g., the Legacy MSAN may be an exception.

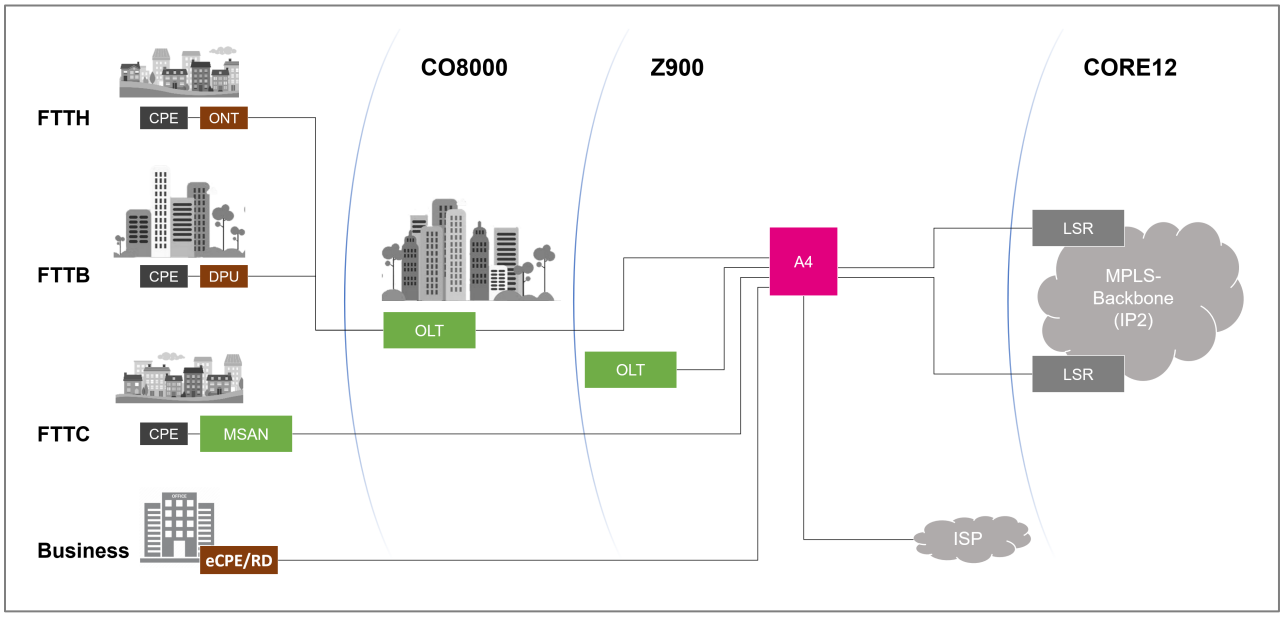

Figure 9 Access 4.0 Deployment

An example deployment of an A4 Pod in a Mini Data Centre is shown in Figure 10 where the A4 POD interfaces with a Centralized A4-EMS/NEMO (NEMS) component interfacing to the DT OSS/BSS systems for Planning and Fulfilment. In the A4 POD todays BNG function is disaggregated with the Leaf Switches providing the Service Edge including:

1) Terminating Subscriber PPPoE sessions and enforcing QoS for Retail Services,
2) An L2TP Access Concentrator (LAC) function for the L3BSA Service
3) A VLAN Cross-Connect/Pseudo-wire encapsulation function to switch L2BSA traffic to an A10NSP Switch for handoff to third party Service Providers.

The Spine switches provide MPLS/LER capabilities interfacing with the IP2 Core network, and the Access Nodes themselves OLT, RD, MSAN, DPU may be geographically distributed and not necessarily co-located with the Server Cluster and Fabric itself. The DPU hardware is the same as used in the current BNG Architecture based deployments but is disaggregated in the new A4 deployments and may be white box in future.

The A4 POD additionally supports a Lawful Intercept Box (x86 based server platform) for monitoring subscriber services as required by Government Regulation.

The goal is to reach feature parity between the current BNG based deployments and the Access 4.0 Disaggregated Solution and then build on the flexibility and scalability of the solution to accelerate/simplify future changes in the network deployments.

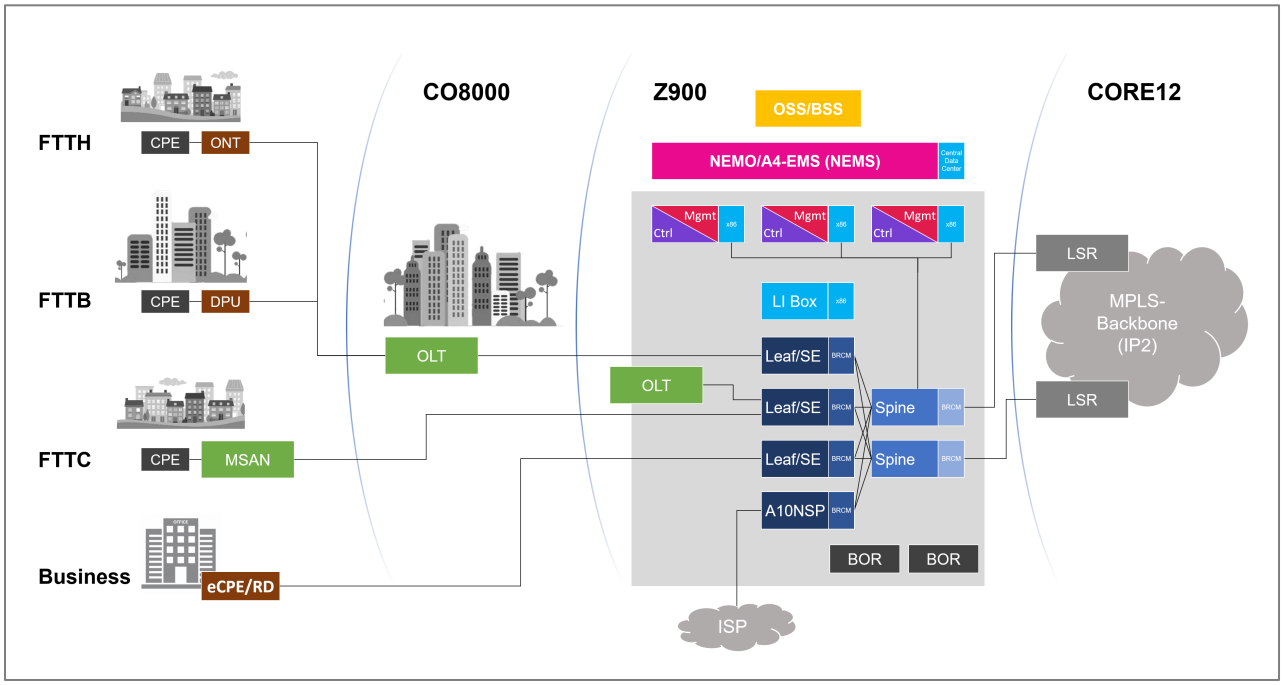

Figure 10 Access 4.0 Mini Data Center Deployment Model

3.1 A4 POD High Level Architecture

Access 4.0 abstracts the control plane and management plane of traditional access hardware (e.g., OLT, switch hardware etc.) and implements the same, in microservice-based scalable architectures using virtualization concepts. This enables service providers to disaggregate their networks by moving control and management functions from proprietary black boxes to commodity platforms over vendor neutral and open interfaces defined by open-source forums like ONF Virtual OLT Hardware Abstraction (VOLTHA). This disaggregation allows service providers to leverage white box platforms for specific access and applications. Using white box platforms and Open-Source platforms breaks vendor lock-in while Access 4.0 adds programmability to the network, thereby enabling service providers to leverage SDN-type control and manageability.

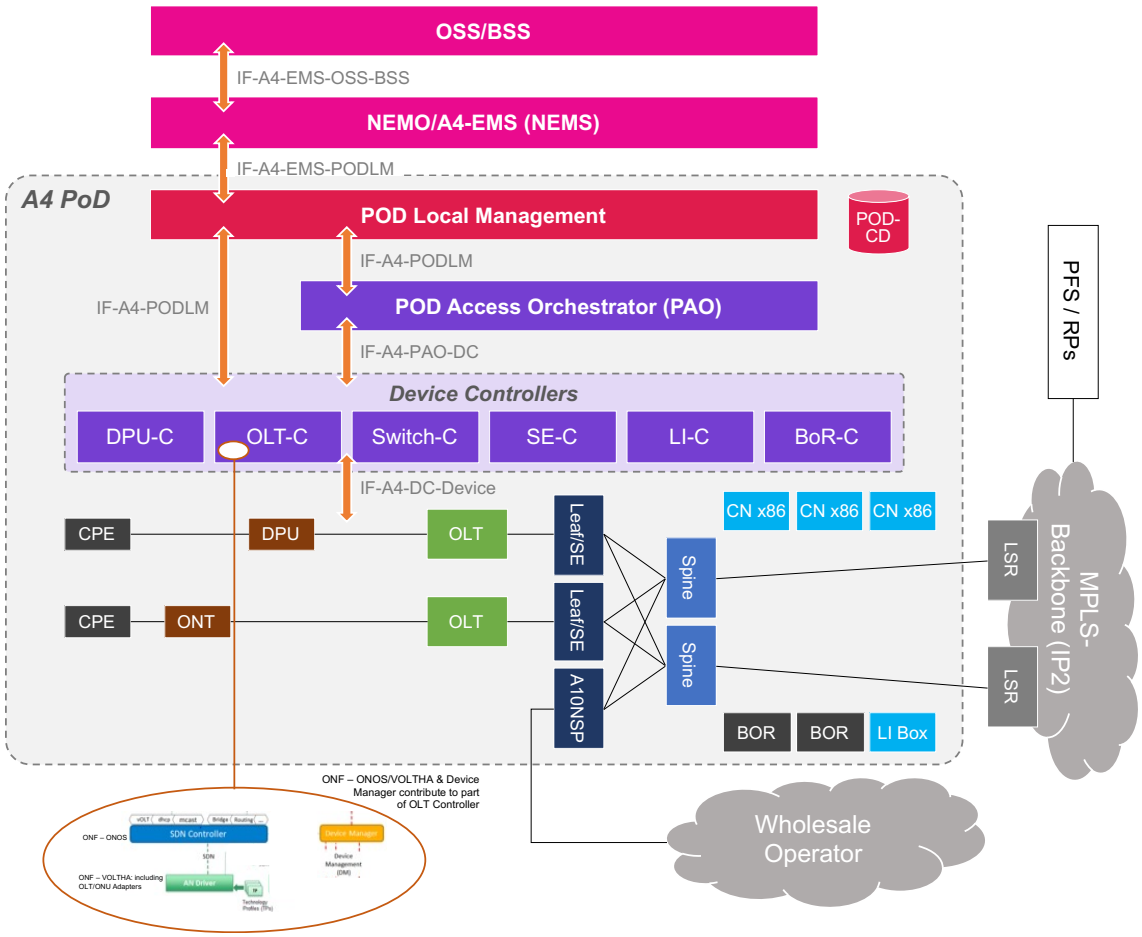

Figure 11 Access 4.0 and A4 POD High Level Architecture

Access 4.0 aims to deliver a common software platform that can be used across several broadband access technologies such as:

- Gigabit Passive Optical Networks (G-PON),
- 10 Gigabit Symmetric Passive Optical Networks (XGS-PON),
- Combo PON(GPON/XGS-PON)
- Fiber P2P (1G/10G termination)
- FTTB/DPU (G. fast)
- FTTC

The Access 4.0 solution additionally includes the Disaggregated BNG capabilities with Leaf switches supporting the Service Edge functionality and dedicated Spine switches supporting the Routing capability, interfacing with the IP2 Core Network. Additional switches provide the A10NSP handoff capability.

The DT T-DCN network connects the OSS/BSS Systems to the Centralized NEMO/A4-EMS and the Centralized NEMO/EMS to the A4 PODs as shown below in **Error! Reference source not found.**. The NEMO/A4-EMS is hosted on Telekom's “Das Schiff” Cloud (Kubernetes hosting platform of DT). It should be noted that the management network could also be “in-band’ in other deployments.

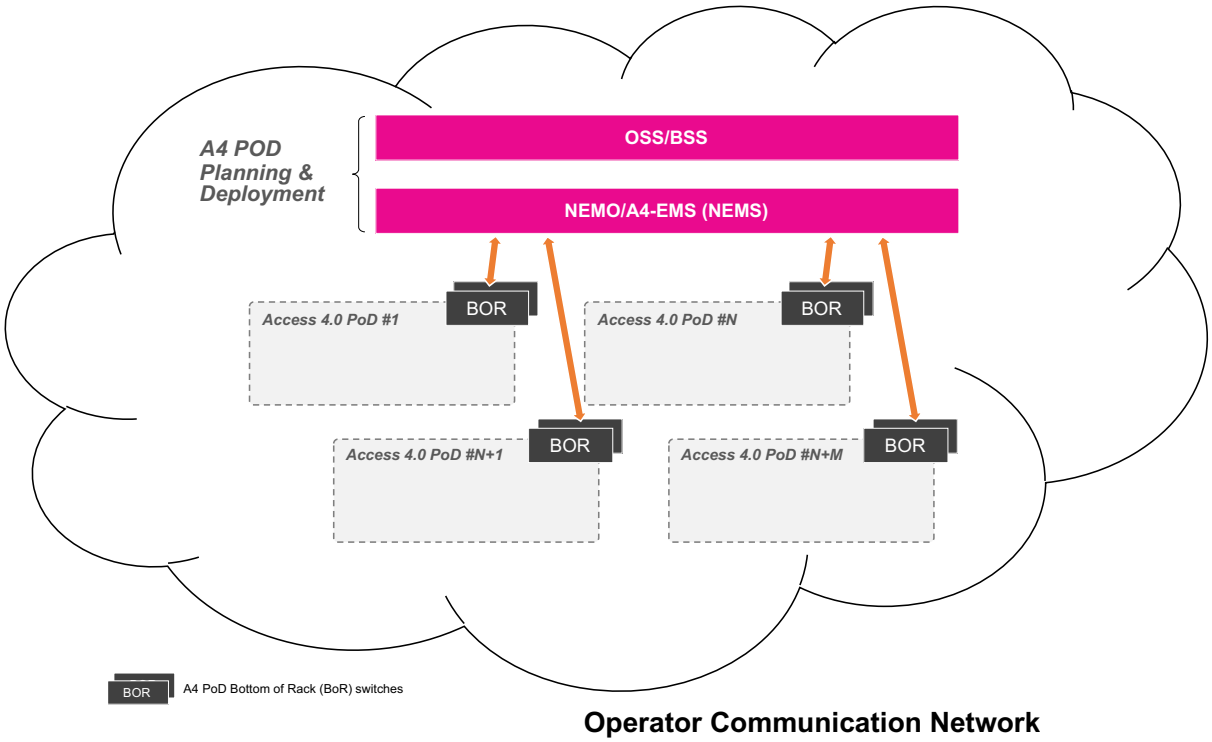

Figure 12 N+M A4 PODs Planned by OSS-IT and Deployed by NEMO/A4-EMS (NEMS)

3.2 A4 POD Traffic Models and Scalability

The Leaf switches where the Service Edge and the subscriber QoS is implemented typically supports a mixture of traffic types:

- Retail
- L2TP(L3BSA)
- L2BSA
- Business (RD)

From a Data-Plane perspective each type of service requires a certain number of Queues to provide its' required per subscriber QoS. The Queue resources required to provide the necessary QoS for the different services require at least the following allocations:

- Retail: 1/4/8 Queues
- L2TP: 4 Queues
- L2BSA: 4 Queues
- RD: 4 Queues

The current Leaf hardware has 128k queues. Assuming 75% of the interfaces consume 4 queues and 25% consume 8 queues, a total of 25600 subscribers can be provisioned in theory. Since there are also limitations in terms statistics and metering A4 aims a target of 22000 subscribers.

Find information on currently tested and supported scale profiles on:
https://wiki.telekom.de/display/ACC4/Subscriber+Scalling+24.1.1.1

The A4 POD Data-Plane should be capable of supporting the target subscriber scale with this typical mix and define the scaling numbers with 100% Retail Subscribers and 100% Non-Retail Subscribers.

3.3 Key Physical Components of an A4 POD

The Figure below breaks down the A4 POD into a set of key 'Orderable' components. The "purple" elements are part of a POD Bill of Materials (BOM). The other grey components are required for the A4POD to provide service but are external, for example, in Germany the ONT can be bought at a retail store and a (certified) ONT should work with the A4 Solution.

Physically the Access 4.0 model defines a POD (Point of Delivery) where a POD consists of a set of 2n+1 Servers (n >= 1 for high availability purposes) and the POD Hardware; Access Nodes (e.g., OLT/DPU), Leaf/Spine/A10NSP/BOR switches, Lawful Intercept Server.

The disaggregated Control and Management Plane software executes on the POD Servers including the hardware abstractions and the Controllers required to interface the virtualized Management and Control Planes with the POD hardware components themselves.

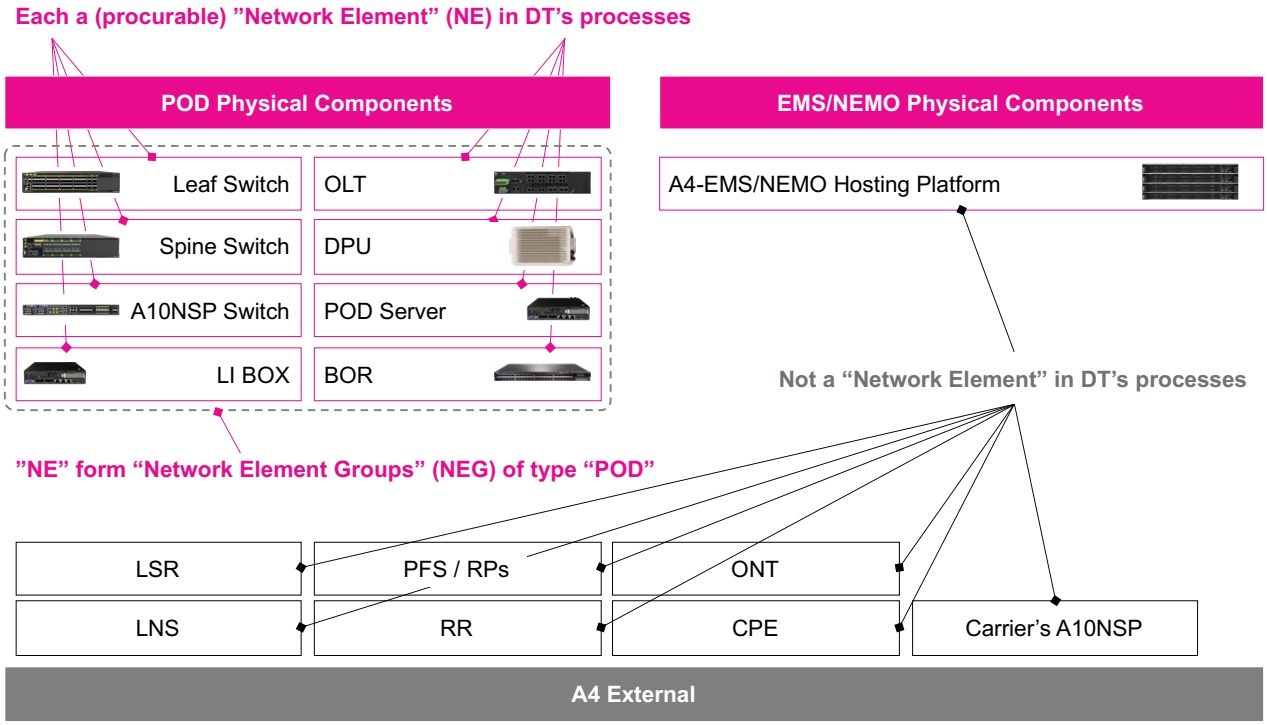

Figure 13 Access 4.0 components

An A4 POD is built from:

- A number of standard Server Platforms (generally, 2n+1)
- N Leaf Switches (where N is usually between 1 and 5) each Leaf Switch targeting up to 22K subscriber terminations
- 2 Spine Switches for interfacing with IP2 Core Network
- 1 A10NSP-Switch for interfacing with third party service providers
- 1 LI-Box – for providing Lawful Intercept capabilities.
- N White/Grey box OLTs (Must be compliant to ONF VOLTHA architecture)
- N DPUs (where FTTB is supported by the A4 POD).
- Other ANs e.g. MSAN, RD
- 2 Bottom of Rack Switches (BOR)

The model is to support three Servers (2n+1, n=1) in a highly available cluster running Kubernetes where the Disaggregated Control and Management components of the A4 POD reside.
The Fabric has two Spine switches connecting to the IP2 Core network with several Leaf switches (depending on the number and type of customers) providing the attachment point for OLTs and in future other Access Devices.

These Orderable members of an A4 POD (Network Element Group) are modeled as Logical Resources in the OSS-IT Management Plane and are provided to the Centralized EMS for deployment as a Network Element Group (NEG) Logical Resource and the NEG component Network Element Resources are modelled to represent individual types of Network Functions. This type of modelling abstracts the Network Function of a Resource from the actual Physical Device and allows the OSS-IT to be immune from the effects of changing from one Vendor Device to another or even a new generation of a Device from the same Vendor.

3.4 Access 4.0 Data-Plane

3.4.1 Fabric Overview

The Fabric of A4 itself consists of two building blocks:

1. A Spine layer that interconnects to the:
    - Leaf Switches
    - A10NSP switch
    - IP2 (DT MPLS Backbone) Connectivity
    - Servers
    - LI-box
2. A Leaf Layer:
    - Broadband-Access (OLT, MSAN, RD)

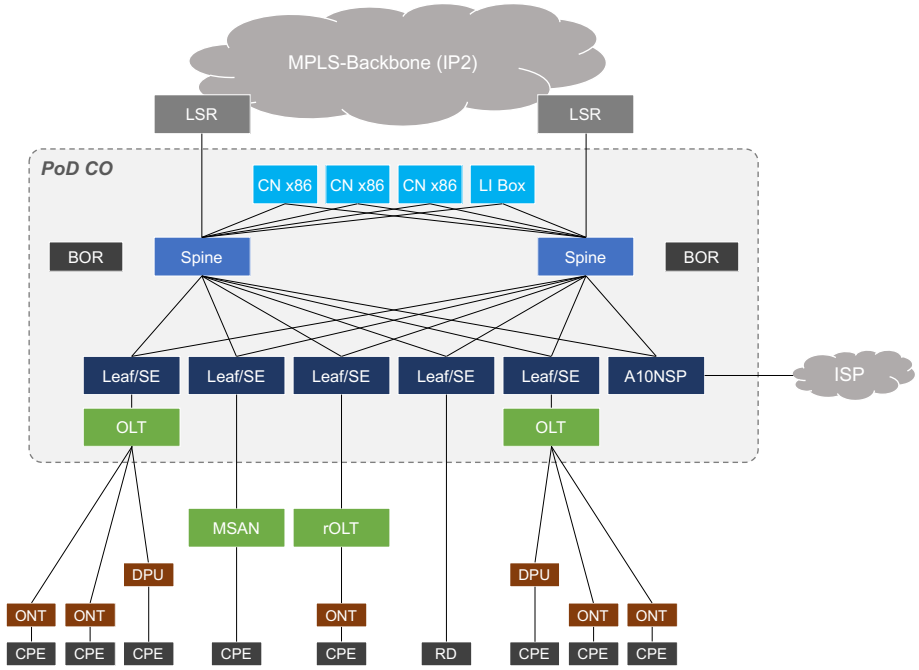

Figure 14Access 4.0 Leaf-Spine Fabric Overview

The minimal Fabric setup consists of two Spine switches and at least one Leaf switch to keep the basic architecture independent from the actual size of the POD and to provide redundant uplinks.

NOTE: This design differs from a standard CLOS architecture to the effect that the WAN connectivity is collapsed into the Spine switch instead of providing dedicated gateways. With that the Spine acts as a gateway between the POD Fabric and the MPLS Backbone of DT. Nevertheless inside of the Fabric the design follows the CLOS architecture principles such as the physical setup as well as load-balancing (ECMP) and redundancy principles.

All network elements of the POD are rack-mounted in 2 different racks in a symmetrical way. There is rack design and a cabling design, that makes sure all PoDs and fabrics are built in the same way.

More details find here: https://wiki.telekom.de/display/ACC4/POD+-+Rack+-+DESIGN

3.4.2 Fabric Underlay – Segment Routing/BGP-SR

The PoD/Fabric is a closed MPLS domain, where the Spines implement gateway for the overlay services. From an outside networking perspective (the IP2 and RRs), one entire PoD/Fabric appears like a single node.

Inside of the Fabric A4 makes exclusive use of MPLS, with BGP IPv6 underlay, for the data-plane to forward traffic across the Fabric. MPLS (BGP-SR RFC 8669) is chosen because it has proven to offer operational simplicity and scalability while keeping the flexibility to implement various services (e.g., layer-2 VPNs, layer-3 VPNs, etc.) on top of it. In addition, an MPLS data plane dramatically reduces the size of the FIB table by encapsulating packets and thus hiding customer layer-2 and layer-3 headers within the Fabric as much as possible.

To implement the MPLS data plane, segment routing concepts are used. In segment routing, there are two different kinds of MPLS labels known as segment identifiers (SID):

- Prefix SID which represents an IP prefix (or subnet) and is unique within the Fabric. The node SID is a special prefix SID which represents the node itself (i.e., the loopback prefix)
- Adjacency SID (adj-SID) which represents a link and is only locally significant to the node attached to the link

To provide MPLS transport between any two nodes within the Fabric, a label switched path (LSP) needs to be setup. With segment routing, the nodes within the Fabric only need to know the prefix SID of all other nodes loopback address to establish LSPs to any potential destination. As the node prefix SID is unique, pushing the node prefix SID on top of a packet results in forwarding the packet through the Fabric until it reaches the destination node. The penultimate node removes (pops) the node prefix SID and forwards the remaining packet to the destination inside the Fabric. For sake of simplicity, the A4 POD uses a statically assigned node prefix SID for the nodes which are unique within the POD to avoid the swapping process.

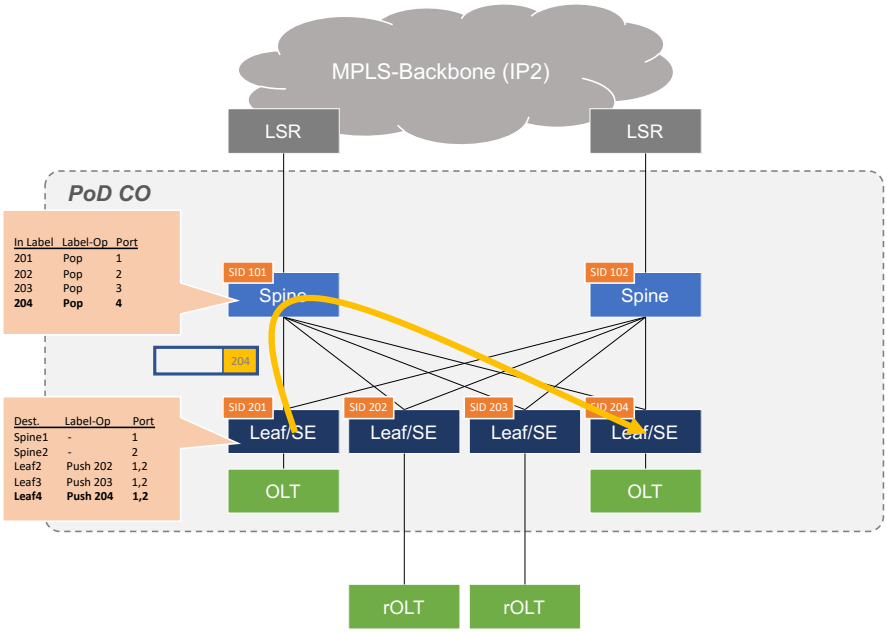

Figure 15 Access 4.0 Leaf-Spine SIDs and Labels

The Figure above illustrates a simple Spine/Leaf setup with two Spine switches and 4 Leaf switches. The number of switches on the Spine layer or Leaf layer depends on the number of ports required. However, every Leaf switch needs to connect to all Spine switches as there is no interconnection between the Spine switches.

The Figure also illustrates the path of a packet, which is received from OLT1, traversing Leaf1 switch to Leaf4 switch through the Spine layer. In this scenario, there are two equal-cost paths to destination leaf4 through Spine1 and Spine2. As the destination switch has the unique prefix SID label 204, leaf1 pushes this label on top of the frame and forwards the frame to one of the Spine switches based on the result of a hash value calculation of the frame header (per flow load balancing). Upon reception, the Spine switch will perform a MPLS label lookup and as result pop the top-most label (aka penultimate hop popping) and forward the frame to leaf4.

Therefore, the number of MPLS forwarding entries for transport underlay in the Fabric increases linear with the number of:

- Spine switches (typically two)
- Leaf switches
- OLTs
- Service Edges (SEs)

Note that non-Fabric devices only need to be aware of potential endpoints (and not the Fabric nodes themselves).

To populate the MPLS forwarding tables on the A4 POD Fabric nodes, eBGP signaling is used. The general concept is based on RFC7938. In this framework, eBGP is used as a single protocol to discover the Fabric topology and distribute the routing information for the Fabric underlay. The Spine layer represents a single autonomous system (AS) while each Leaf switch represents its own unique AS.

In the A4 POD, only BGP is used for various reasons:

- BGP is required anyway to connect to networks outside the A4 POD (e.g., IP backbone) and thus reduces the need for two different routing protocols.
- BGP is more robust and stable when it comes to interoperability.
- BGP is less complex and has a much simpler state-machine than IGP. In addition, BGP has less protocol overhead since it only propagates best paths by default and does not flood the whole link-state database periodically.
- BGP allows to use a single peering session to signal all address families routing information.

NOTE: Another approach to signal the Fabric underlay would be the use of an interior gateway protocol (IGP), such as IS-IS-SR, but for Access 4.0 the underlay is BGP only! in IP Core IS-IS-SR is used.

The eBGP protocol uses multiprotocol extensions to distribute node prefix SIDs for each node in the Fabric. The extensions are described in “draft-ietf-idr-bgp-prefix-sid-08”. Each switch announces exactly one node prefix SID for its loopback address. These SR labels are used for transport purposes only and are allocated from a pre-defined range, e.g., Leaf switch #1 uses 201, Leaf switch #2 uses 202, etc.

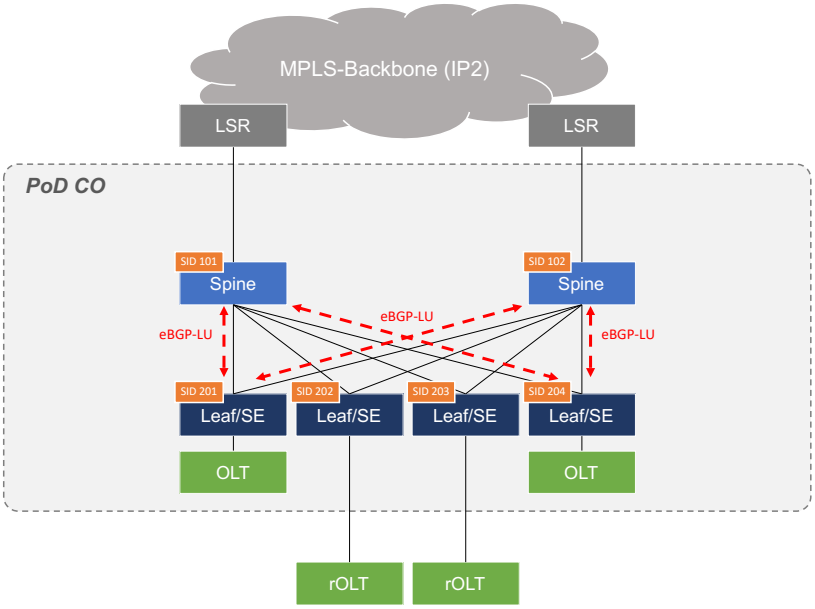

Figure 16 Access 4.0 Leaf-Spine BGP-LU underlay

3.4.3 Fabric Underlay Auto-Provisioning

To simplify the configuration of the switches and the setup of these eBGP sessions as well as to avoid complex IP address management, the following procedure is used for Leaf switches:

- A Leaf switch is installed and physical connectivity between the new Leaf switch and the Spine layer is provided. Afterwards the Leaf switch is booted with a network operating system.
- Upon boot-up, the Leaf switch sends out DHCP request to learn a management IP address (either IPv4 or IPv6) as well as its hostname, which has the format “Leaf<NID>” where <NID> represent the unique node ID.
- Based on the node ID, the switch can calculate its own unique autonomous system number (ASN), e.g., ASN=65100+<NID>, as well as a unique loopback IPv6 address, e.g., LoopIPv6Addr=2001:DB8::1:<NID>/132
- The Leaf switch activates the uplinks to the Spine switches and uses IPv6 unnumbered links with default link-local addresses only
- Upon activation of the uplink interfaces, the Leaf switch sends IPv6 Neighbor Solicitation messages to the Spine switch which responds with IPv6 Neighbor Advertisement messages.
- The Leaf switch extracts the link local IPv6 address of the Spine switch from the IPv6 neighbor Advertisement
- The Leaf switch initiates an eBGP session with the Spine switch using the Spine’s link local IPv6 address as peer address. The Spine switches will always use the same ASN, e.g. 65100. Note, that a pre-shared key will be used to secure the BGP session. In addition, all BGP packets will be sent and accepted with TTL=255 only.
- After BGP session is establishment, BGP routing updates are exchanged.

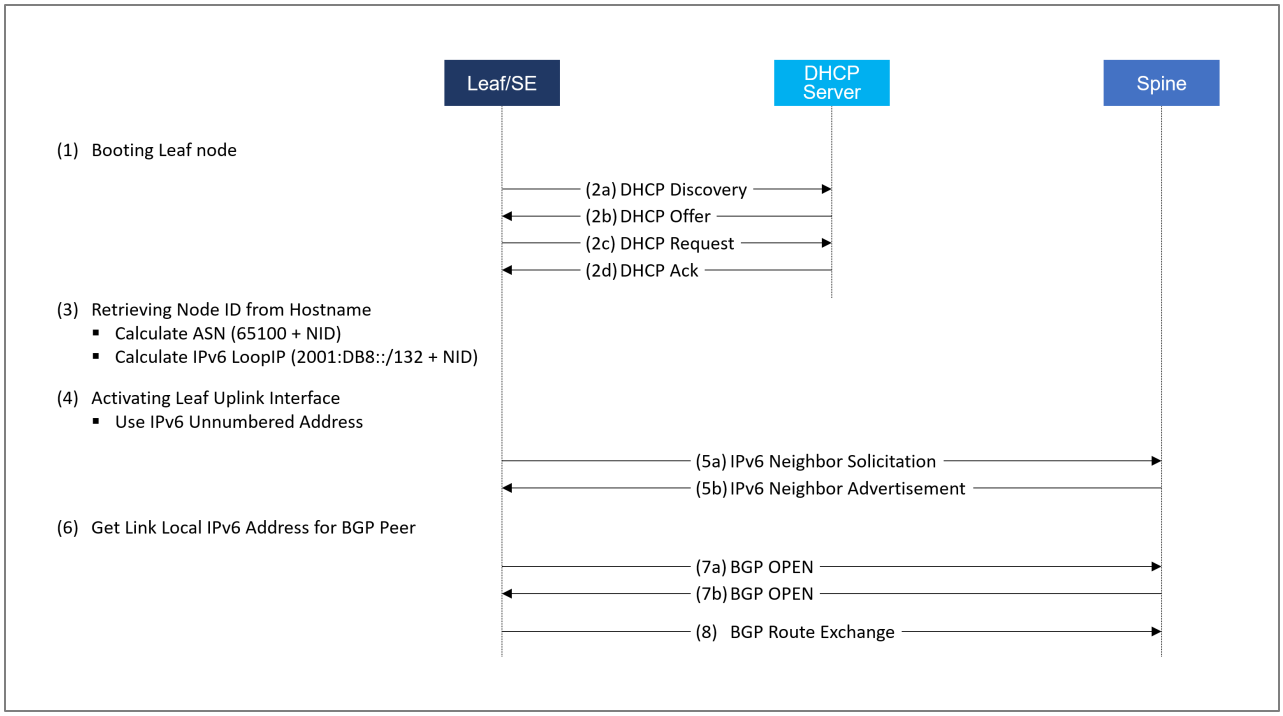

Figure 17 Access 4.0 Leaf-Spine BGP-LU underlay – auto-provisioning

For Spine switches the procedure differs slightly from the one used for Leaf switches:

- The hostname has the format “Spine<ID>”.
- The ASN is always the same, e.g. AS65100, and the LoopIPv6Addr is derived from the Spine ID but within a different address range from the Leaf switches, e.g. 2001:DB8::<ID>/132, in order to avoid conflicts.
- The Spine switches do not initiate eBGP sessions themselves but accept all incoming eBGP sessions to its link local IPv6 addresses if the peer AS is within a pre-defined range, e.g. 65101-65120, and the shared secret matches. In addition, all BGP packets will be sent and accepted with TTL=255 only.
- The eBGP protocol uses multiprotocol extensions to distribute node labeled IPv6 segment routing IDs (SIDs) for each node in the Fabric. The extensions are described in “draft-ietf-idr-bgp-prefix-sid-08”. Each switch announces exactly one prefix SID for its loopback address. These SR labels are used for transport purposes only and are allocated from a pre-defined range.

3.4.4 Fabric Extension to Access Nodes and Service Edges

NOTE: The following chapter is not implemented, as the OLTs do not support MPLS.

For certain types of services, it might be useful to extend the MPLS data-path to other nodes within the POD's data-plane (e.g., OLT-MAC, service edge, etc.), according to BBF TR-178.

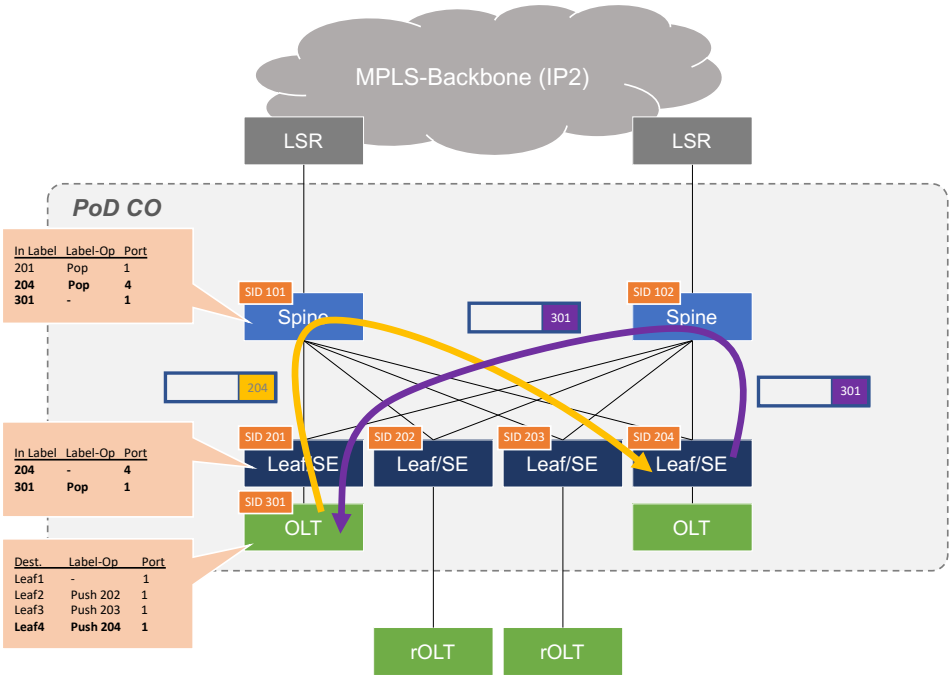

Figure 18 Example of Extended Fabric underlay Data Path (Receive and Transmit Direction)

To further reduce complexity and signaling overhead and to keep access nodes as simple as possible, pre-defined deterministic labels are used to implement the transport underlay. As described in the previous section, the prefix SID labels of the Fabric switches are well-known. Therefore, to reach a particular Leaf switch from any access node, only the unique prefix SID label needs to be pushed on top of the packet and the packet must be sent via its upstream interface to a Fabric switch – no matter which one.

There is no need to implement any routing protocols or dynamic signaling protocols to populate the forwarding table of an access node which can easily be populated by a centralized entity (A4 PAO/SDN Controller) via any Device Controller using the specific Device Adapters and protocols (see the section on Access 4.0 Architecture). The only dynamic information which is needed to build the appropriate packets is the MAC address of the upstream Fabric switch. This information can again be extracted from IPv6 Neighbor Discovery messages. Therefore, the link between access node and Fabric switch uses IPv6 unnumbered addresses.

To receive return packets from the Fabric, the reachability of the specific access node must be announced throughout the Fabric. The Leaf switch attached to the access node must distribute the SID label representing the access node as the access node itself does not use dynamic protocols. Because the Leaf switch must pop this access node SID label (aka penultimate hop popping) and forward the packet to the interface towards the access node, this label cannot be a standard prefix SID label but must be an adjacency SID label. The adjacency segment is associated with the interface of the Leaf switch connecting to the access device in question. This concept is widely known as egress path engineering.

3.4.5 MPLS Backbone Connectivity

POD Spine switches are the boundary between the Deutsche Telekom IP2 backbone and the A4 Fabric and access. The IP2 backbone has 12 central or core locations in 12 German cities, each city has 2 georedundant core sites (POPs) operating LSRs and LERs. There are 74 locations operating LERs in total. Access 4.0 PODs are always directly connected to LSR's. There are 2 Spines per A4 POD, one connecting to the IP2 A-Plane, the second one connecting to the IP2 B-Plane. The IP2 backbone has the following architecture:

- IPv4 underlay
- MPLS with ISIS and LDP
- BGP overlay
    - Business VPNs (RFC4364)
    - Internet VRF with full routing table
        - Dual-Stack IPv4/6

From the view of the IP2 backbone the A4 Spine is a LER. With that the Spine must support all GW functions between the Fabric overlay and the IP2 overlay, plus support both underlay architectures.

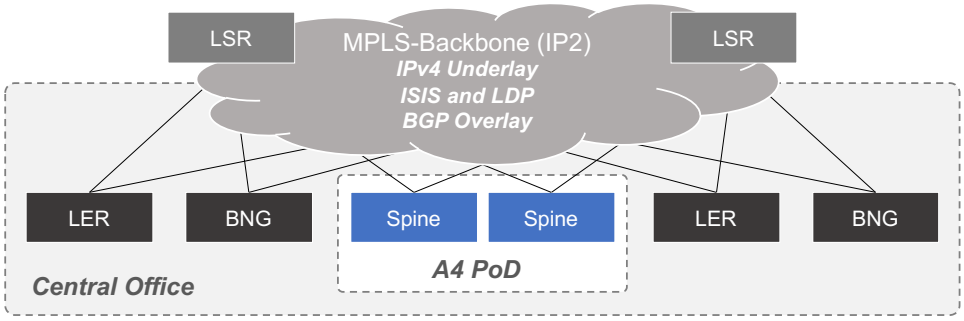

Figure 19 A4 POD backbone connection

3.4.6 Services and Fabric Overlay

Access 4.0 provides the following customer services:

- DT FFTH retail/business
- DT FTTB retail/business
- DT FTTC retail/business
- L2BSA (on FTTH or FTTB or FTTC access)
- IP-BSA (L3 bitstream on FTTH or FTTB or FTTC access)
- RD (Remote Devices = Business Services on direct fiber access)

Based on the Fabric-underlay the solution provides the separation of customer endpoints and access variants by VLANS, L3VPNs, PPPoE tunnels, MPLS Pseudo-Wires (PWE), EVPN and combinations of those networking technologies.

3.4.6.1 Service-Edge (BNG)

Within the A4 project the term Service Edge (SE) is used to denote the subscriber termination function on the Leaf switch. In A4, the SE is a function that does not need to be mapped to a specific single network instance. It may span across multiple physical entities.
To understand what the SE does, we first look at the existing BNG (Broadband Network Gateway). The BNG as defined by the Broadband Forum (BBF) and deployed in our DT network comprises two main functional blocks which are:

- the "subscriber-aware packet handling" (termination of L2 tunnels, conversion to IP, Access lists, shaping per class / max b/w, port speed) which we denote as Service Edge (SE) function
- the "uplink routing function" where the BNG acts as provider or label edge router (LER).

This is depicted in the top row of **Error! Reference source not found.**.

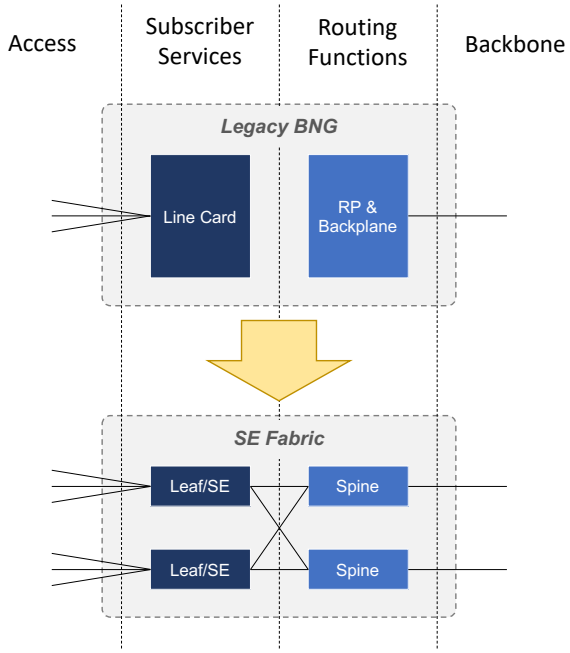

Figure 20 Disaggregated BNG solution

On the SE side (A4-SE), subscribers can be individually identified by a unique identifier that may e.g., be defined as a combination of attached physical port, VLAN IDs and MPLS labels. The state created is denoted as a subscriber circuit (SC). Per-subscriber services such as QoS, accounting and LI will always become applied to the SC.

On the uplink side (A4-RF), the identification of a subscriber is lost. For internet customers, all IP packets are handled in the same routing domain. In principle, a subscriber can still be identified by the IP address / subnet used, but no policies can be applied on such a fine granular level.
De-composition of the BNG into these two main functions SE (Service Edge) and RF (Routing Function) is a design goal of A4.

3.4.6.1.1 Service Edge Core Functions

For retail, wholesale internet access and business VPN services the SE must provide at least the following key functions:

Function	UP impact	CP impact
PPPoE termination	Encap/Decap at line rate, keepalives	PPPoE / PPP state engine for IPv4/6 according to DT specification
Multicast inside PPPoE	Encap in PPP, map to IP core multicast	Handle requests in real-time
L2 and L3 VPN	Support of EVPN and/or MPLS-VPN	Support of EVPN and/or MPLS-VPN
Hierarchical QoS	Multi-stage QoS based on classification rules. The number of stages is depending on the scenario. ~3-5 stages may be needed.	Real-time provisioning of distributed enforcement points and adaption of peak rates in the G.fast case
Lawful Intercept	Mirror selected user traffic	Being compliant to ETSI-based LI systems at DT
ACL (Access Control Lists)	Supporting ACLs per SC with wild cards	Provisioning ACLs and mapping to RaBaPol policies
Accounting	Counting packets based on ACLs	Provisioning ACLs and mapping to RaBaPol policies
Antispoofing	Assuring users only use their assigned IP addresses / prefixes as source addresses	
Tunneling (L2TP)	Acting as LNS, tunneling PPP	Insert attributes (line rate etc.)
DHCP Relay Agent	PADI/PADO/PADT	Line ID
RADIUS Client		

Table 2 Service-Edge core functions

3.4.6.2 Service-Edge Data-Plane

This chapter illustrates the encapsulation and tunnel/tagging stitching data-plane for the different services being provided by an A4 PoD.

3.4.6.2.1 FTTH

The following diagram describes the encapsulation and tunneling design for FTTH:

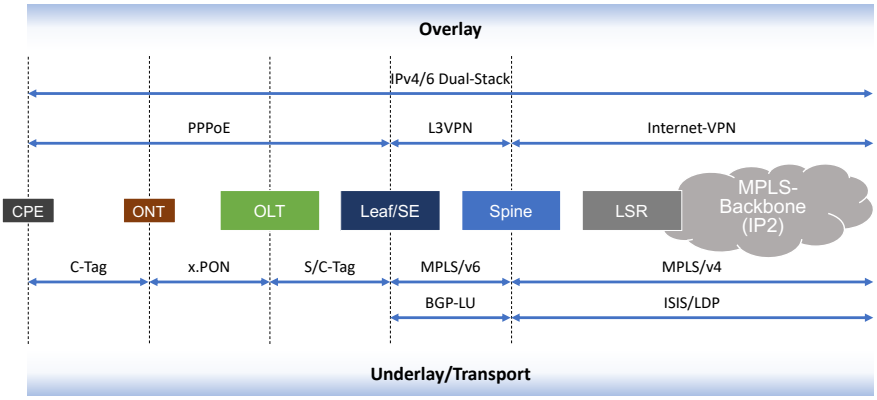

Figure 21 FTTH Data-Plane

The subscriber always gets IPV4/6 dual stack access to the internet. PPPoE is mandatory due to regulations. The PPPoE session is terminated at the Leaf/SE. From there a Fabric internal VPN is used for connectivity to the Spine. The Leaf/SE uses a default route to the internet routing table of the Spine.

For details on the FTTH service orchestration please refer to chapter: FTTH Service Orchestration

3.4.6.2.2 FTTB

The following diagram describes the encapsulation and tunneling design for FTTB:

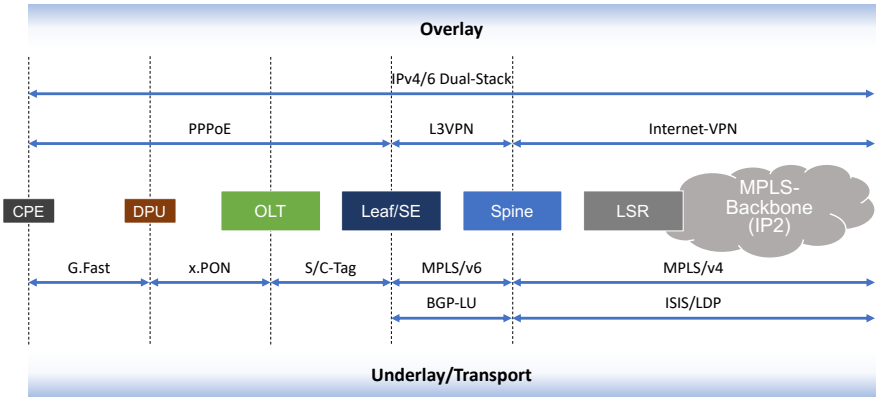

Figure 22 FTTB Data-Plane

The FTTB data-plane is equal to the FTTH data-plane from underlay and overlay IP-Routing perspective. The only difference is the usage in the underlay/transport between the DPU and the actual CPE.

For details on the FTTB service orchestration please refer to chapter: FTTB Service Orchestration

3.4.6.2.3 FTTC

The following diagram describes the encapsulation and tunneling design for FTTC:

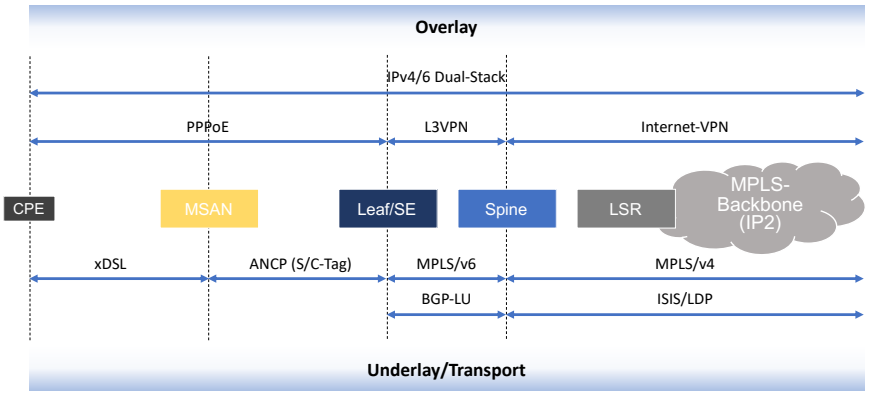

Figure 23 FTTC Data-Plane

The internet service is modelled similarly as in the case of FTTH/B. In case of FTTH, the voice and internet service runs on the same connection. In the case of FTTC, the voice is separated from internet/broadband at the MSAN. The internet is delivered using xDSL towards the consumer and the customer traffic is double tagged between MSAN and BNG. ANCP protocol is used between MSAN and BNG for events related to subscribers. The ANCP protocol is used to convey subscriber xDSL port events, the VLANs and LineID used by the subscriber for data traffic, etc. Both ANCP and the subscriber traffic run on the same physical connection.

NOTE: The MSAN management and orchestration is not part of A4 except for transport of the MSAN Inband Management channel between MSAN and Vendor EMS.

For details on the FTTC service orchestration please refer to chapter: FTTC Service Orchestration

More details on Telekom-Wiki: https://wiki.telekom.de/display/ACC4/FTTC+Background

3.4.6.2.4 Whole-Sale (L2BSA)

Layer 2 Bitstream Access (L2BSA) refers to a scenario in which a service provider makes his access infrastructure available to other service providers. Traffic from access infrastructure is handed over to the wholesale service provider over Ethernet. In Germany, this service is mandated by the Federal Network Agency (German: Bundesnetzagentur or BNetzA) which is the regulatory office for electricity, gas, telecommunications, post, and railway markets.

The following diagram describes the encapsulation and tunneling design for Whole-Sale and L2BSA:

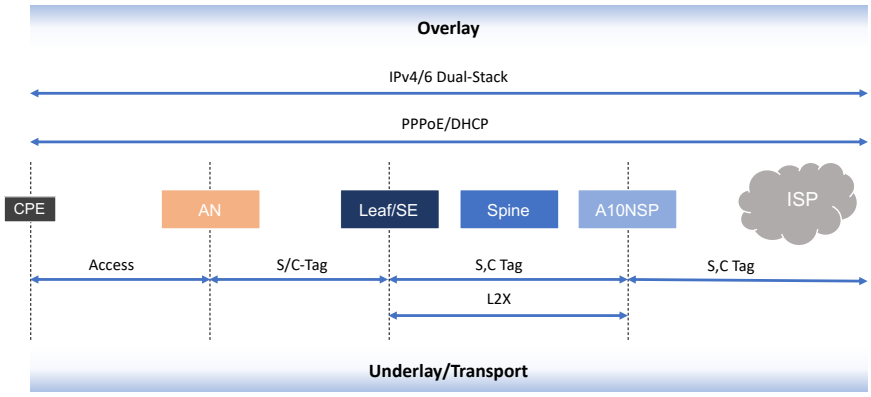

Figure 24 Whole-Sale and L2BSA Data-Plane

Access can be either FTTH/B or FTTC. The subscriber session (PPPoE/DHCP) is terminated outside of the DT network, and therewith outside of the A4 infrastructure. The A4 PoD forwards the sessions as L2 packets across the Fabric to the A10NSP switch. The A10NSP switch is part of the A4 PoD.

For details on the Whole-Sale/L2BSA service orchestration please refer to chapter: Access 4.0 Service Orchestration

More details on Telekom-Wiki: https://wiki.telekom.de/display/ACC4/L2BSA+Services

3.4.6.2.5 Whole-Sale (L3BSA)

TBD - Planned for next version with MS6

3.4.6.2.6 Whole-Buy (N:1)

TBD - Planned for next version with MS6

3.4.6.2.7 RD

For details on RD data-plane please refer to chapter: Deutsche Telekom Business Services RD (Remote-Device)

3.4.6.2.8 MTU Considerations

TBD - Planned for next version with MS6

3.4.7 Fabric Routing

As mentioned in previous chapters of this document the Fabric is implemented as a 3-stage CLOS Fabric consisting of Spine and Leaf switches based on draft-ietf-spring-segment-routing-msdc-11.

Between the Spine and Leaf switches eBGP sessions over IPv6 link local addresses are established. The eBGP session will support segment routing extensions according to draft-ietf-idr-bgp-prefix-sid-27. In order to exchange the underlay topology information, the switches will advertise their IPv6 loopback addresses as a unicast (AFI/SAFI=2/1) as well as label unicast route (AFI/SAFI=2/4). The 2 Spines reside in the same AS, each Leaf is a dedicated AS.

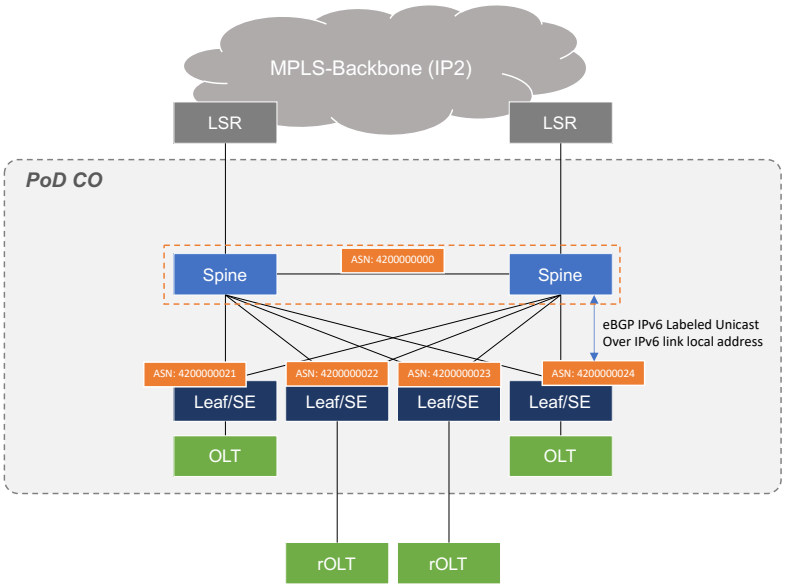

Figure 25 Fabric BGP and IPv6 Routing

In general, RFC2545 (Use of BGP-4 Multiprotocol Extensions for IPv6 Inter-Domain Routing) defines that value of the length of Next Hop Network Address field on a MP_REACH_NLRI attribute shall be set to 16, when only a global address is present, or 32 if a link-local address is also included in the Next Hop field. However, the links between the switches do not have any global addresses. Therefore, the LINK-LOCAL-ONLY-NEXT-HOP capability as defined in draft-kumar-idr-link-local-nexthop-02 must be supported on the BGP session. As a result, the value of the length of Next Hop Network Address field on a MP_REACH_NLRI attribute will be set to 16 and network address will be set to the link-local IPv6 address of the next hop which is the peer address.

On the control plane, labeled BGP implementation across IPv6 control plane is fully implemented. In order to secure the BGP sessions, AES authentication on the TCP layer has been added. There is support for one transmit key and two receive keys to allow key rollover.

The Fabric implements three Layer-3 VPNs and with RD it will also support VPWS-L2 services:

- Instance IP2 which contains all the routing information for connectivity to the global internet.
- Instance inband_mgmt which provides connectivity between servers connected to the Spine switches and the Fabric switches themselves for sake of configuration and management.
- Instance LI-BOX which provides connectivity between the Leaf switches and the LI-BOX on the inside (pre-IPSEc encryption).

For global internet connectivity, the Spines switches are connected to IP2 backbone network within instance IP2 and receive 3x full table via iBGP from the backbone route reflectors. The Leaf switches on the other

hand provide connectivity to the subscribers and thus should have a minimal routing table from IP2 perspective.

The Leaf switches have a single peer-group (BGP Peering Groups) within the default instance reflecting the eBGP sessions towards the Spine switches running different address families. Route policies are attached to this peer-group. The Spine switches have two peer-groups within the instance default: the LEAF peer-group which handles the BGP session to all the Leaf switches and the SPINE peer-group which handles the session to the neighboring Spine switch.

The default instance of the switches provides the basic connectivity between the switches and therefore only exchanges IPv6 labeled-unicast routes. On the Leaf switches, export policy is only used to mark the IPv6 address of the lo-0/0/0 interface with the internal community 100:1.

In order to prevent suboptimal routing, e.g. Leaf1->Spine1->Spine2, there is a BGP export policy on each Spine switch which advertises its own loopback IPv6 address with a better metric towards the Leaf switches (MED=50) than the loopback IPv6 addresses of all other Fabric switches (which have MED=100).

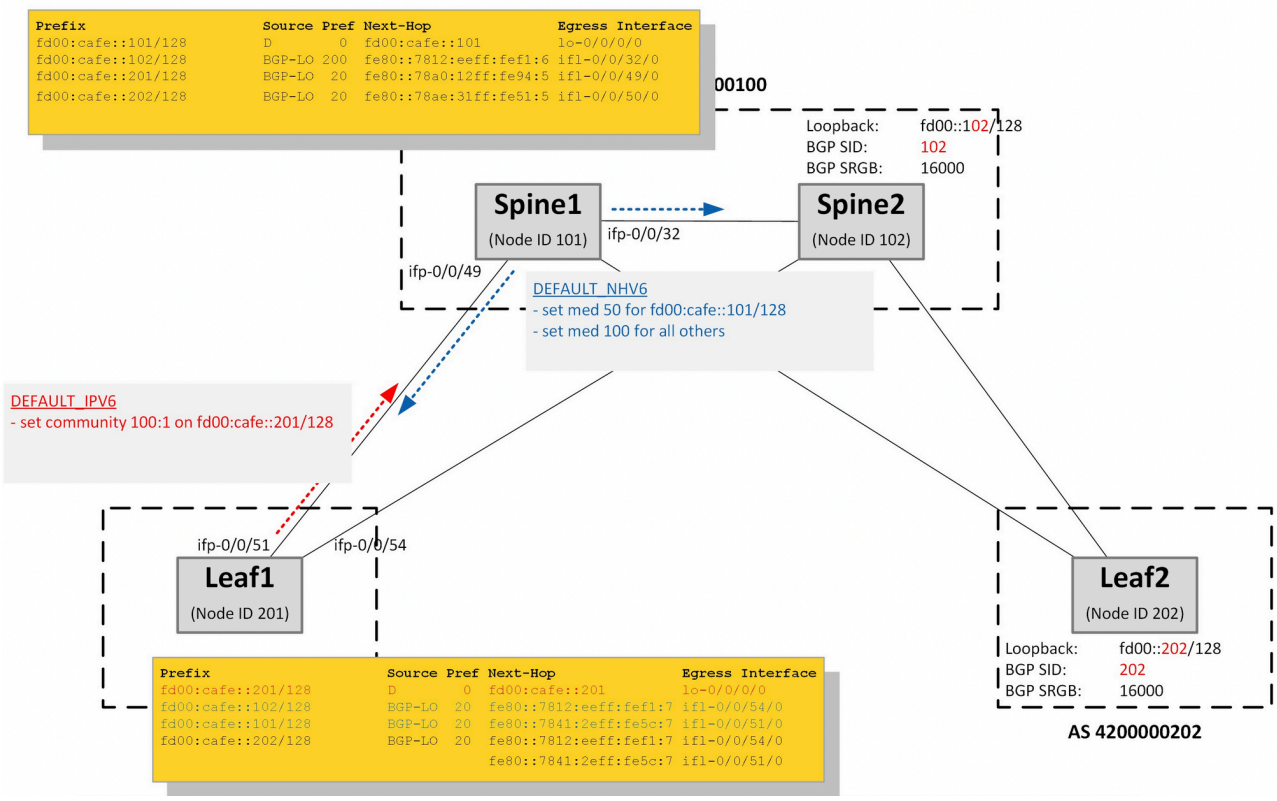

Figure 26 Default Instance: Routing for IPv6 Labeled Unicast

The routes for the IP2 instance are exchange as IPv4 vpn-unicast routes within the Fabric and as IPv4 unicast routes towards the IP2 route reflectors. The Leaf switches will advertise two kind of prefixes within the IP2 instance towards the Spines:

- loopback IPv4 address (lo-0/0/1) which is tagged with the “external” community 100:2
- IPv4 address pool fragments (allocated by PAO) for PPP subscriber address assignment which are tagged with the “internal” community 100:1

The Spine switches need to advertise the IPv4 loopback addresses, and IP2 subscriber pool fragments received from any Leaf switch to all other Leaf switches with BGP nexthop unchanged in order to avoid additional route lookups on the Spine switches.

In addition, the IPv4 loopback address of the Spine switch itself needs to be advertised as well as loopback addresses of services (e.g. PAO, LI-BOX, etc.) that are directly connected to the Spine switches. These routes will be tagged with "Spine local" community 100:3. Note, there is a difference between internal routes send by Leafs (community 100:1) and internal routes send by Spines (community 100:3) for two reasons: (1) on the Spine switch, next hop of local routes need to be rewritten, while next hop of Leaf routes remain unchained and (2) Leaf switches should prevent from exporting Spines routes learned from one Spine back to another Spine.

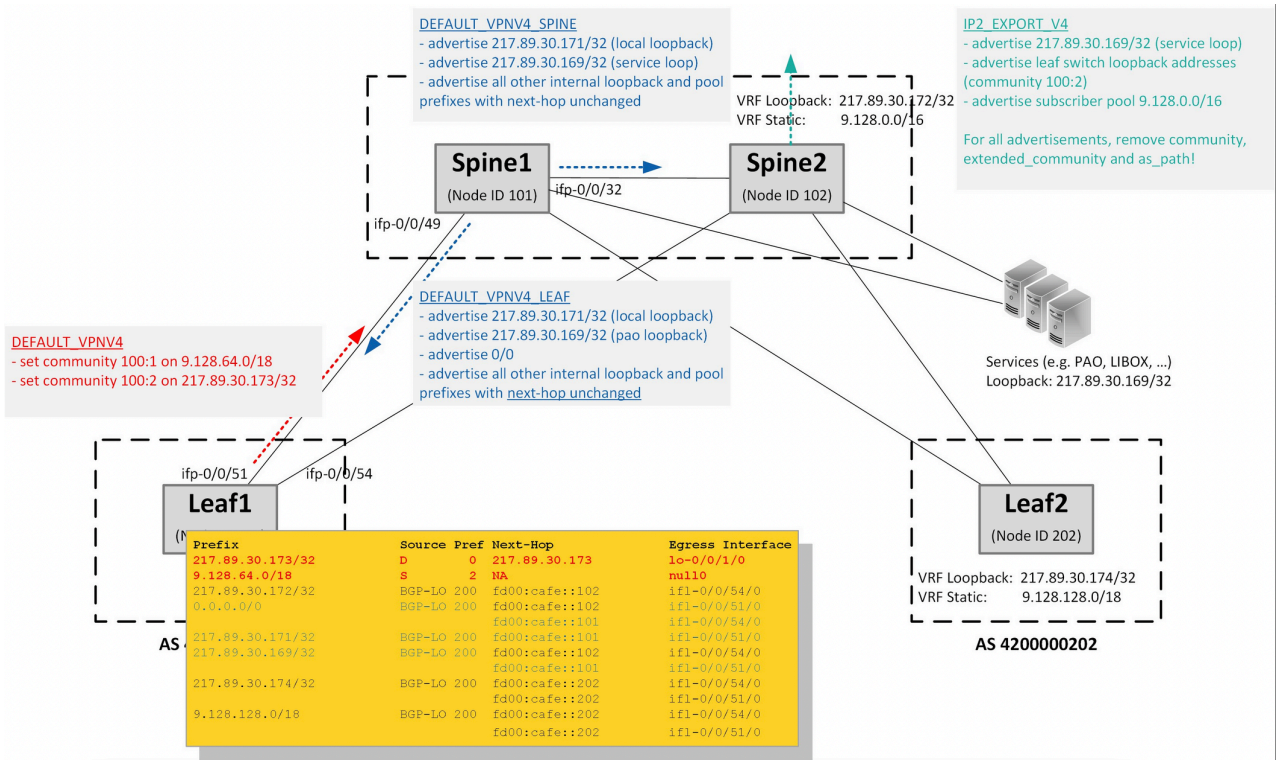

Figure 27 Default Instance: Routing for VPNv4

Finally, a default route is advertised to the Leaf switches for all destinations that are reachable via the IP2 backbone. Note, there will be no backbone routes propagated to the Leaf switches.

For more details, please refer to: https://wiki.telekom.de/display/ACC4/Routing

3.4.8 Service-Edge QoS

3.4.8.1 FTTX QoS

Deutsche Telekom provides Single, Double and Triple Play services to their subscribers. These services consist of the following traffic classes:

- Internet Traffic – Best Effort Traffic
- Voice DT
- Voice 1&1
- IPTV DT
- IPTV OTT

Subscriber services and the relating COS and QOS parameters are signaled to the BNG or Service-Edge via Rabapol (Radius Based Policing). Find here the full list of RabaPol policies or services that are enabling QoS towards the subscriber:

#	RaBaPol	Descriptive Name	Rationale
1	SRL	Session Rate Limit	CoS shaping for all the subscriber services in the egress and the hierarchical rate-limit policing in the ingress. SRL covers any traffic, even unmanaged traffic (UMT)
2	VOICE	Voice Service	Classify the Voice Data and Control traffic according to the received parameters
3	REDIRECT	HTTP Redirect Service	Limit access for PPP sessions with a faulty or incomplete user profile (e.g. missing or malformed Line-ID)
4	HSF	Hot Spot Fon (aka, WLAN to Go)	Classify and limits the rate of the transparent L2TP tunnel traffic from/to the CPE
5	UMT	Unmanaged Traffic	Classify traffic either exceeding the time/volume quotas or not marked by any other RaBaPol Filter
6	IPTV	IPTV Service	Enable IGMPv3 (IPv4 Multicast) and MLDv2 (IPv6) on the PPP session, assign the SSM map as parameter in the service activate, apply a filter on multicast destination address for service accounting
7	WIA_QOS	Virtual Provider L2 Classifier	Classify, rewrite (and optionally policy) layer 2 services of a virtual Internet provider
8	L2TP_QOS	L2TP L2 Classifier	Classify and rewrite traffic based on layer 2 or tunnel header information

Figure 28 RaBaPoL QoS Attributes/Policies

The “Session Rate Limit” refers to the maximum accumulated PPPoE (Session) bandwidth up and downstream to the subscriber. The entire traffic must be shaped downstream to this attribute at the Service-Edge or Leaf. Then i.e. in the case of PON, the Service-Edge needs also to shape traffic to the bandwidth the PON tree overall is providing. Then some services such as IPTV are shaped. Overall, there may be 5 levels of shaping at the Service-Edge to provide internet access services for the 3 access technologies FTTH, FTTC, or FTTB:

- Level-1 Physical Interface Shaper
- Level-2 PON TREE
  - Each PON tree is a TDM based shared medium with typically ~2.5 GBit/s (GPON) shared by up to 32 consumers (ONT or DPU).
- Level-3 DPU
  - In case of FTTB there is a single DPU with multiple consumers via G.Fast DSL connected which requires an additional hierarchy. This level is not needed for FTTH or FTTC.
- Level-4 PPPoE Session
  - The Access Node Port (ANP) or outer VLAN level describes a single customer line. This might be an ONT in case of FTTH or DSL interface behind a DPU in case of FTTB. This level can be also represented on PPPoE sessions as long as just one session is permitted per VLAN.
- Level-5 QUEUE

- The Queue level shaper is required to limit the class-of-service bandwidth like Voice or IPTV traffic limit.

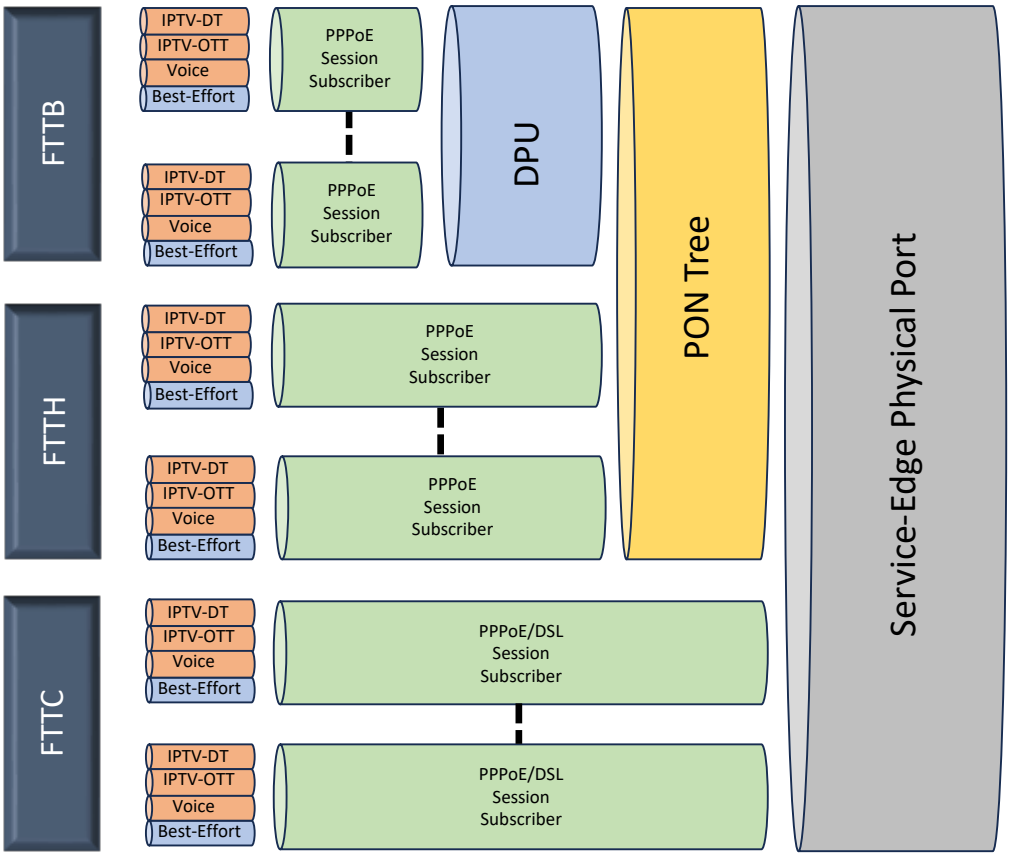

Figure 29 H-QoS Levels for FTTB/H and FTTC

PTA: PPP Terminated Aggregation

SRL: SessionRateLimit

VSA: Vendor Specific Attribute (VSA)

The ACCESS_BANDWIDTH_UP and ACCESS_BANDWIDTH_DOWN in the below tables are access technology agnostic (independent) and they are at Bit/s Layer 3.

Before using access bandwidths received from the access technology (Voltha, ANCP) in calculations it needs to be normalized with the following correction factors. The correction factors are given by VOLTHA/ANCP implementors.

3.4.9 PoD Management Networking

Each POD has two Bottom-Of-Rack switches for the device management. The BORs provide Out-Of-Band (OOB) 1GE Ethernet access to the compute and switch platforms. The BORs are connected to the so called DCN, the network management network for Deutsche Telekom. The DCN is a self-contained and separate IP-MPLS network, with the purpose of managing the entire Deutsche Telekom critical infrastructure.

The POD might be split into 2 fire sections inside of the central offices. Compute has usually minimum one ILO or BMC port for the bootstrapping of the hosts, plus 4 x 1 GE OOB management network ports. The OOB ports are distributed to the 2 BORs, so that in case of a failure (BOR/Fire section) there are still 2 ports active. Switches do usually have only one management port. These ports are distributed over the BORs, so that in case of a failure, at least a critical quantity of switches is still reachable from the DCN. Find here a schematical illustration of the OBB management design. More information can be found in chapter 7.2.

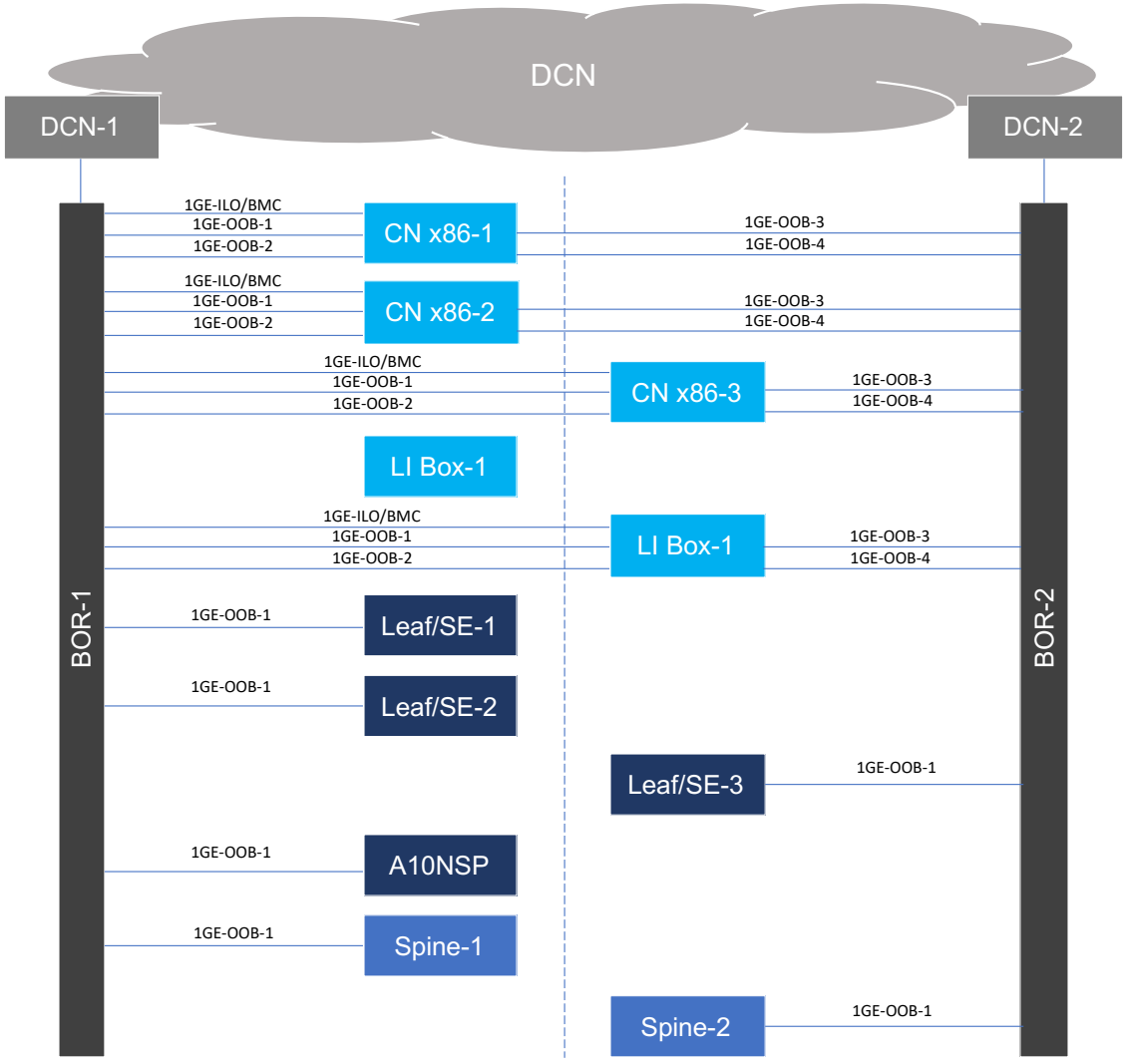

Figure 30 A4 POD Out-Of-Band Management

3.5 A4 Architectural Model and Interfaces

This section describes the interface between the OSS-IT and the Central EMS and the Information model defining the Abstraction provided by the Logical versus Physical view of the hardware components.

3.5.1 Access 4.0 Architectural Model, Logical Resource View

The A4 Functional/Logical view is represented by the Functions that are planned by the DT Back Office planning system, external to the NEMO/A4-EMS (NEMS) and A4-POD and these external components interface with the Centralized NEMS to send the planning object classes that represent the inventory:

- NetworkElementGroup (NEG == A4-POD)
- NetworkElement (NE)
- NetworkElementPort (NEP)
- NetworkElementLink (NEL)

See Reference: A4-M-SEQ-001: Provide results of 'planning' to POD which describes how the planned Inventory Resources, are sent to the NEMS system. These 'Planned Inventory Resources' are augmented by NEMS to add 'Operational Data' to the 'Planned Data' and are then Synchronized to the A4 POD, however the vision is to automate the addition of the Operational Data elements either 'autogenerated' or received from partner systems (e.g. IP-ADB). Operational data management is described here: Operational Data management in A4 as Sub-sequence 'SEQ_Plan_2a'.

The NEMS provides an auto generation function for some aspects of the Resource Operational Data in EMS, while some Operational data is required to be entered by OPS Personnel via the NEMS GUI. NEMS internal processing of the Logical planning Resources augmented with operational data is described in 01 A4-M-SEQ-001: Provide results of 'planning' to POD.

The Logical Resources are 'owned' by the OSS-IT, but there is another class of Resourceop that is owned by the A4 POD itself called the 'Termination Point'. This class of Resource are considered as "Bookable Resources" and need to be exported back to the external OSS-IT System. Termination Point Resources are auto generated by the A4 POD based on the equipment detected and the Termination Points supported by that equipment. The A4 POD will report the Termination Points to the NEMS and the NEMS will report the available Termination Points to the OSS-IT. The Master Sequence: A4-M-SEQ_0006 describes this process and this is described in Figure 31.

The Termination Points are used for Service attachment points required for the fulfilment and the OSS-IT associates Network Service Profiles (NSPs) to the exported Termination Points or "Bookable Resources".

The Network Service Profile (NSP) represents access infra resource configuration profile for service fullfilment or a end to end customer service activation (either FTTH or FTTB with a PFS derived service profile or L2BSA Service with an A10NSP cross-connected NSP termination see Section 'Access 4.0 Service Orchestration') profile. NSP contains (or by Reference) all the data needed by the A4 POD to autonomously activate partial circuit or end to end customer service. The Master Sequence: A4-M-SEQ_007 describes this process and this is described in Figure 32 below.

Logical Resources with respect to FTTC services are described in a following chapter in this section. Legacy MSAN support in the Access 4.0 Architecture is treated as a special case because MSANs do not exist as a distinct NetworkElement and are handled as an 'off network device' transparently managed by Legacy Vendor EMS. Additionally, FTTC subscriber service profiles do not exist as a OSS-IT managed Resource.

Logical Resources are themselves defined by the A4 Information model, however the standard TMF639 Resource classes used to communicate the Logical Resources between the OSS-IT and NEMS do not match the Information Model directly and so the A4 TMF639 API augments the TMF information model by

using the TMF 'Characteristics' attribute to add the A4 Information Model attributes needed for each Resource above and beyond the standard model.
The mapping between the A4 Information Model artifacts and those of the TMF 639 (version 1) are defined here: A4 Information Model to TMF639 Mapping .

All the Logical/Functional Resources are pre-planned top down by the planning systems and sent to EMS and the A4 POD before any related physical hardware is installed.

This process is described in the Telekom Wiki : NetworkElement Planning, Installation and Bringup - Access 4.0 - Telekom Wiki

A4 POD Stores the NEG Resources in its Local DB, Notifies the POD of the DB Changes and the POD Controllers (see **Error! Reference source not found.**) under direction of PAO/SDN, generate the Physical Resource configuration based on the information model data, EMS Operational Data and Policy Profiles/Templates.

These processes are described in Figure 31.

An exception to this behaviour relates to the FTTB DPU NE and its NEL connection to the OLT PON tree and the NEPs allocated to the G.fast ports of the DPU.

The model for DPU hardware extension to A4 POD requires that the PON tree connection a given DPU is connected to is an installation time decision. When the installation technician enters the DPU serial number and the PON tree it connects to into the specific DPU Installation OSS-IT User Interface the OSS-IT will generate/update the related DPU NE/NEP/NEL Resources and send them to the NEMO/A4-EMS where additional Operational Data will be added and then the data is sync'd to the A4 POD where the object creation/change notifications will cause the DPU Controller to generate the required DPU configuration. Note that the enrichment of operational data is an asynchronous process executed in parallel with the inventory synchronization, meaning the synchronization process will be triggered twice:

1. based on the incoming data from OSS-IT
2. after the operational data calculation

The DPU-C then follows the process described in Figure 32 to generate the required Termination Point information and update the POD CD which then causes a synchronisation to occur to A4-EMS and then to the OSS-IT. This results in the FTTB NSPs to be associated with the DPU Termination Points and then updated to NEMO/A4-EMS where additional Operational data is added and then all sync'd to the A4-POD Local Manager.

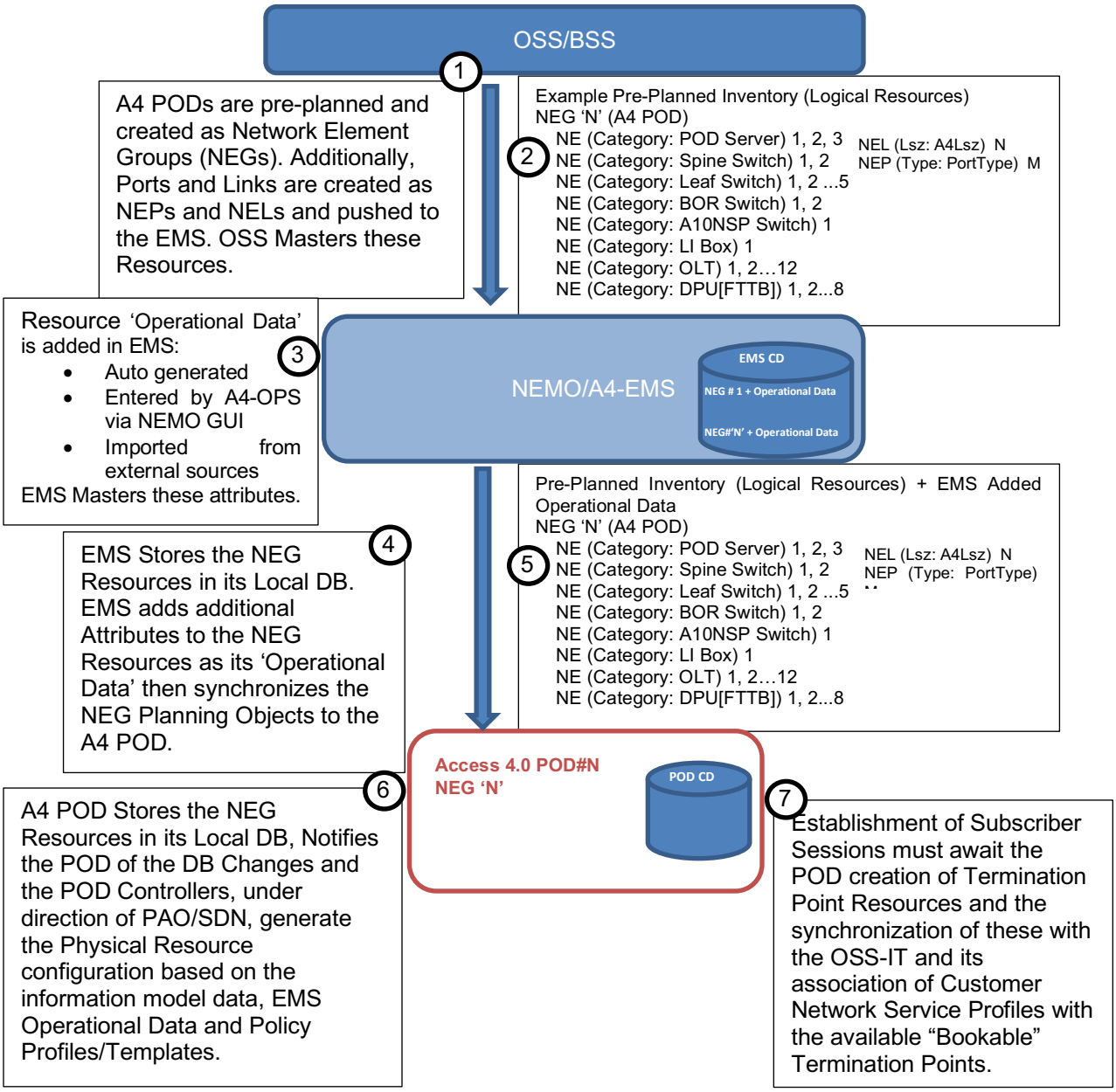

Figure 31 A4 POD Logical Resource creation via OSS-IT Pre-planning Deployed by EMS

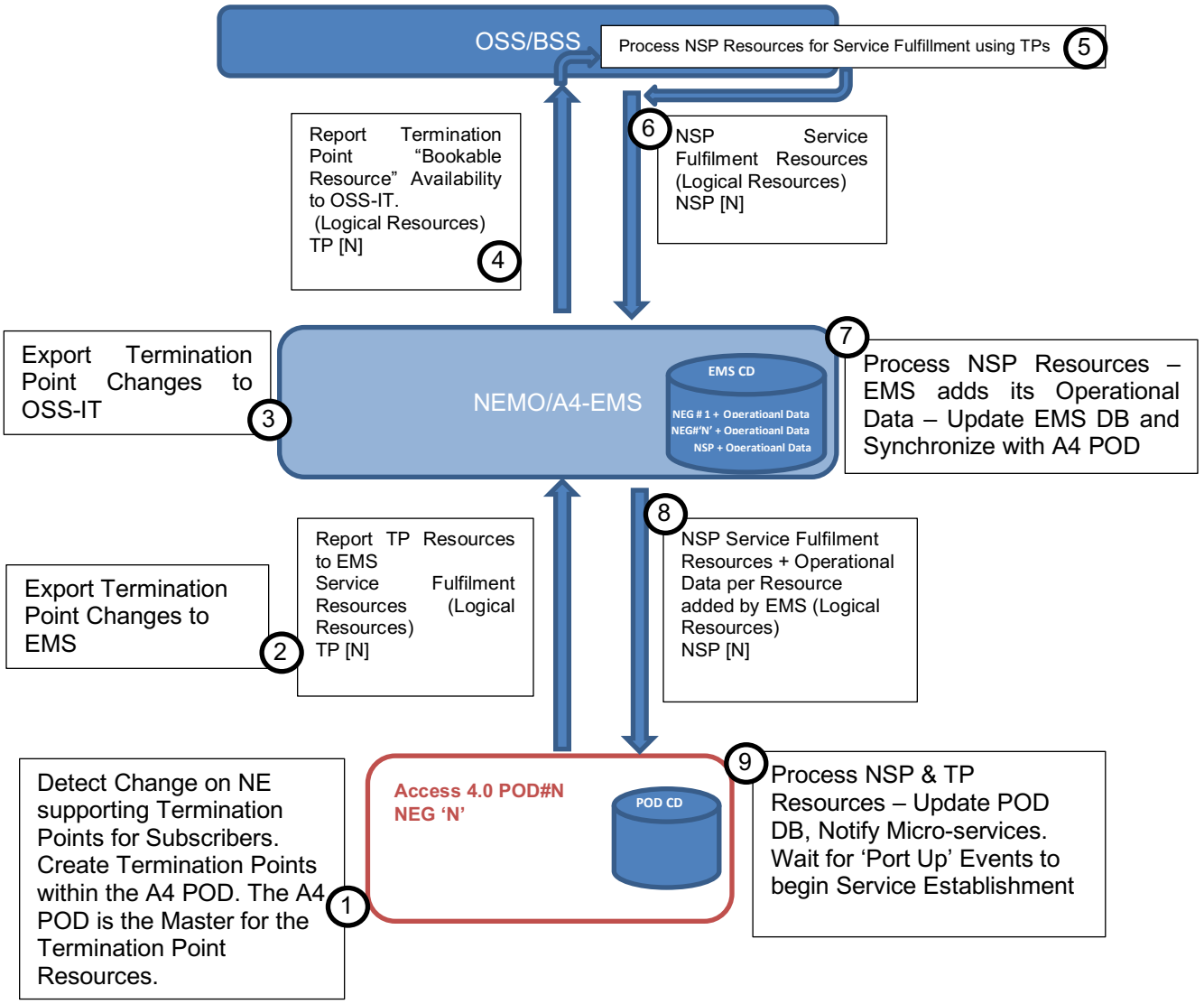

Figure 32 A4 POD generates Termination Point Resources for OSS-IT for Customer Service Fulfillment

Note that the scenario for initial startup of A4-EMS and A4 POD synchronization is described as part of the POD Bootstrapper scenario in Section 'Error! Reference source not found.'.

3.5.2 Access 4.0 Architectural Model, Physical Resource View

The Physical Resources view is described in Telekom Wiki : A4 Physical Resource Concept.

Master Sequence A4-M-SEQ-005: POD HW Inventory Changed describes generation of hardware inventory information within an A4 POD and its transmission to the consumers (EMS, OSS-IT).

In general, the Physical Resources within a POD represent the Hardware Resources present in the A4 POD and this information model is “Mastered” by the POD itself.
The POD Device Controllers are responsible for detecting state from the hardware network elements, managing the auto-installation and authentication of detected hardware including ensuring the software artifacts are up to date.

Master Sequence in Telekom Wiki: A4-M-SEQ-003: POD HW Extension describes the sequence of events around a POD hardware change.

Master Sequence in Telekom Wiki: A4-M-SEQ-003D: POD DPU HW Extension describes the sequence of events around a DPU hardware change.

3.5.3 Access 4.0 Architecture Integration and Migration of Legacy MSAN

DT Legacy MSANs and their subscribers will be migrated to the new Access 4.0 Architecture, the existing MSAN implementation in the DT network with a BNG connection, and proposed migration strategies to A4, is described here: MSAN Migration@A4.

The high level design for FTTC in A4 is described here in the Telekom Wiki: FTTC in A4 - High Level Design
.

Note that for MSAN migration to the A4 POD the MSAN is not viewed as a Logical Resource managed/owned by Digi-OSS nor as a Physical Resource to be discovered/owned by the A4 POD. The MSAN is not directly managed as a Resource by NEMS or the A4 POD instead the A4 POD is configured to tunnel through management connections directly to a Vendor specific EMS system which provides device and service configuration for the MSAN and its Subscribers directly.

In order to facilitate the integration of Legacy MSANs into the A4 POD the MSANs will remain managed by the existing legacy systems.
Interaction of the Legacy Systems with the A4 POD via Digi-OSS is described here in the Telekom Wiki: Sequence Diagram showing interaction of Legacy Systems and A4 POD via Digi-OSS.

For every MSAN connected/planned for an A4 POD, specifically connected/planned for a NetworkElement Resource (Leaf Switch) Port, a ‘NetworkElementPort' Resource with an attribute for ‘offNetworkLink’ set in the NepOperationalData and a single ended ‘NetworkElementLink’ Resource attached to the NEP are created.

3.5.3.1 FTTC Specific Architecture for Bookable Resources

For MSANs, a clearing of the LoS for the MSAN Connected Port on the Leaf Switch (MSAN physical Uplink) represents the FSoL and detection of the MSAN itself. Service Orchestration follows the provisioning and detection of an MSAN connection

The A4 NEMS is provisioned via Digi-OSS for an MSAN connection to an A4 POD Leaf Switch by the creation of a NEP with an 'OffNetworkLink' with endpoints of the referenced NEL referencing an MSAN Termination. The NEMS will synchronize these Logical Resources to the A4 POD so it can recognise the MSAN connection to the Leaf Switch and allow the A4 POD to generate an MSAN Link 'Termination Point'. This MSAN Termination Point is synchronised to NEMS and Digi-OSS allowing an association of a NspMsanUpink NSP with the Termination Point. This MSAN TP associated NSP contains all required configuration information for the A4 POD to establish Management and Control connectivity to the MSAN.C

There is a single Termination Point (Bookable Resource) created per MSAN connection point and that is associated with a single NSP for the MSAN Uplink which provides configuration parameters for system operation.

3.5.4 Architectural Interface Definitions

The architecture defines the External Interfaces between Components shown in **Error! Reference source not found.**. A reference to a Telekom-Wiki Wiki Table defining the Versioned interfaces is here:. Architectural Component Versioned Interface Definitions.

3.6 A4 POD Components and Functional Decomposition

The previous sections described the interface between the OSS-IT and the Central EMS and the Information model defining the Abstraction provided by the Logical versus Physical view of the hardware components.

This section focuses on the POD-local software components providing local device and service management, network programming (to e.g. enable session steering) and managing and controlling network components such as switches and OLTs.

Error! Reference source not found. shows the main POD-local Control and Management Plane Components of Access 4.0 and denotes the Interfaces between the Components.

All subscriber services are orchestrated by the PAO interfacing to each of its Device Controllers as needed for managing the end-to-end set-up and tear down including interfacing to the PFS Radius controller which is external to the A4 POD and reachable through the IP Core network.

The PAO (POD Access Orchestrator) Local Controllers are responsible for management and configuration of the A4 POD hardware components based on the Operational Data provided by EMS and the A4 POD Local Management and the associated Logical Resource objects and their Attributes provided by OSS-IT. Available Termination Points are also calculated by the Local Controllers and exported via the EMS interface to the OSS-IT.

The OSS-IT provides the association between the reported A4 POD Termination Points and the customer Network Service Profiles.

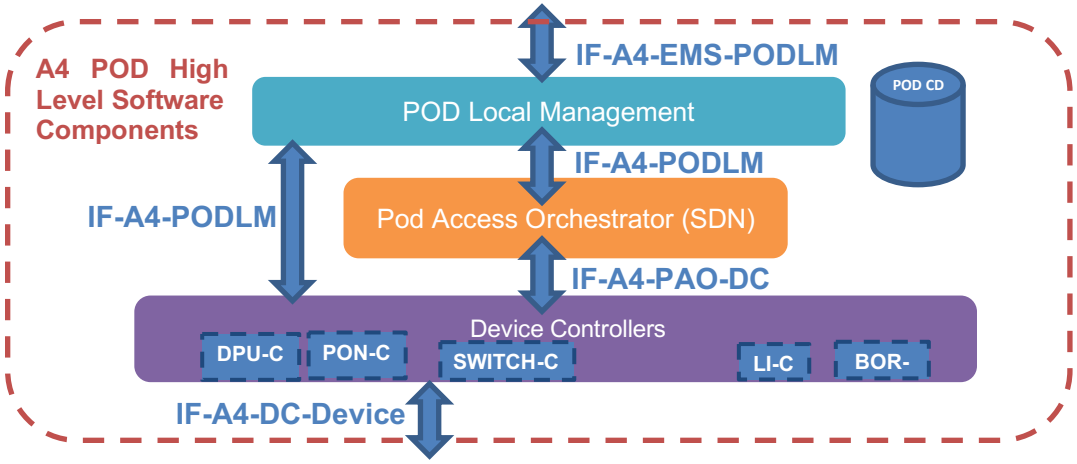

Figure 33 A4 POD Main Software Components

The depicted local management and PAO components run on the local IT infrastructure provided by the A4 Edge Cloud platform. This infrastructure includes physical servers, host OS, virtualization layer as well as infrastructure services (IT modules used by multiple applications) and is described in a separate section.

Both, local management and orchestration components reside as modules on top of this infrastructure. Modules are wherever possible fully stateless which implies they get invoked, fetch additional data from the local database, execute and action and put the result back into the database. The services may trigger a follow up activity by another module by sending a request to the internal messaging framework.

3.6.1 A4 Components: POD Local Management (POD_LM)

The A4 POD Local Management Component is responsible for providing the Northbound Interface of the A4 POD to the Centralized EMS Component (using the DT T-DCN network) exchanging Logical, Physical and Operational Inventory information and state information. The central EMS provides the A4 POD Local Manager with what resources like Network Element functions, Network Element Ports and Links are allocated for the POD-Servers, OLTs, DPUs, Switches of various types (Leaf, Spine, A10NSP) and the LI server. The Local Manager provides current state information and Event/Alarm messages. The A4 POD also calculates the "Bookable Resources" it can provide, e.g., Termination Points which can be used to assign subscriber Profiles NSPs to initiate Subscriber Service sessions. The A4 POD Local Manager provides the Resources available to the POD control plane and handles the reporting of the discovered physical equipment and topology, and the Object create/update notifications which drive the configuration generation of the Controllers for their managed network elements. In addition, the local management plane also hosts components such as Prometheus Webhook to transform Prometheus alerts into A4 Alarms and the GELF API to transform the Events from Leaf/Spine/A10NSP switches into A4 business events.

POD local management further provides the execution of the centrally managed software life cycle of the POD components.

3.6.2 A4 Components: POD Access Orchestrator (POD_PAO)

The PAO SDN Orchestrator is a microservice-based SDN controller providing the following capabilities:

- Subscriber Management:
  - 'Port-Up'/'Port Down' Event handling.
  - Determine subscriber type/product/access technology.
  - Manage Two Stage Session establishment:
    1. Session Steering; 'Empty' Session context establishment:
       I. select Leaf/Service Edge,
       II. install Fabric Path via SWITCH-C controller
       III. install Access Node Flows via Access Controller (PON-C, DPU-C)
       IV. install Service Edge service endpoint via SE-C controller
    2. Subscriber Session Establishment:
       I. Service Edge requests subscriber Policy and Authorization
       II. Query Subscriber NSP Operational Data for Production Scheme
           a. POLICY_CONTROLLER_LOOKUP: POD external Source e.g PFS Radius Server(Always the case for FTTC – even L2BSA)
           b. POD_INTERNAL: POD internal Lookup
       III. Authenticate Subscriber
       IV. Retrieve Subscriber Policy and install on SE via SE-C controller
- Overall topology reporting – connectivity of NEP, NEL
- VLAN label calculation and assignment for empty sessions and L2BSA
- QoS profile calculation, monitoring and control
- Traffic accounting
- Radius proxy for the whole POD
- Subscriber IP assignment and management
- Interface to the specific Device Controllers to Manage POD equipment and Services
- FTTB DPU Device Empty Session management where Service Flow and ANP-Tag allocation is made after DPU FSoL to minimize the updates to the DPU/ONU configuration as this causes a loss

of connection for all G.fast Subscribers on that DPU. This happens independent of the ANCP reported “PortUp” status report and the DPU/OLT PON Service Flow configuration is maintained in place, only the SE edge provisioning comes and goes with the ANCP PortUp/PortDown transitions.

- FTTC MSAN Subscriber ‘Port up’ is received via the ANCP MSAN uplink specific session. The A4 POD terminates all the per MSAN Uplink ANCP sessions and will create an FTTC Subscriber Empty Session based on the Subscriber attributes received in the ANCP Port Up message. PAO knows that all FTTC MSAN sessions are POLICY_CONTROLLER_LOOKUP: POD external Source i.e. PFS Radius Server in the DT case. As for FTTH/FTTB subscribers the session is authenticated, and the policy is retrieved from the PFS and programmed into the SE.

Note: Path calculation is currently not provided by PAO. PAO instructs the Fabric to cross connect end points over the best path.

3.6.2.1 PAO System Architecture

The POD Access Orchestrator is an orchestrator as well as an SDN controller spanning across multiple network domains. It does contain device abstraction via the device controller modules, a topology and flow management module to steer sessions as well as an intent-based north bound API. Those are the three key characteristics of an SDN controller. As the PAO spans across access, aggregation and IP edge, it is a multi-domain controller which basically makes it also an “orchestrator”. In contrast to other SDN controllers like ONOS or OpenDaylight, the PAO does not program every network element on the data path. While it could in principle do that, the Access 4.0 team had decided to leave the actual chosen path up to the edge devices in the Fabric (ingress and egress points e.g., Leaf switch and A10 NSP for Layer 2 wholesale). This method is explained in the Fabric section of this document.

In addition, the PAO includes a set of applications that make use of its unique position in the network. LI modules steer the lawful interception and the radius proxy allows to communicate with the radius platform (PFS) while being able to abstract specific proprietary radius VSAs as well as from PFS (“Rabapol” services) and southbound to the SE NOS.

On a high level, PAO is positioned in the DT A4 system as follows:

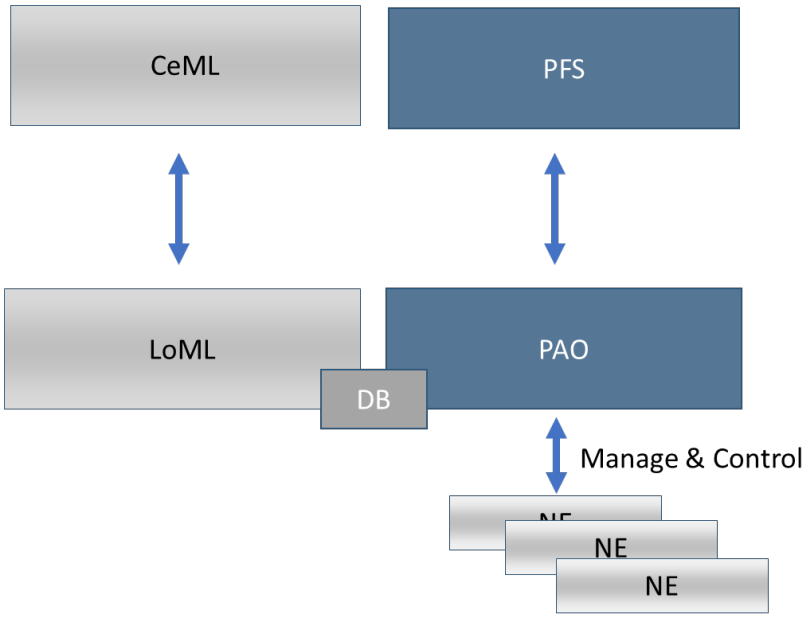

Figure 34 PAO, 10,000ft level view

PAO is co-located with local EMS (LoML), uses a common set of databases, and interacts directly with the PFS (via the PFS proxy layer).

Main tasks of PAO are:

- Manage and control NE
- Manage and control Sessions
- Provide LI
- Provide IP Address Management
- Provide accounting

As can be seen, PAO is a crucial part of the A4 system and needs to be extremely scalable and reliable. In order to build such a reliable and scalable system, the A4 team had decided to go along with a service-based architecture and an event-driven approach.

In a service-based architecture, functions are usually de-composed into small modules that are usually called microservices. These microservice stay on a rather low level of complexity as they perform ideally atomic / single tasks. This makes them easier to build and debug. Microservices are invoked by events that appear on a common messaging system that the microservices subscribe to. If such an event happens, one of the running microservices will pick it up, perform its action and send the result back to the messaging system and optionally store parts of the result or a system state change in the central database (PAO-DB, part of the POD-CD). When following this approach, microservices can massively run in parallel and complex workflows can be realized by chaining microservices. Obviously, the smaller the microservices get, the more overhead occurs on the signaling layer. Thus, the A4 team had designed the following system as a result of balancing complexity, performance, maintainability:

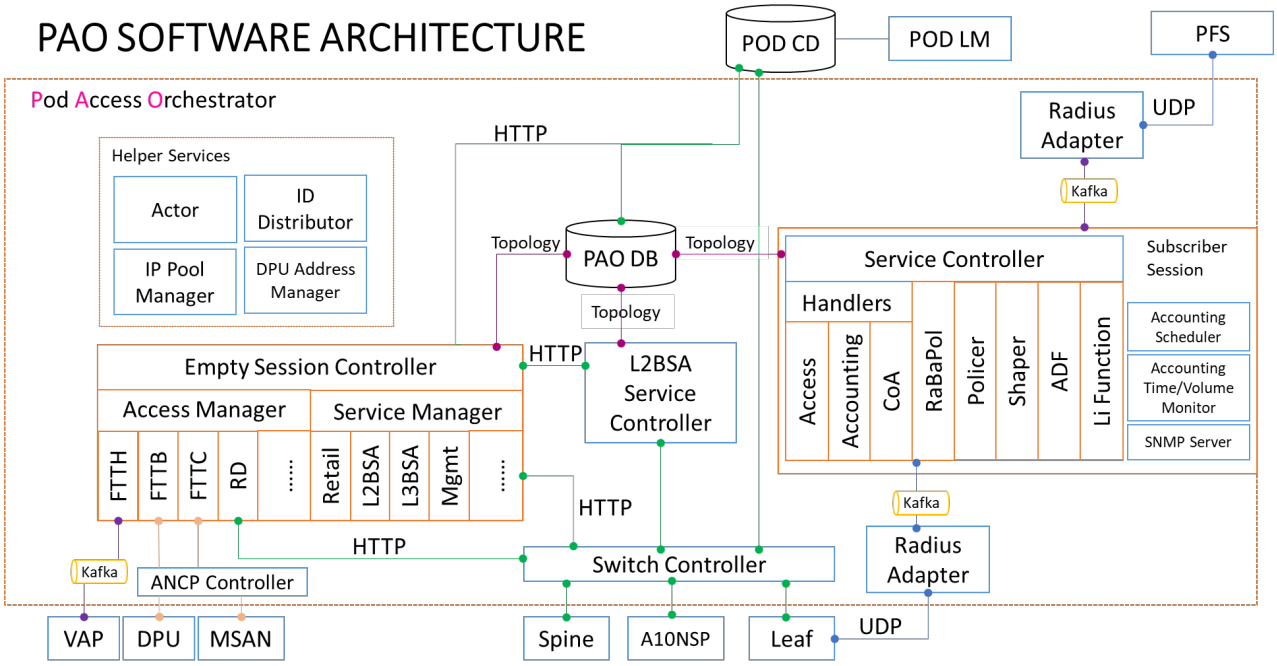

Figure 35 PAO System Architecture

As said, PAO consists of modules (microservices). Those run in individual containers, orchestrated by Kubernetes (see section on cloud platform). These modules use a set of infrastructure components such as

- PAO DB which is the PAO database, which is part of the central POD CD (central database). This database contains configuration as well as state information such as the NSP's empty session states.
- Kafka message bus. The Kafka message bus facilitates communication between modules that work asynchronously

All modules depicted in the Figure above are stateless. Most of the communication between them is over http and the Kubernetes-embedded load balancer. While Kafka is used mainly on the interfaces to external systems, inside the PAO, GRPC call over HTTP are used for highest performance.

The main modules are the controllers, which run multiple services meaning they are not very small microservices. This again is the result of the performance tradeoff. All controllers are multi-threaded. At the time of writing, only one controller instance per server is allowed. This will change soon with the delivery of A4 Milestone 5.

The **Empty Session Controller** manages the setup and life cycle of an **empty session**. An empty session basically denotes an ingress port to the Fabric, a termination port on or towards the SE and its readiness for establishing a subscriber session (PPPoE for example). While the empty session is a local mapping on a Leaf switch in case that Leaf switch includes the SE for that customer, for L2BSA the empty session denotes the path from Leaf to A10NSP handover point. Further, the empty session includes also the assigned VLANs for subscribers connected to an OLT.

The **L2BSA service controller** manages the VLAN cross connect over the empty session via the A10NSP switch towards the wholesale partner's BNG. It further is able to identify the physical port of the handover point to e.g. Vodafone or 1&1. Based on policies it can make a decision to load balance sessions.

The **Service Controller** is needed for subscribers that terminate locally in the POD. Besides the empty session, a subscriber session needs to be set up on the SE. As DT requires PPPoE, this session setup on top of the empty session requires interaction with DT's radius platform (the PFS).

The communication with the PFS is achieved via the **Radius Adapter**. This adapter talks radius to the SE and the PFS and uses the Kafka message bus to communicate with the service controller. It piggypacks radius messages over GRPC to the service controller. Although shown twice on the viewgraph, both modules are the same. K8s-embedded load balancing ensure the PFS "sees" only one IP address from the PAO (i.e. the POD). The service controller read radius messages and re-writes them for abstraction of the Rabapol syntax towards the NOS of the SE. besides the radius translation, service controller also is in charge of accounting and IP address assignment, using the IP Pool Manager.

The PAO-PFS Tutorial available on the DT wiki includes in-depth explanation of the Radius related workflows including IP address assignment.

Further modules in PAO are:

- Helper services such as IP Pool Manager and ID distributor (generating VLAN numbers for FTTH customers and managing UDP ports to PFS)
- ANCP controller, which is part of the same load balancing process as the other modules

PAO provides the functions to compose the workflows described further down in this sections. Especially those for network attachment via various access technologies.

On the southbound towards the action devices, the device controllers come into play. Those are also described further down here.

Given the flexible architecture, PAO is designed for change. It can easily be amended by adding additional control modules.

3.6.3 A4 Components: Device Controller Layer

The Device Controller layer provides technology specific management and Control for the POD. All the specific device Controllers:

- PON-C
- DPU-C
- SWITCH-C
- A10NSP-C
- LI-C
- BOR-C

Interface with the POD Local Manager and the PAO/SDN Controller accepting Management Resource and Operational Data for use in generation of device and subscriber specific configuration as well as service orchestration control requests. The Management Abstractions are described in Section 'A4 Architectural Model and Interfaces

This section describes the interface between the OSS-IT and the Central EMS and the Information model defining the Abstraction provided by the Logical versus Physical view of the hardware components.

Access 4.0 Architectural Model, Logical Resource View' and the intent for Technology Specific Access Nodes to be viewed as simple Ethernet Switches with standard Tagging operations and ACL operations, the technology specific parts are abstracted into technology specific Profiles applied to service flows when needed. For instance the PON-C Controller integrates PON specific Profiles to abstract the specific Transmission Convergence Layer attributes like TCONTs and xGEM Port IDs (see, for example, the ITU G.984 standards as well as the BBF TR-385 standard), similarly the DPU-C abstracts the G.fast specific profiles and the two stage Service Flow establishment is enabled with standard Ethernet Flow configurations along with associated technology specific profiles.

In addition to the functional requirements supported by Controllers they additionally support a set of non-functional requirements along with the Local Manager and PAO/SDN Controller including:

- Metrics
- Logging
- Notifications
- Tracing
- Diagnostics

The device Controllers are also responsible for managing the physical inventory present in the POD database. Physical Inventory is automatically propagated from A4-POD to A4-EMS (using the existing mechanisms of Master Sequence A4-M-SEQ-004 : A4-M-SEQ-004) and from there to OSS-IT. The Physical Resource Concept is described in the Telekom Wiki page: A4 Physical Resource Concept.

3.6.3.1 POD DPU Controller (DPU-C)

The DPU Controller provides support for the Management (FCAPS) and Control of the DPU hardware abstractions and capabilities including supporting the A4 POD internal facing interfaces (Northbound from the DPU-C):

- Local Manager Logical/Physical Resource interface (**IF-A4-PODLM**)
- PAO SDN Control interface (**IF-A4-PAO-DC**)

And adapting for and supporting the Southbound DPU Adapter Netconf/Yang interfaces:

- **IF-A4-DC-Device (Netconf/Yang between DPU-C and DPU device)**

The DPU controller will take the A4 Management Resource and Operational data and convert to a Standards based Broadband Forum Yang Model (BBF standards TR-301i2 and TR-355) and use the standard Netconf/Yang interface for Management and Control of the DPUs designed using the BBF TR-301 Architecture and Requirements for Fiber to the Distribution Point models.

Metrics, Logging, Notifications and Diagnostics are supported in addition to the generation of specific configurations for the supported DPU Adapters based on Templates provided for DPU operation.

As part of DPU startup the DHCP based Netconf 'Call Home' protocols are supported and the DPU will provide X.509 certificates to the DPU Adapter for authentication purposes.

The DPU Controller will in addition be responsible for generating Termination Points for the G.fast Ports supported by the DPU and creating Resources in the POD CD.

When the DPU Controller receives NSP resource records for the FTTB subscribers associated with a Termination point the DPU Adapter will configure the G.fast port and a FSoL can be reported to PAO via the DPU ANCP Client with the ANCP Port Status messages 'Port Up/Port Down' for FSoL "Port Up".

3.6.3.2 POD OLT Controller (PON-C)

The OLT Controller incorporates the ONF ONOS, VOLTHA, Device Manager open-source components which support Management (FCAPS) and Control of the OLT and ONU hardware Abstractions via the A4 POD internal facing interfaces (Northbound from the PON-C):

- Local Manager Logical/Physical Resource interface (**IF-A4-PODLM**)
- PAO SDN Control interface (**IF-A4-PAO-DC**)

And adapting for and supporting the Southbound OLT Adapter Netconf/Yang interfaces:

- **IF-A4-DC-Device** (Netconf/Yang between PON-C and Adtran OLT Device Manager/OLT Adapter)

The PON-C is responsible for detecting and configuring the OLTs and ONUs in terms of PON service Flows, utilizing the ONF Technology Profile model to abstract the PON Transmission Convergence Layer attributes with template/profile values and manage the PON Upstream Bandwidth Profiles for PON DBA operation.

Vendor specific OLT Adapters and standard ONF OLT Adapters are supported to convert the Flow and Technology Profile information into either vendor specific or open ONF messaging to the Agent resident on the specific OLT.

The PON-C provides Device Management, Metrics Collection, Software Artifact management and the triggering of FTTH Customer '(FSoL)Port Up'/'Port Down' Events to PAO causing the establishment/Tear-Down of Subscriber Sessions.

The PON-C also detects and configures the ONU part of DPUs and notifies the PAO of DPU FSoL and 'Port Up'/'Port Down' Events triggering the PAO to establish DPU 'Empty' Sessions.

3.6.3.3 POD Switch Controller (Switch-C)

The PAO uses the Fabric/Switch Controller to manage the POD Switches: Leaf/Spine/A10NSP.The Switch Controller implements the Switch specific Drivers to communicate with the switch specific Agents.
Switch-C is responsible for generating the switch startup and running configuration based on Operational data from the A4 Information Model and static configuration in the POD.

Switch-C provides support for Device Management, Metrics Collection, Software Artifact management, Logging, Notifications, Tracing, Diagnostics.

Switch-C works in the following way for device configuration: it has access to a reporsitory of configuration templates as well as generic operational data. Switch-c fills these templates with the operational data. In doing so, storing the operational data is fully decoupled from the actual representation on device level. This shall allow for much easier and faster integration of other devices / NOS in the Fabric.

The PAO Empty Session Manager additionally uses the Service Edge Switch-C Controller to create a PPPoE Termination when setting up an Empty Session.
The Switch-C Service Edge Controller interfaces with PAO and the Radius Adapter to generate a Radius Access Request from the PAO to the PFS.
A Radius Access Accept response with the Subscriber Profile and traffic management parameters is forwarded from the Radius Adapter to the Leaf Switch via the Switch-C SE Controller Radius interface.
The Leaf Switch SE also interfaces via Radius Protocol to the Switch-C SE Controller to perform accounting requests (Start, Interim, Stop) and COA radius requests.

3.6.3.4 POD Lawful Intercept Controller (LI-C)

The POD Lawful Intercept Controller is responsible for interfacing to the Lawful Intercept Server to enable the configuration for mirroring of explicit subscriber traffic for government interception. The Lawful Intercept Controller implements the specific Drivers to communicate with the LI specific Agent and is responsible for generating the LI startup and running configuration based on Operational data from the A4 Information Model and requests for LI from the PFS.

LI-C provides support for Device Management, Metrics Collection, Software Artifact management, Logging, Notifications, Tracing, Diagnostics.

3.6.3.5 POD Bottom-of-Rack Controller (BOR-C)

The BOR Controller is responsible for interfacing with the specific switch used to provide connectivity to the T-DCN network and generating the configuration for that device. The BOR switch connects the POD Servers, Spine/Leaf/A10NSP switches, Lawful intercept Server.

BOR-C provides support for Device Management, Metrics Collection, Software Artifact management, Logging, Notifications, Tracing, Diagnostics.

3.7 Access 4.0 Service Orchestration

Access 4.0 is planned to support a combination of Access Technologies including:

1. FTTH (Fiber to The Home)
2. FTTB (Fiber to The Building)
3. FTTC (Fiber to The Curb) – MSAN Legacy Deployment Migration
4. Direct Ethernet (Also termed Remote Device)

Production Models:

A. Retail (PFS)
B. Wholesale L3BSA (PFS)
C. Wholesale L2BSA (POD_LOCAL - except for FTTC which is always PFS)
D. GK-INNO/DCIP (Business IP)/EVPL (L2-EVPL)

Table 3 summarizes the existing Access 4.0 production models

Production Model	Retail - PPP Termination and Aggregation (PTA)	L3BSA/IP BSA	L2 BSA
Service Edge Location	PPPoE Local Termination in Service Edge of Leaf Switch	L2TP Access Concentrator (LAC) in Service Edge of Leaf Switch	L2X from SE to A10NSP switch, Service Edge in Wholesaler network
Service Encapsulation	PPPoE to IP	PPPoE to L2TP Encapsulation of PPP with MPLS Label	ANP VLAN encapsulated with L2X MPLS Label
Access 4.0 Northbound Egress Point	Spine Switch to IP2 Network	Spine Switch to IP2 Network	A10NSP NNI to Wholesaler
Subscriber Profile derived from:	PFS	PFS	POD-CD (NSPL2BSA) except for FTTC which is always PFS
Subscriber Accounting	Radius	Radius	A10NSP

Table 3 Service Production Models for Access 4.0

3.7.1 FTTH Service Orchestration

The A4 POD orchestrates subscriber services based on an initial First Sign of Life (FSoL) triggered from the PON-C Controller for PON based FTTH services as a Subscriber ‘Port Up’ Notification.

The FSoL is sent to the PAO/SDN Controller which will initiate the Subscriber Session Context to be created in the PON-C controller managed components (OLT/ONU) and the SWITCH-C controller managed components (Fabric/Service Edge Switch). This first stage Session Context is considered the ‘Empty’ Session state awaiting the subscriber control protocols/packets to bring up the session itself, e.g., PPPoE Discovery phase packets.

Note that for L2BSA services the Empty Session instantiates a cross connect from one Subscriber denoted by the NspFtthAccess Resource associated with an FTTH PON Termination Point and a second Subscriber NspL2BSA Resource and A10NSP switch Termination Point - the Service Edge does not exist in the Local POD and there is no PFS AAA lookup required.

For FTTH subscribers the FSoL (‘Port Up’ indication) is also considered the indication fully identifying the subscriber as being ready to participate in service establishment and allowing PAO to look up the Network Service Profile Record for that subscriber based on the ONU Serial number and allocate dynamic resources such as the Outer VLAN Tag to be applied at the OLT depending on the type of service required and the Production Scheme required e.g. ‘Policy Controller Lookup’ or Local ‘POD CD’ Resource definition of the service parameters.

3.7.1.1 FTTH Retail Service

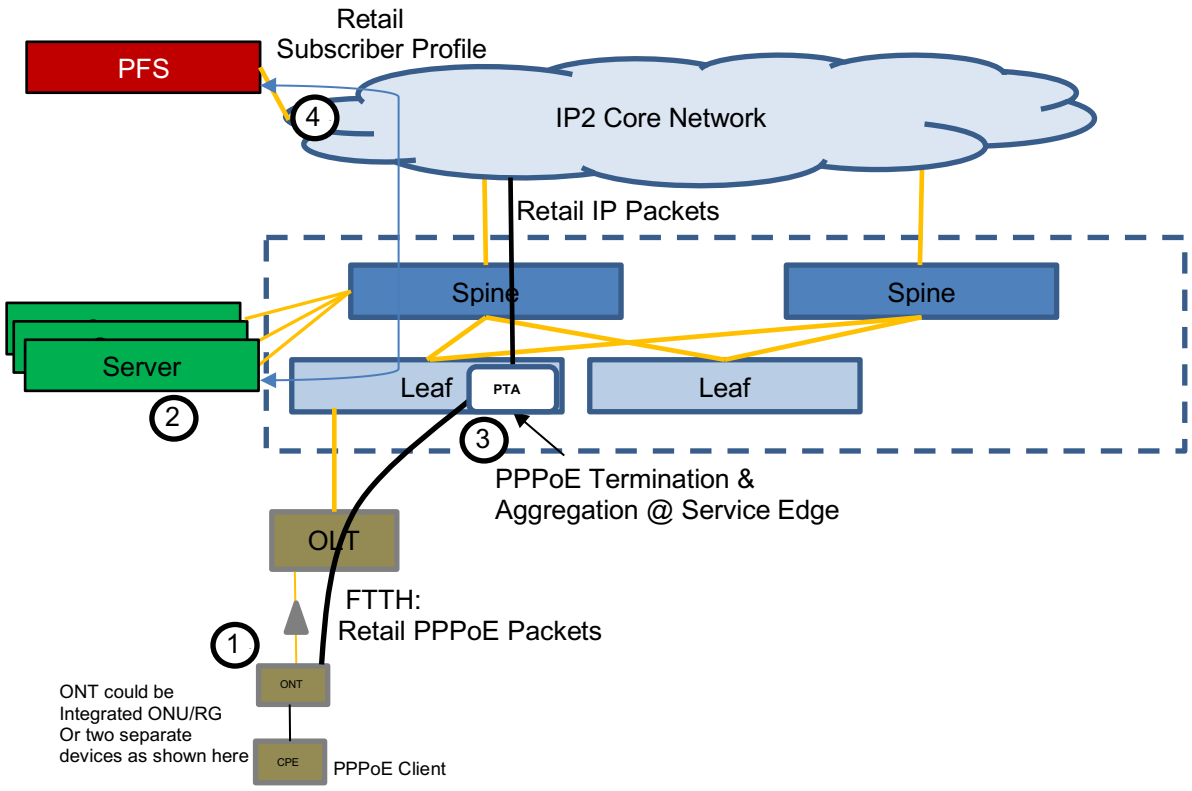

Figure 36 FTTH Retail PPPoE Service Orchestration

1. For the FTTH Customer the arrival of the ONU at the OLT PON Port causes the ONU to be Ranged in and the OLT to establish the ONU default TCONT/GEM Port for physically connecting the OMCI Management channel between the ONU and the OLT.

2. A **"Port UP" FSoL** indication to the PAO/SDN Controller from the PON-C Controller triggers the PAO to create the "Empty Session" steered to where the selected Service Edge termination point is located i.e. usually, the connected Leaf Switch. PAO uses the PON-C controller to configure Subscriber service flow on the PON adding an Outer ANP Tag (dynamically selected) on the OLT to identify the Customer uniquely. PAO also uses the SE-C Controller to configure the Service Edge on the Leaf Switch. Note : more detail on Empty session establishment is described here: FTTHRetailEmptySessionEstablishment

3. The Client side PPPoE Discovery phase begins.

4. The SE will request PAO via Radius to contact the PFS to authenticate the subscriber and assign IP addresses and QoS Policy. Note more detailed Retail Session Activation is described here: Retail Session Activation

Once SE configuration is complete and the PPPoE session established the customer can reach the services he has subscribed to.

The A4 POD Components orchestrate the FTTH Retail Empty Session as defined in the following Message Sequence Diagram:

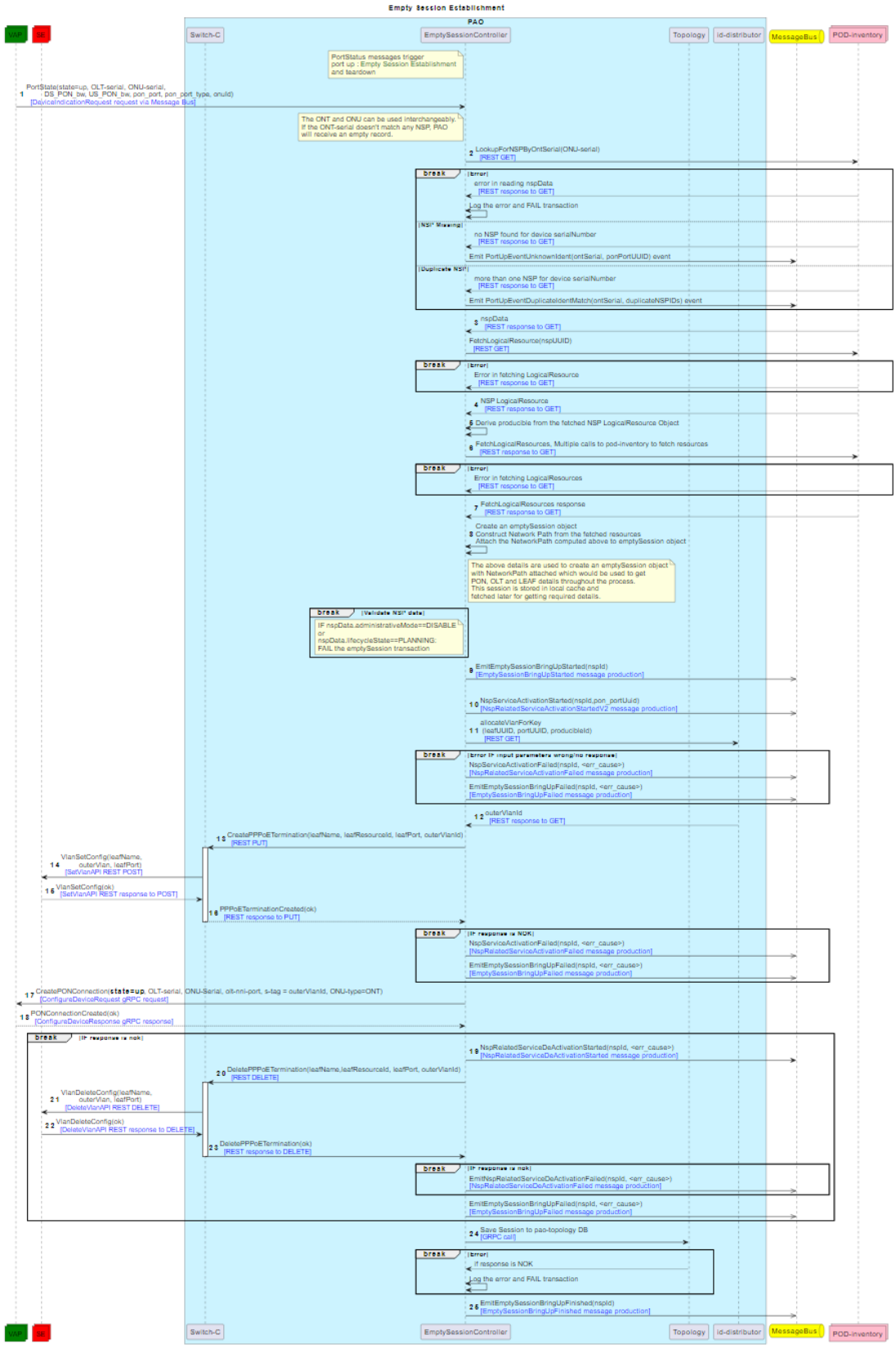

Figure 37 A4 POD Orchestration of FTTH Retail Service Empty Session Activation

This flow as well as the one for L2BSA have been harmonized with the flows in BBF WT-474. While the terminology is slightly different, the informational elements conveyed and actions executed are the same.

Note: The PAO / Session management related workflows are documented in the Telekom Wiki for every release. For current Uber-Ice, the flows can be found here
[UberIce Workflows - Session Management [Draft]]

3.7.1.2 FTTH L3BSA Service

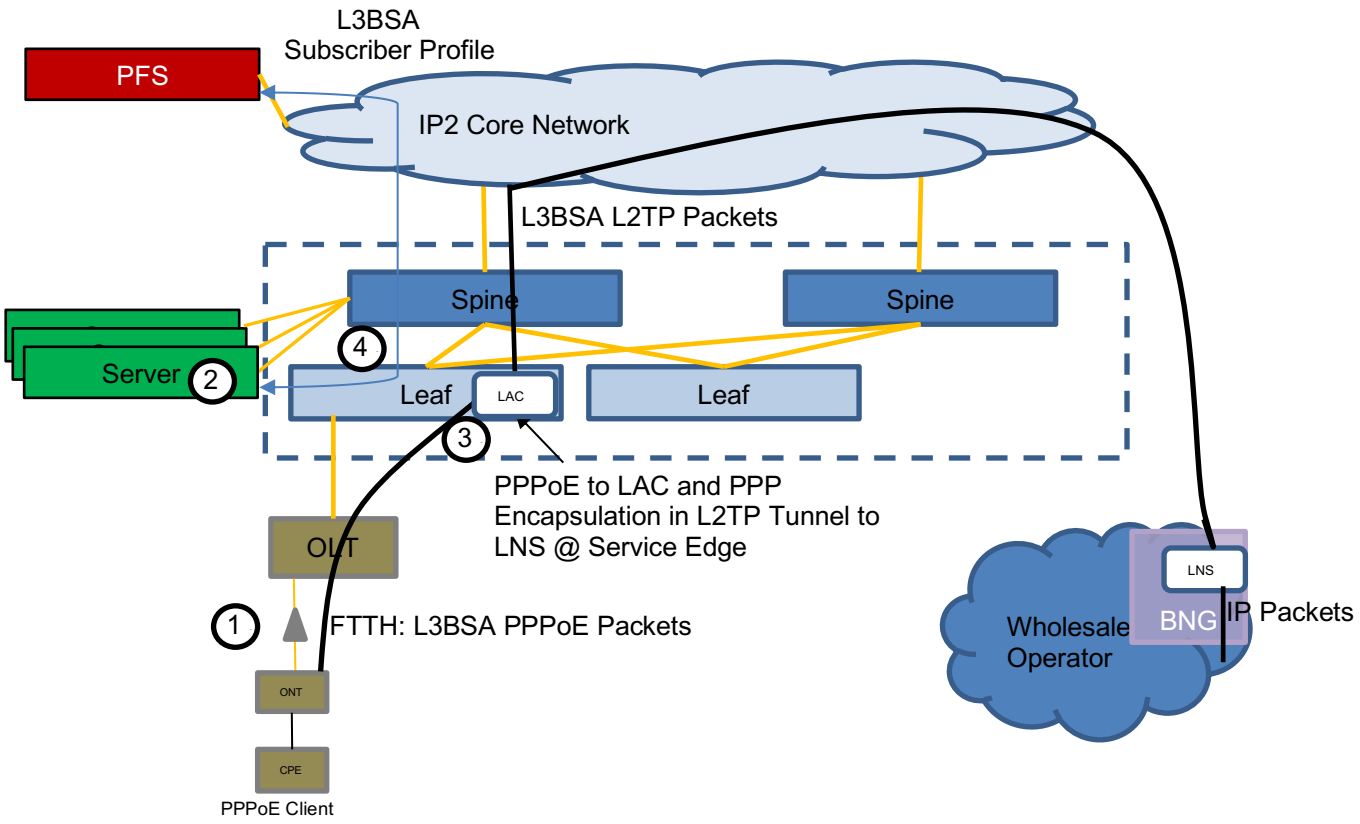

Figure 38 FTTH L3BSA Wholesale Service

The DT L3BSA (FTTH) service Steps 1 through 4 are the same as for the FTTH Retail service with the exception that the PFS access request will return L2TP tunnel parameters and QoS Policies.

The L2TP service is described in the RTBrick documentation here: [l2tp_profile_configuration]

And the implementation is described in more detail here:

[L2TP Specific RaBaPol Services - Access 4.0 - Telekom Wiki]

[SE Services Overview and RaBaPol Mapping table (L2TP) - Access 4.0 - Telekom Wiki]

3.7.1.3 FTTH L2BSA Service

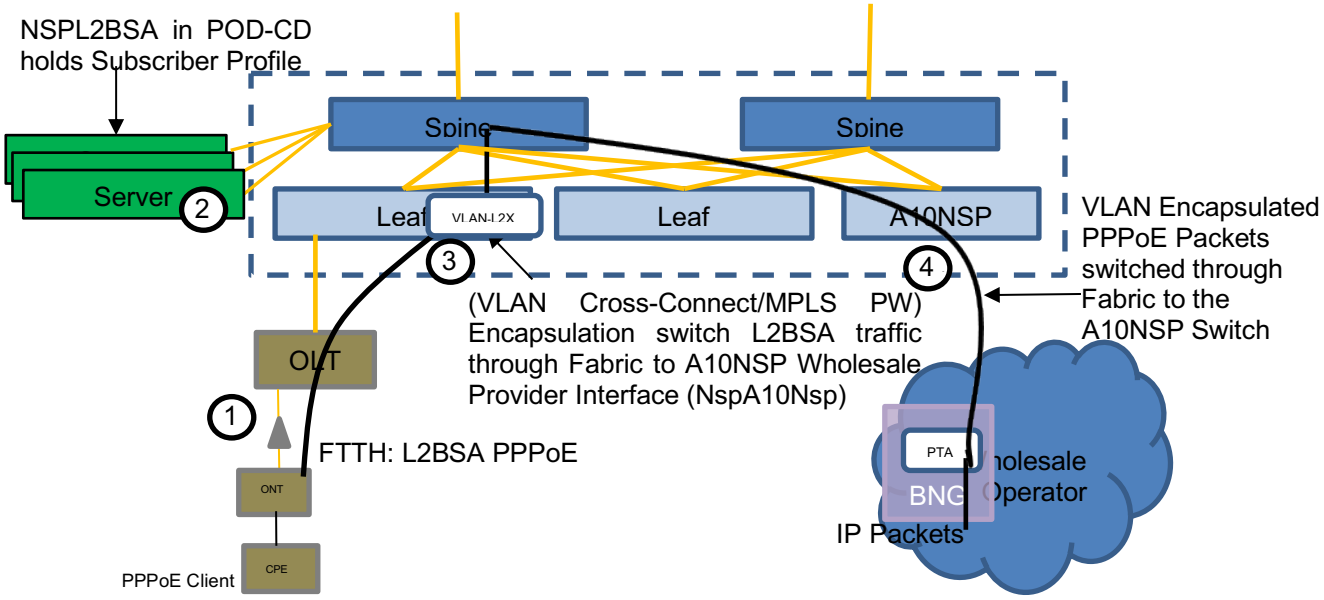

Figure 39 FTTH L2BSA Wholesale Service

The DT L2BSA (FTTH) service Step 1 and Step 2 are the same as for the FTTH Retail service with the exception that at Step 2 the PAO will see that the NspFtthAccess Resource Operational data indicates that the Production scheme == 'POD_INTERNAL' with a valid 'relatedNsp' attribute 'NspL2Bsa'.

At Step 2 the operation of the L2BSA service becomes different from the Retail and L3BSA model since in this case there are essentially two subscriber endpoints which need to be connected through the POD Fabric. Additionally, the wholesaler takes on the responsibility of AAA services and so the A4 POD itself need only rely on the POD Local service configuration without needing to contact the PFS.

Previously when the A10NSP LAG interface connected to a third-party wholesaler is turned up and the NspA10Nsp (Port connecting to the third-party wholesaler on the A10NSP switch) port is in the 'Working' state then Figure 32 step 1 applies for the A10NSP port and the A4 POD generates the number of Termination Points required for the range of supported VLANs on the Port. (Note the NSPA10Nsp could also be in a LAG).

In this case when ordered by the Wholesale Provider the DigiOSS associates an NspFtthAccess endpoint on an ONT with a NspL2Bsa endpoint (VLAN) on the A10Nsp Switch NspA10Nsp Port.

Once a FSol is detected, and 'PortUp' reported to the PAO by PON-C then as part of Step 2 PAO selects an ANP VLAN for the L2BSA Subscriber and establishes the operation to add the outer ANP Tag to the subscriber service frames on the OLT via the PON-C controller for that subscriber and then in Step 3 on the Leaf Switch with a L2X Label encapsulation is added for the Subscriber via SE-C controller and the Fabric is configured via the Switch-C controller to forward the subscriber traffic to the A10Nsp Port. Step 4 has the

PAO use SE Controller install the final VLAN translation to the desired Wholesaler S-VLAN - see Figure 40. Note: See Section 'Fabric Overview' for a description of switching Pseudo-Wire encapsulated frames through the Fabric.

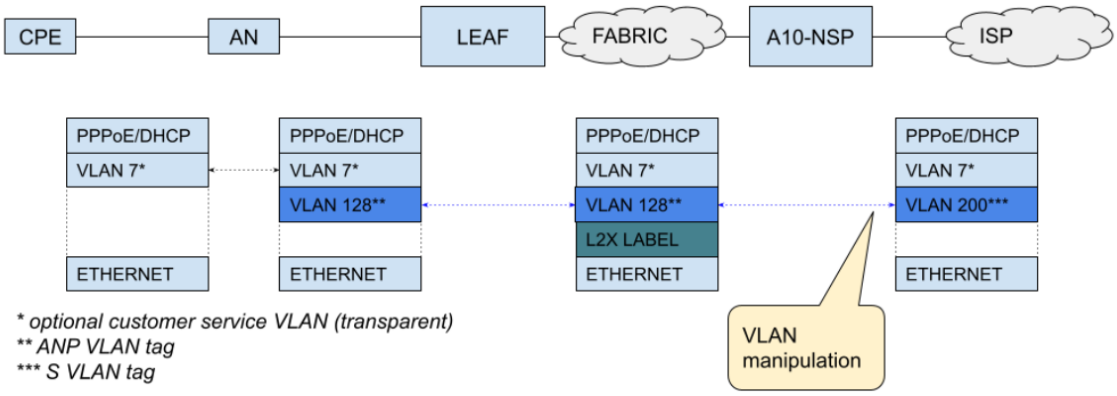

Figure 40 L2BSA Service as described in the 'RTBrick L2BSA Guide' Figure 4

The L2BSA service is described in the RTBrick documentation here: L2BSA User Guide

Detailed design of the L2BSA Service Activation/Deactivation is described here: L2BSA Service Activation/Deactivation

The current L2BSA flow is as follows:

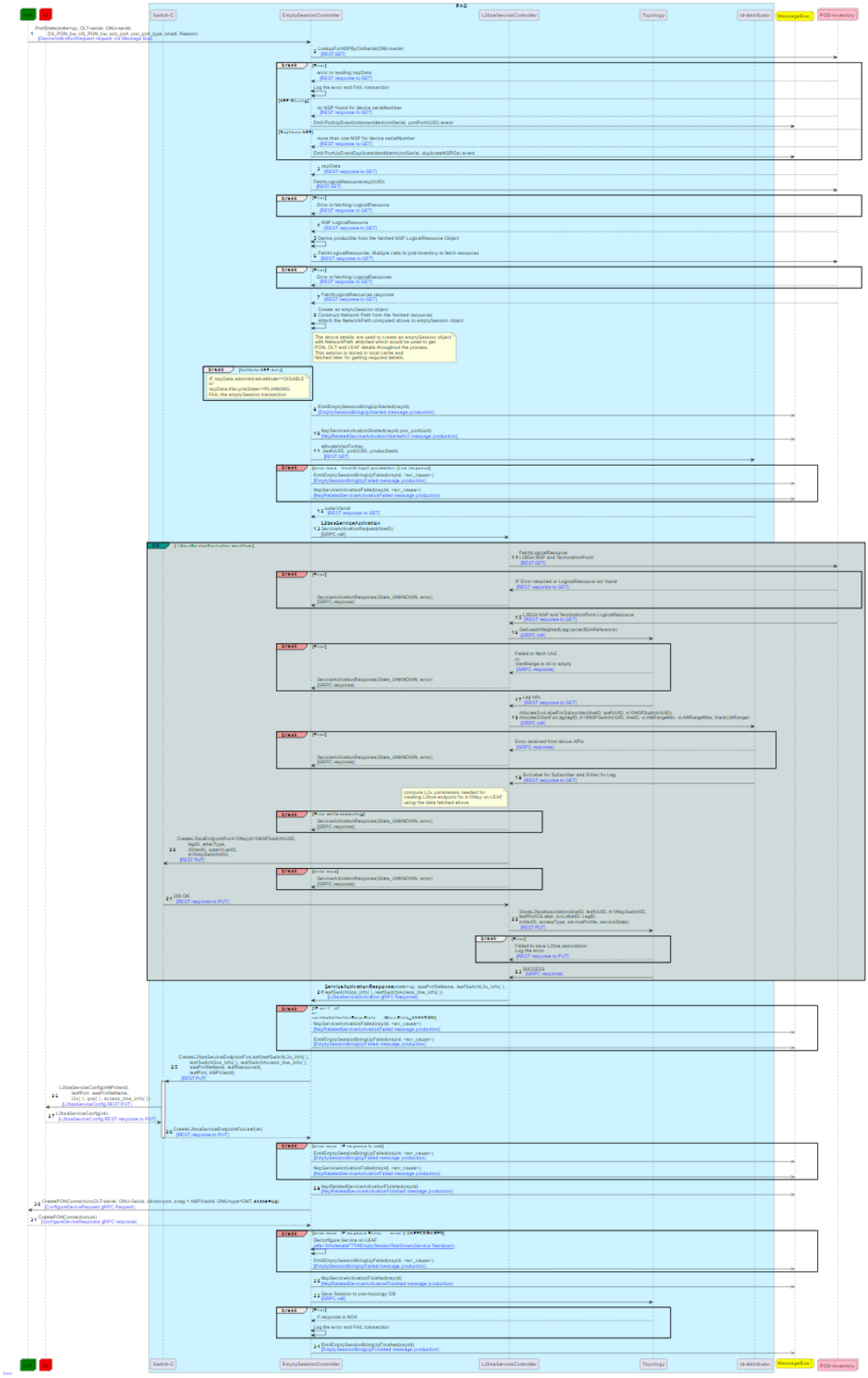

Figure 41 Figure A4 POD Orchestration of FTTH L2BSA Service Activation

3.7.2 FTTB Service Orchestration

For FTTB Subscribers the DPU FSoL represents the detection of the DPU ONT function by the PON-C controller. When the DPU/ONT "PortUp" status is reported to the PAO it will not find an 'NspFtthAccess' NSP Resource associated with the reported ONT Serial Number, however, in this case the PAO will check to see if a DPU NE Resource is associated with it and if so PAO recognizes that a DPU FSoL has been reported (see Messages Sequence in Telekom-Wiki: https://gard.telekom.de/gardwiki/display/ACC4/Master+Sequence+3D+-+POD+Hardware+Extension+for+FTTB+DPU).

Note that the DPU NE Resource has already been created along with the NEPs and NEL associated with the DPU NE due to the Technician entering the DPU Serial Number and its' attachment point (OLT PON tree). The Local Manager update of these Objects in the POD CD will have triggered Object Create/Change Event messages to the DPU-C controller allowing generation of the DPU configuration. Termination Point updates for the DPU are sent to the Local Manager and forwarded to A4-EMS and the subsequent update to OSS-IT will cause the updating of the A4 POD CD with the TP associated 'NspFttbAccess' resources (see section on architectural model).

After DPU Boot up and initial Discovery by the PON-C, the PAO will report detection of the new DPU hardware to the A4 POD and will notify that the DPU installation has started. The DPU itself will proceed to DHCP Discovery which allows acquisition of an IP Address and the information required from DHCP Options to begin the "Call Home" phase and verify the DPU Artifacts are up to date, authenticate with the DPU Controller and retrieve a configuration update via Netconf. Subsequently the DPU will connect its' local ANCP Client to the PAO ANCP Server. (See DT Concept Paper for DPU and FTTB support in A4 architecture: DT Project A4 Concept Paper DPU@A4 V5.1)

Additional details around DPU bring up are described in the Telekom Wiki: DPU-Planning and -autoinstallation/-startupThis is a subsection of the Component Lifecycle (SW/HW) Telekom Wiki: Component lifecycle (SW/HW)

As will be described later the DPU will require special handling by the PAO due to the need to minimize the per G.fast subscriber ONU part flow provisioning over the OMCI interface because any change of OMCI provisioning will cause all subscriber traffic to lose connection with the OLT for a short period of time. This is an artifact of the DPU implementation not an A4 POD requirement.

In the DPU case the PAO will assume all NspFttbAccess Resources to be available for a DPU/ONU after the DPU FSoL is received. The PAO will configure the OLT Controller with the required Subscriber Flows and the OLT will be programmed with the needed (unique per Subscriber) S-Tag values to swap with the G.fast ANP-Tag in the frames received upstream from the DPU and the reverse in the downstream direction from OLT to DPU (See **Error! Reference source not found.**). Note the current proposal differs from the proposal in the DT Project A4 Concept Paper DPU@A4 V5.1. where the ANCP "PortUp" status message would cause PAO to dynamically allocate a unique subscriber S-Tag and configure the Subscriber Empty Session with the subscriber tagging and the Service Edge from the DPU to the Leaf switch itself like the FTTH model. The model described requires a static S-Tag allocation from the PAO for all NspFttbAccess Resources associated with the DPU NEP Resources/Termination Points independent from whether the subscriber is active or not.

For FTTB the Subscriber 'Port Up' indication is independent of the DPU detection on the PON itself and is communicated via ANCP Client 'Port Up' ANCP status messages using the ANCP VLAN encapsulation informing the PAO ANCP Server of the G.fast subscriber port state and identifying the 'Line-ID' and other attributes of the specific Subscriber interface. The Service Edge will not be programmed until the DPU sends an ANCP 'Port Up' FSoL status message to PAO indicating that the Subscriber is ready to connect.

3.7.2.1 FTTB Retail Service

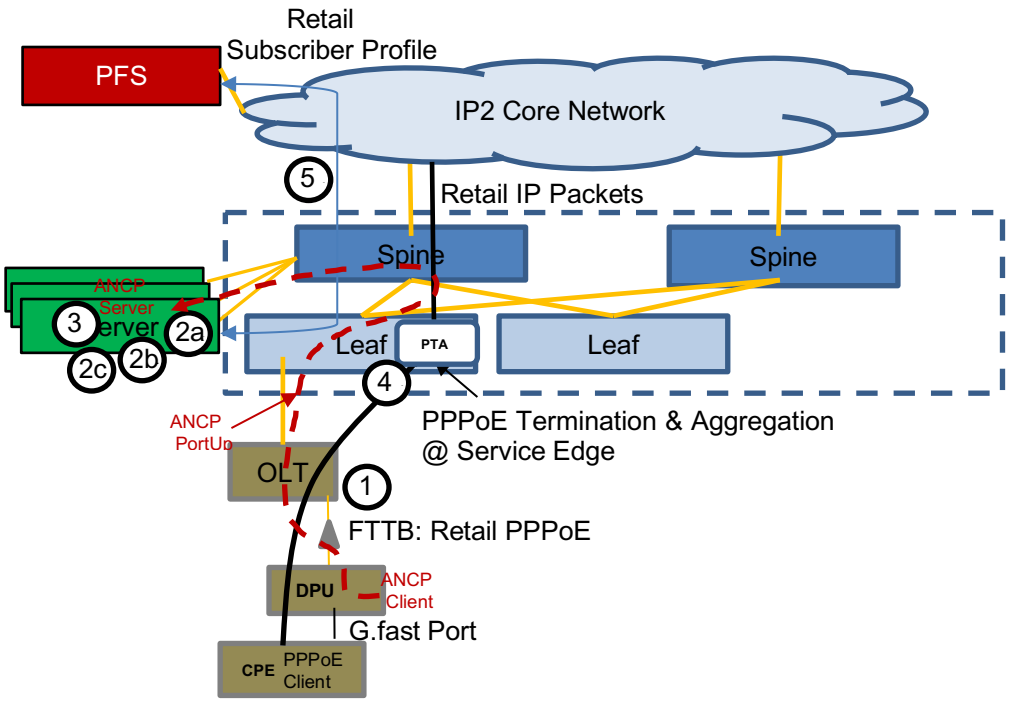

Figure 42 FTTB Retail Service

1. For the FTTB Customer the arrival of the DPU/ONU at the OLT PON Port causes the ONU to be Ranged in and the OLT to establish the ONU default TCONT/GEM Port for physically connecting the OMCI Management channel between the ONU and the OLT.

2. For FTTB Subscribers the DPU FSoL represents the detection of the DPU ONT function by the PON-C controller.

When the DPU/ONT "PortUp" status is reported to the PAO by PON-C it will not find an 'NspFtthAccess' NSP Resource associated with the reported ONT Serial Number, however in this case the PAO will check to see if a DPU NE is associated with it and if so PAO recognizes that a DPU FSoL has been reported. (Master Sequence in Telekom Wiki: A4-M-SEQ-003D: POD DPU HW Extension describes the sequence of events around a DPU hardware change).
The PAO will orchestrate the PON-C such that the service Flows for Management and ANCP Control of the DPU/ONU are implemented.

a. Once the DPU can communicate with the POD DHCP Server the DPU will send a DHCP Discover and the POD Server will authenticate the DPU and assign its Management IP Address and the pre-defined options for enabling a 'Call Home' protocol, allowing connection with the DPU-C Netconf client and subsequent receipt of the auto-generated configuration via Netconf protocol.
b. When the PAO is notified that the DPU Installation is complete the PAO will provision the PON-C controller with all potential service flows for the customers on the G.fast ports of the DPU statically allocating the OLT S-Tag VLANs to swap for the pre-configured DPU

ANP-Tag. Note: a notification of DPU/ONT 'Port Down' from the PON-C to the PAO will cause all subscriber service provisioning to be removed by the PAO. When an ANCP 'Port Up' is received by PAO for an FTTB Retail subscriber the per Subscriber PPPoE SE session configuration for the Empty Session is established. The Client side PPPoE Discovery phase begins, and FTTB follows the FTTH model once the DPU is brought up and the per Subscriber Empty Session established.

4. The SE will request PAO via Radius to contact the PFS to authenticate the subscriber and assign IP addresses and QoS Policy. Once the PPPoE session is established the customer can access the services to which he has subscribed.

3.7.2.2 FTTB L3BSA Service

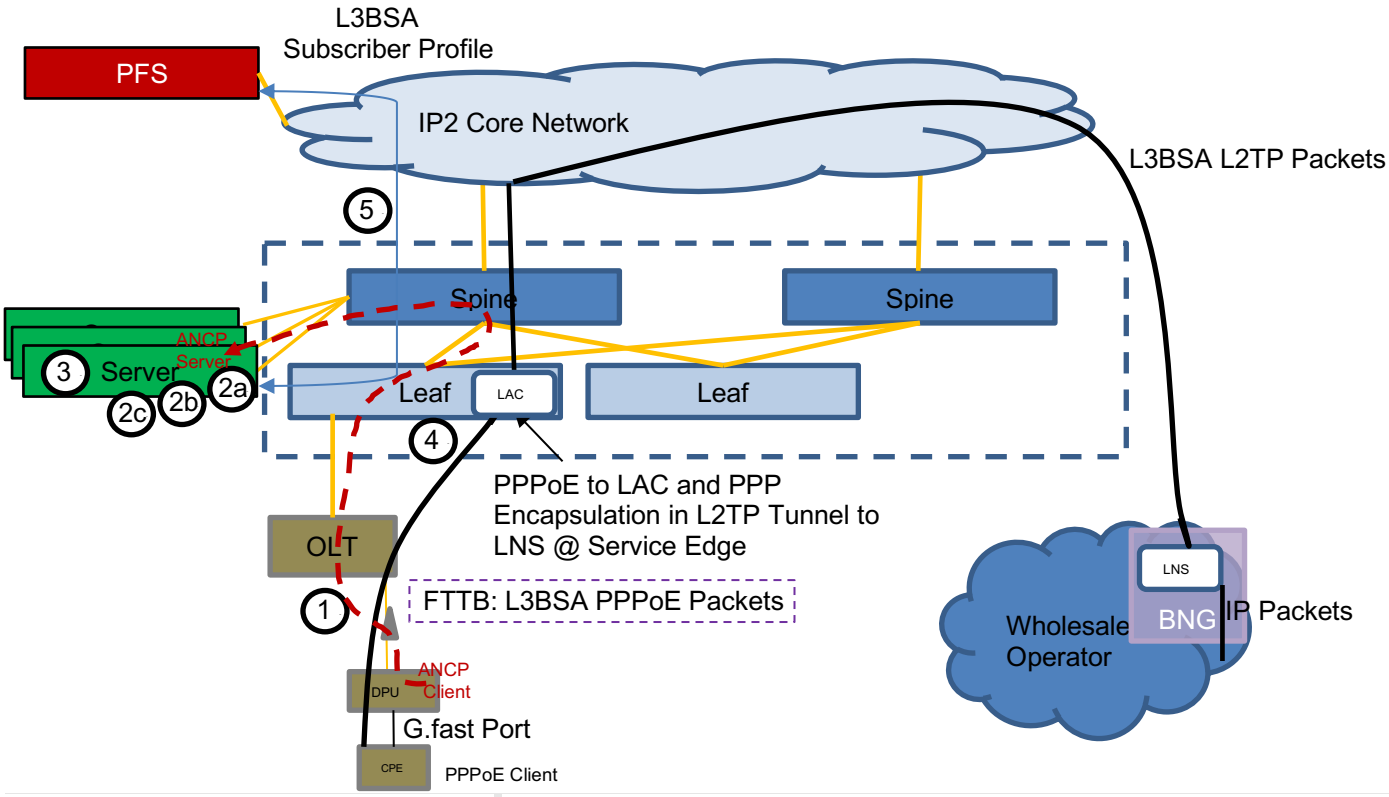

Figure 43 FTTB L3BSA Wholesale Service

The DT L3BSA (FTTB) service Steps 1 through 4 are the same as for the FTTB Retail service with the exception that at Step 5 the PFS access request will return L2TP tunnel parameters and QoS Policies.

The L2TP service is described in the RTBrick documentation here: l2tp_profile_configuration

The implementation is described in more detail here:

L2TP Specific RaBaPol Services - Access 4.0 - Telekom Wiki

SE Services Overview and RaBaPol Mapping table (L2TP) - Access 4.0 - Telekom Wiki

3.7.2.3 FTTB L2BSA Service

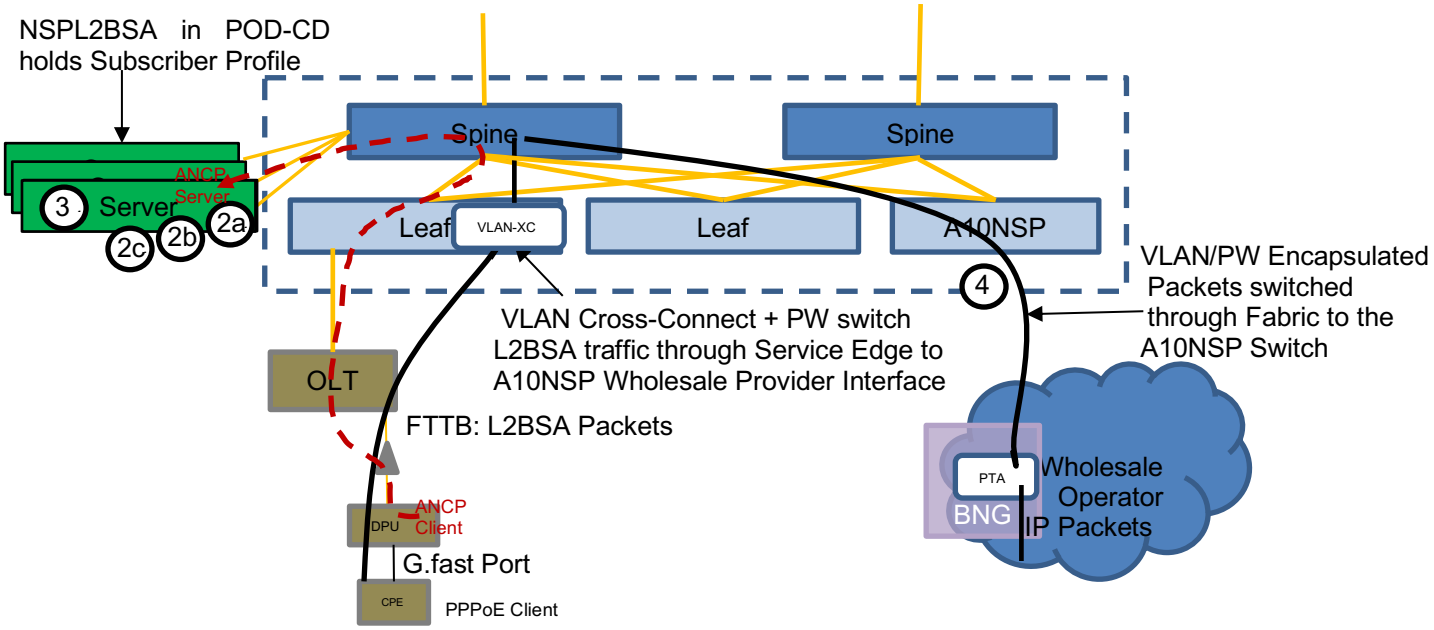

Figure 44 FTTB L2BSA Wholesale Service

The DT L2BSA (FTTB) service Steps 1 through 3 are the same as for the FTTB Retail service with the exception that at Step 3 when the ANCP 'PortUp' Status is received the PAO will see that the NspFttbAccess Resource Operational data indicates that the Production scheme == 'POD_INTERNAL' with a valid 'relatedNsp' attribute 'NspL2Bsa'.

At Step 3 the operation of the L2BSA service becomes different from the Retail and L3BSA model since in this case there are essentially two subscriber endpoints which need to be connected through the POD Fabric. Additionally, the wholesaler takes on the responsibility of AAA and LI services and so the A4 POD itself need only rely on the POD Local service configuration without needing to contact the PFS.bPreviously when the A10NSP LAG interface connected to a third-party wholesaler is turned up and the NspA10Nsp (Port connecting to the third-party wholesaler on the A10NSP switch) port is in the 'Working' state then Figure 32 step 1 applies for the A10NSP port and the A4 POD generates the number of Termination Points required for the range of supported VLANs on the Port. (Note the NSPA10Nsp could also be in a LAG).bIn this case when ordered by the Wholesale Provider the DigiOSS associates an NspFttbAccess endpoint on an ONT with a NspL2Bsa endpoint (VLAN) on the A10Nsp Switch NspA10Nsp Port.bOnce a subscriber ANCP 'PortUp' reported to the PAO by the ANCP Server then as part of Step 3 on the Leaf Switch a L2X Label encapsulation is added for the Subscriber via SE-C controller and the Fabric will forward the subscriber traffic to the A10Nsp Port. Step 4 has the PAO use SE Controller install the final VLAN translation to the desired Wholesaler S-VLAN - see Figure 40. Note: See Section 'Fabric Overview' for a description of switching Pseudo-Wire encapsulated frames through the Fabric.bThe L2BSA service is described in the RTBrick documentation

here: L2BSA User GuidebbDetailed design of the L2BSA Service Activation/Deactivation is described here: L2BSA Service Activation/Deactivation

3.7.3 FTTC Service Orchestration

For MSANs, a clearing of the LoS for the MSAN Connected Port on the Leaf Switch (MSAN physical Uplink) represents the FSoL and detection of the MSAN itself. Service Orchestration follows the provisioning and detection of an MSAN connection.

The A4 NEMS is provisioned via Digi-OSS for an MSAN connection to an A4 POD Leaf Switch by the creation of a NEP with an 'OffNetworkLink' with endpoints of the referenced NEL referencing an MSAN Termination. The NEMS will synchronize these Logical Resources to the A4 POD so it can recognise the MSAN connection to the Leaf Switch and allow the A4 POD to generate an MSAN Link 'Termination Point'. This MSAN Termination Point is synchronised to NEMS and Digi-OSS allowing an association of a NspMsanUpink NSP with the Termination Point. This MSAN TP associated NSP contains all required configuration information for the A4 POD to establish Management and Control connectivity to the MSAN including:

- Management VLAN
- MSAN Management IP Address
- ANCP VLAN
- ANCP Client IP Address
- ANCP Server IP Address
- ANCP Gateway IP Address
- POTS Card VLAN
- POTS IP Address subnet

See the description of the MSAN Management Network in Telekom Wiki : MSAN Management Network

After Management and Control Channel configuration MSAN Subscriber Services can be dynamically learned and established.

The A4 POD orchestrates MSAN subscriber services based on an initial Subscriber FSoL, which for MSAN is provided by ANCP 'Port Up' messages received after the A4 POD has established an ANCP Adjacency with the MSAN. The 'Port Up' messages contain Subscriber identifying information such as VLANs and LineId.

Usually The A4 POD has a per subscriber NSP pre-configured to allow definition of the type of service but in the case of FTTC the A4 POD follows the legacy MSAN model and only the MSAN has subscriber service configuration which is shared with the A4 POD via the 'Port Up' ANCP message.

For The MSAN Services there is some ambiguity with regard to L2BSA services with respect to other MSAN services and there is logic which must be implemented to identify a L2BSA service which is described here : Empty Session Handling With and Without an NSP.
Upon detection of an MSAN subscriber via the ANCP 'Port Up' the A4 POD will generate a Radius 'Access-Request' utilising a special User-Name the Line ID and other ANCP TLVs with an implicit assumption that the request will return an 'Access-Accept' reply.
If the L2BSA Radius 'Access-Request' results in an 'Access-Reject' the MSAN ANCP Controller will request the empty session controller to create a PPPoE Termination Point and proceed with session establishment as per the FTTH model.
This process is described in detail in the Telekom Wiki here: FTTC Pre-authorization work flow .

3.7.3.1 FTTC L2BSA Services

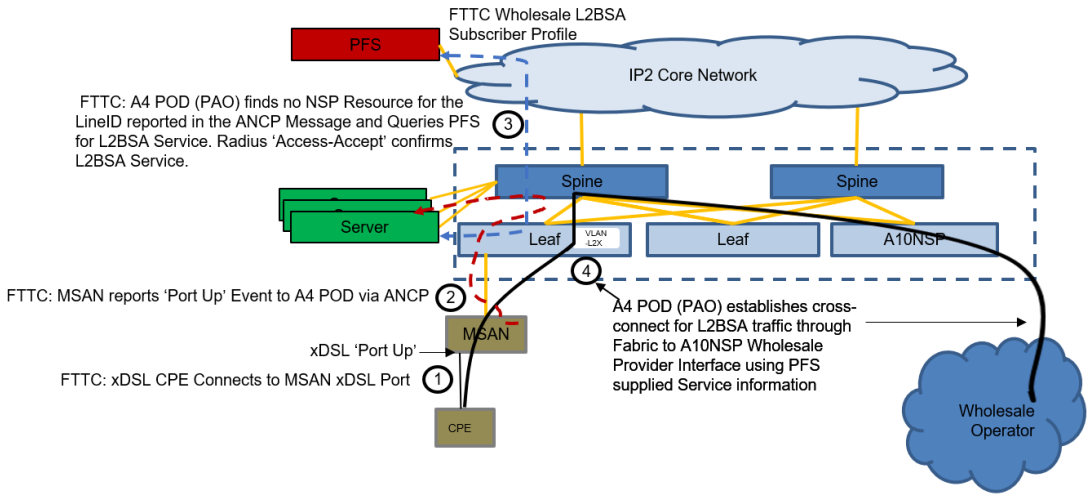

Figure 45 FTTC L2BSA Wholesale Service

The DT L2BSA (FTTC) service differs from the FTTH/FTTB model since the A4 POD does not receive per Subscriber NSP resource information either for the Access side or the A10NSP side, instead the A4 POD must depend upon querying the PFS for the Service information.

When the customer modem (CPE) successfully connects to the MSAN (Step 1 in the above Figure) the MSAN xDSL port transitions to an 'Up' state and the MSAN reports this condition to the A4 POD via the ANCP 'PortUp' Status message (Step 2 in the above Figure). When the PAO is informed of the customer 'Port Up' it will not find any matching NSP Resource records and will first check the PFS for an FTTC based L2BSA service (Step 3 in the above Figure).

At Step 3 the operation above the reception of a 'Access-Accept' from the PFS for the L2BSA service forks the processing of the ANCP 'Port Up' Event to continue with the FTTC L2BSA processing whereas reception of an 'Access-Reject' at Step 3 would cause the PAO to assume the Service is either an FTTC Retail or FTTC L3BSA model whose processing is described in separate sections of this document.

Once an 'Access-Accept' from the PFS for the L2BSA service is received by the PAO, it will proceed to Step 4 and use the Service information supplied by the PFS to establish is cross-connect between the Access Port on the Leaf Switch through the Fabric to the A10NSP switch port connecting to the Wholesale supplier. Step 4 also has the PAO use the Switch Controller to install the final VLAN translation to the desired Wholesaler S-VLAN. Note: See Section 'Fabric Overview' for a description of switching Pseudo-Wire encapsulated frames through the Fabric.

Detailed design of the FTTC L2BSA Service Activation/Deactivation is described here:
FTTC Pre-authorization work flow

3.7.3.2 FTTC Retail Services

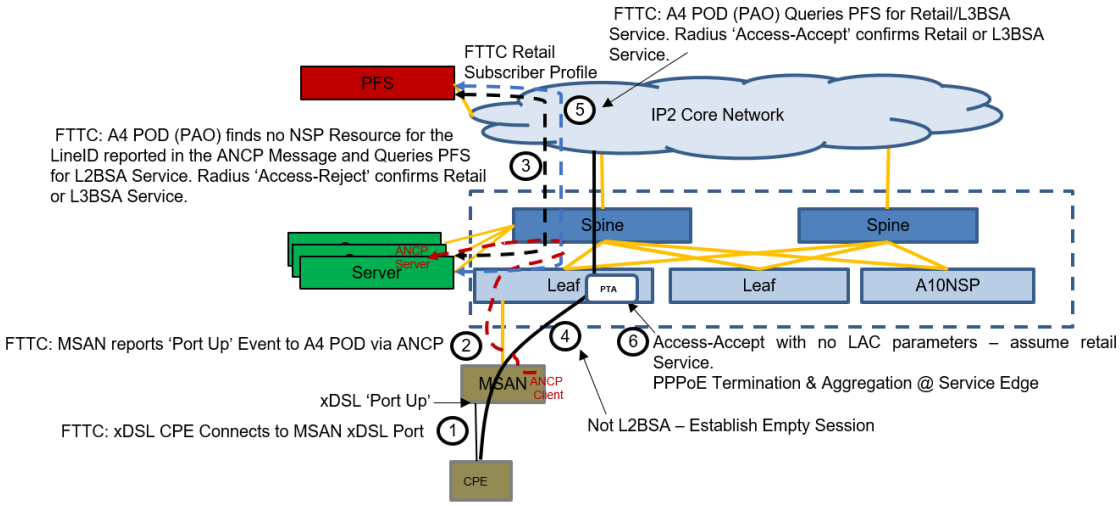

Figure 46 FTTC Retail Services – Query the PFS in all cases

Once the Test for FTTC L2BSA has failed the A4 POD will move on to step 4 and establish an empty session at the Service Edge. Steps 1 – 3 are the same as for FTTC L2BSA except the step 3 will be a Access-Reject reply from the PFS.

At Step 5 the SE will request PAO via Radius to contact the PFS to authenticate the subscriber and assign IP addresses and QoS Policy. Once an 'Access-Accept' from the PFS for the Retail service is received by the PAO and there are no LAC parameters indicating a Retail Service, it will proceed to Step 6 and use the Service information supplied by the PFS to PPPoE session establishment and the customer can access the services to which he has subscribed.

3.7.3.3 Access 4.0 Implemented FTTC L3BSA Service

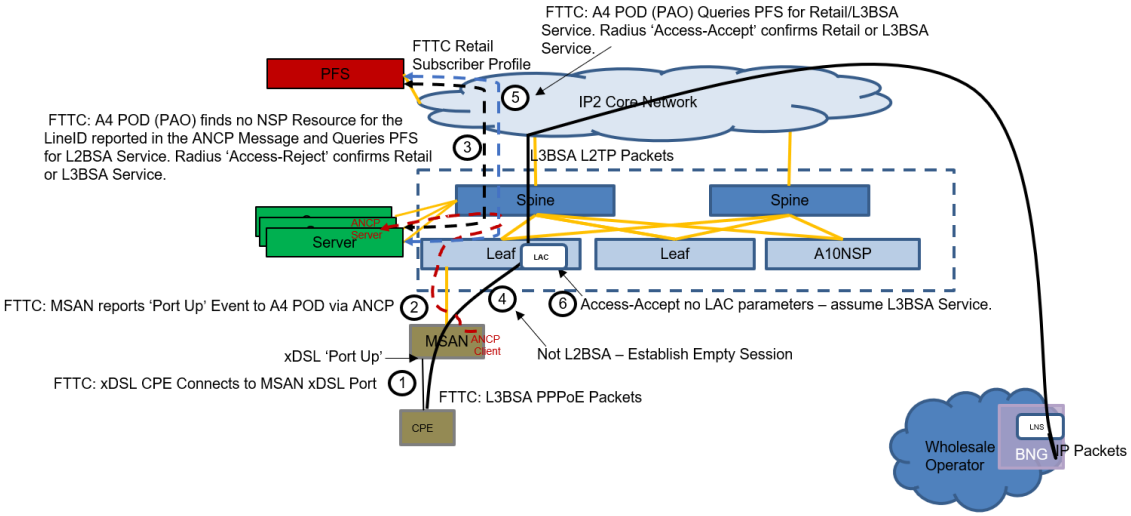

Figure 47 FTTC L3BSA Wholesale Service

The DT L3BSA (FTTC) service Steps 1 through 6 are the same as for the FTTC Retail service with the exception that the PFS access request will return L2TP tunnel parameters and QoS Policies.
The L2TP service is described in the RTBrick documentation here: l2tp_profile_configuration
And the implementation is described in more detail here:

L2TP Specific RaBaPol Services - Access 4.0 - Telekom Wiki

SE Services Overview and RaBaPol Mapping table (L2TP) - Access 4.0 - Telekom Wiki

3.7.3.4 FTTC 'Single Play' POTS Service

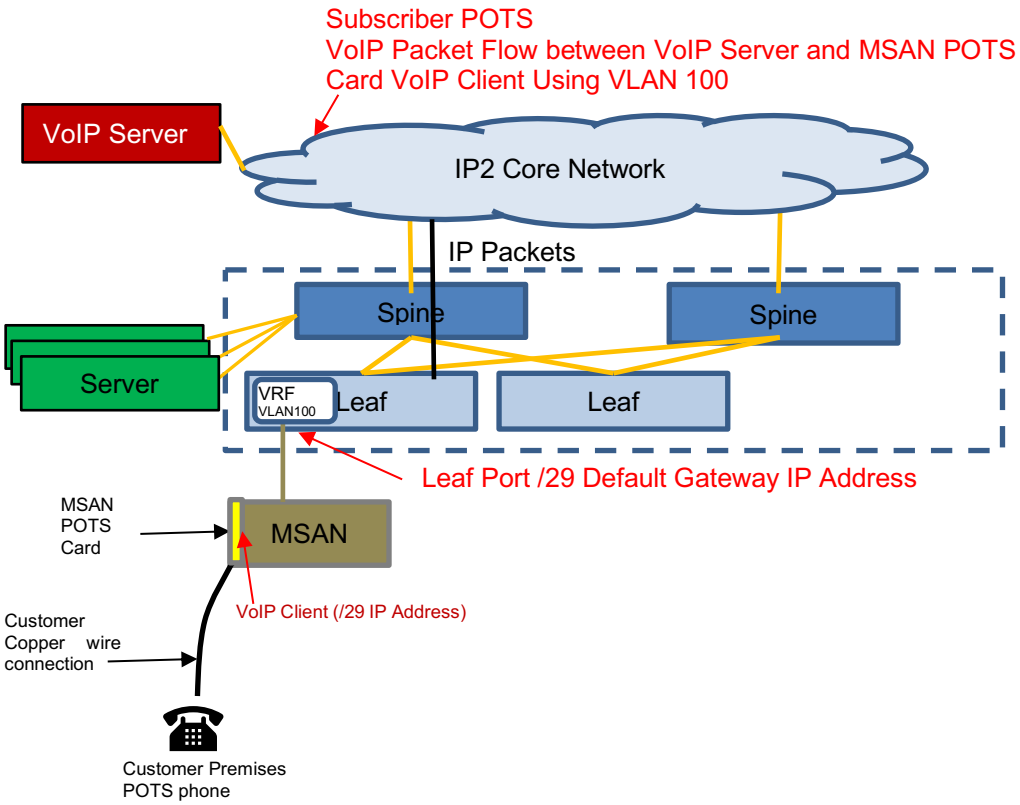

Figure 48 FTTC POTS Service

The existing MSAN deployments support a 'Single Play' Plain Old Telephony (POTS) Service where the copper lines from the customer connect to a dedicated MSAN Chassis Linecard supporting POTS terminations which provide a conversion to/from IP SIP service utilizing VLAN 100 to reach the Voip Server Platforms.
(See Telekom Wiki for detail design for MSAN POTS Telephony here: Design POTS Telephony).

Once the MSAN is connected and the Management VLAN established the MSAN will be configured for the per POTS Linecard IP Address(es) in the \29 subnet which matches with the A4 Leaf Switch Port corresponding /29 default Gateway address.

The MSAN /29 IP addresses are either assigned by the vendor specific Management Platform or via Local technician provisioning, depending on the Vendor. Usually a single /29 address is assigned per POTS Linecard though there are MSAN vendors which may require 2 of the /29 subnet IP addresses per POTS card (the /29 subnet is large enough to support enough IP addresses for any individual MSAN POTS card configuration including the default gateway address used on the A4 Leaf switch port).

The A4 Platform is not involved with the MSAN /29 subnet POTS card provisioning other than providing connectivity for the MSAN Management VLAN and the IP Voice VLAN (100) and assignment of a /29 default Gateway IP address for the Leaf switch Port from the same subnet as the Leaf Port connected MSAN POTS Cards.

Once VLAN 100 on the MSAN Uplink is configured and assigned to a VRF it is then up to the A4 Leaf/Spine Fabric to facilitate the import of Route targets for the Voice Service Area (VoSA) and export route targets for the MSAN connected Leaf Port provisioned /29 IP Addresses utilizing Multi-protocol BGP (MPBGP).

See section 3.4.5 for a description of A4 POD MPLS Backbone and BGP Overlay with support for VRFs which support the Voice VPN as shown in ‘Figure 19 A4 POD backbone connection'.

From a security perspective the current MSAN solution provided by the BNG utilizes a split horizon approach such that the VRF forwards traffic only to the northbound interfaces preventing traffic from other members of VLAN 100 from reaching the MSAN. The MSAN Concept paper indicates that IP Access Lists should be configured but this is not currently implemented on the BNG and so is for future study/specification.

The QoS configuration is such that SIP packets are a high enough priority that there is no loss nor significant delay introduced by the switch Fabric. Specific QoS configuration is to be provided by DT (Andreas Zimber).

4 Deutsche Telekom Business Services RD (Remote-Device)

This chapter explains how Remote Devices (RDs) can be connected to the A4 platform instead of the BNG platform. This is important because the BNG platform is becoming outdated and will soon be phased out. By connecting RDs to the A4 platform, we can avoid purchasing more BNG ports for new enterprise customers. This document focuses on connecting RDs through a fiber P2P connection, which we call "RD fiber". The goal is to migrate all RD connections to the A4 platform before the BNG platform becomes obsolete.

Connecting RDs to the A4 platform has several benefits, including the ability to make business-driven decisions about how to connect to the internet, offloading service creation from the BNG to the A4 platform, and acknowledging the end-of-life of the BNG components.

	Connected to	Connection	IM Service	New RD	Migrated RD
RD fibre	dBNG	PtP fibre 1GE/10GE	L2 VPN	x	x
			L3 VPN	x	x
	sBNG	PtP fibre 10GE	L2 VPN	x	x
RD copper	dBNG	VDSL/SHDSL via MSAN	L2 VPN	x	x
			L3 VPN	x	x

Figure 49 RD-Fiber and RD-Copper

Figure 49 RD-Fiber and RD-Copper provides an overview of the different RD use cases. The term "New RD fibre" refers to the production of new RD fibre access, while "Migration RD fibre" refers to the ability to migrate existing BNG-based fibre RD access to the A4 platform. It is assumed that the "New RD fibre" use case is a precondition for the "Migration RD fibre" use case.

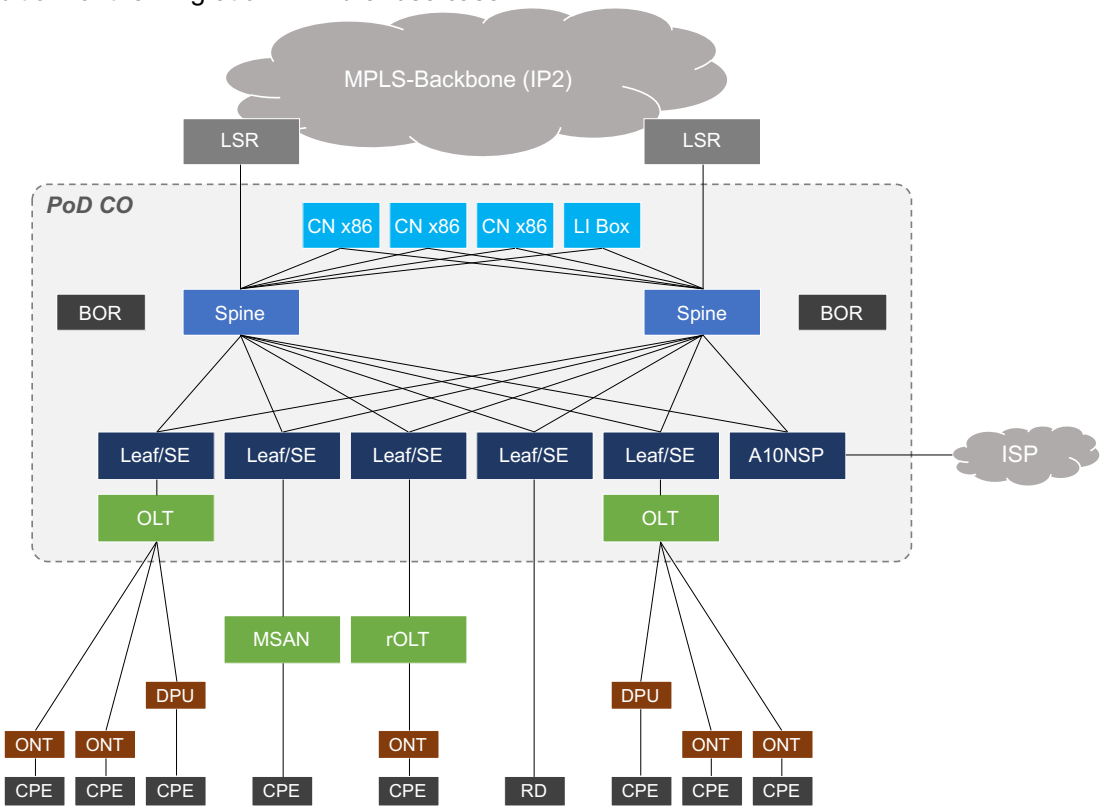

Figure 50 RD-Fiber and RD-Copper in A4

In the "New RD fibre" use case, RDs are directly connected to the A4 platform through a fiber P2P connection. This allows for more efficient and cost-effective management of RD connections. In the "Migration RD fibre" use case, existing BNG-based fiber RD access is migrated to the A4 platform.

RD Fiber (and Copper) support the following Deutsche Telekom Layer2 and Layer3 Business-Services (IM = Individual Market):

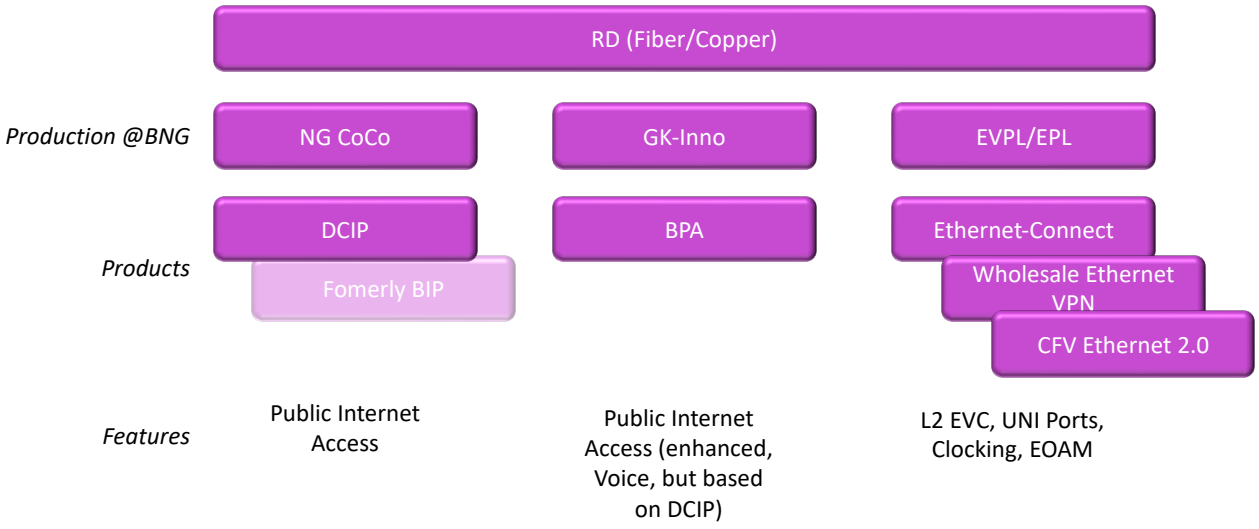

Figure 51 RD-Fiber and RD-Copper Business-Services Overview

4.1 Remote Devices

Today a set of RD's (RD is a network termination device at customer premises, like a managed CPE) is qualified and introduced in the market (Figure 6). All these devices support the same core features and must be supported by A4 to allow seamless A4 introduction and later on potential migration from BNG to A4.

- ADTRAN NV4660 (RD1G-FSV)
- Huawei ATN910I-D (RD1G-F)
- ELCON BIG2862 (RD1G-FSV)
- Huawei ATN910B-D (RD10G-F)
- ELCON BIG4862 (RD10G-F)

REMOTE DEVICE (RD)	1G RD ADTRAN NV4660	1G RD albis-elcon BIG2862	1G RD albis-elcon BIG2862 WAN: Fiber only	1G RD HUAWEI ATN910I-D
				
Supported WAN interfaces	SHDSL/(V)VDSL/Fiber 1G	SHDSL/(V)VDSL/Fiber 1G	Fiber 1G	Fiber 1G

Figure 52 RD-Fiber and RD-Copper 1GE RD's

REMOTE DEVICE (RD)	10G RD albis-elcon BIG4862	10G RD HUAWEI ATN910B-D
		
Supported WAN interfaces	▪ Fiber 10G (SFP+) ▪ Fiber 1G (SFP)	Fiber 10G

Figure 53 RD-Fiber 10GE RD's

The core features are:

- Line rate Layer 2 switching (MEF)
- Line rate Layer 3 IPv4 and IPv6 routing
- H-QoS upstream and downstream
- L2 EVPL and L3 Services
- WAN Interfaces (RD Uplink):
  - SDSL
  - VDSL2
  - 1GE Fiber (SFP based)
  - 10GE Fiber (SFP based)
- LAN Interfaces
  - 1GE RF45
  - 1GE SFP
- Synchronization-as-a-Service
  - PTP 1588
  - SyncE
- DC or AC

4.2 RD as of today - BNG

The RD is shipped pre-configured and boots up with the necessary settings. Once the RD comes online, it receives its full configuration from the DTAG Management system. This configuration is temporary and must be downloaded every time the RD is restarted.

DTAG provides different service types for L2 and L3 services, with the service creation process separated between the BNG platform and the RD. For L3 services, downstream service creation happens on the BNG, while upstream service creation happens on the RD. For L2 services, both downstream and upstream service creation happens on the RD, with the BNG only serving as a connecting device.

- For L3-services (DCIP, SIP-Trunk 1.5, cPBX 1.5) the “downstream”-service creation is on the BNG (incl. some security mechanisms in Upstream direction), the “upstream”-service creation is on the RD.
- For L2-services (EC Connect 2.0, WEV 2.0, CFV EthernetConnect 2.0) the service creation in both “downstream” as well as in “upstream”-direction is on the RD; the BNG only acts as “connecting” device.

4.2.1 RD Fulfillment

Several systems are involved in DTAGs RD Fibre@BNG fulfilment process, including the Service Management (SMK), RMK RD, RMK Access, RMK Transport, and PFS. SMK orchestrates the fulfilment process, while RMK RD manages the RD devices at customer locations. RMK Access is responsible for managing the MSAN and BNG configurations, with the configuration of the BNG happening through JunosSpace. RMK Transport handles the logical resources for services provisioned on top of DCIP/business accesses. Finally, PFS is DTAGs platform control server, handling session initiation requests and providing access profile settings based on policy rules and decisions.

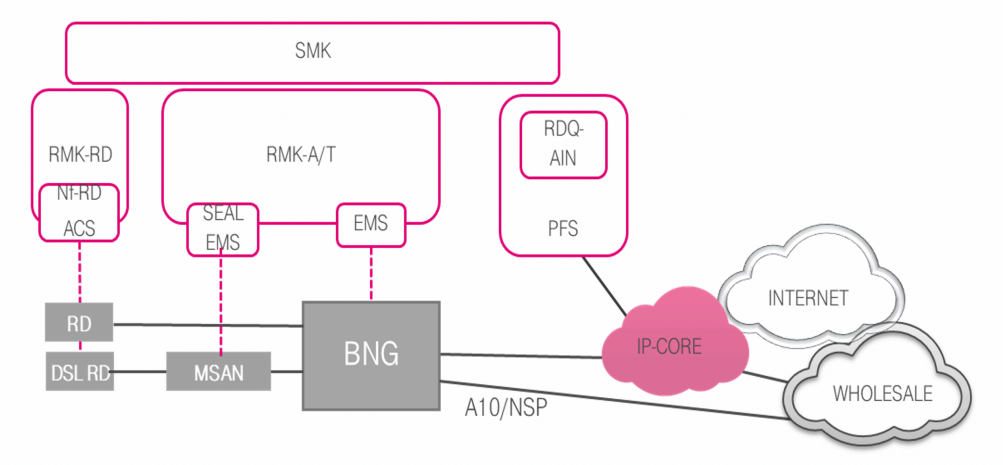

Figure 54 RD Provisioning and Operations Components

- **SMK:** The service management steers the overall fulfilment process for RD Fibre@BNG in OSS-IT. It uses so called Production Plans to orchestrate other IT systems (e.g., the RMKs, Workforce, …) which are needed for producing the requested access/service.
- **RMK Access:** RMK Access is the master for today’s RD Fibre@BNG (DCIP/BPA, EVPL) RFS AccessLines and it is managing the MSAN & BNG configurations. The configuration of the BNG is realized via the Element Management System (EMS) JunosSpace provided by the vendor Juniper. The configuration of the MSANs is realized by the individual vendor specific Element Management System (EMS). In order to adapt/hide the vendor specific functions an abstraction layer SEAL (Standard Element Abstraction Layer) is used towards the OSS Systems. The information for the

business processes ‘Operation Support & Readiness’ (OS&R), ‘Assurance’ (ASR) und ‘Fulfilment’ (FF) are transferred via SEAL.

- **RMK Transport:** RMK Transport responsible for management of logical resources (e.g. IP Addresses) of services (like Ethernet Connection and IP Connection). The services are provisioned on top of the DCIP/business accesses.
- **PFS:** PFS is DT's platform control (network control) which is both AAA and Policy Control Server. PFS receives session initiation requests (currently by BNGs). Furthermore, PFS is the policy controller providing access profile settings based on policies rules and decisions.
- **RMK-RD (NF-RD):** RMK-RD has the task to ‘manage’ the Remote Device (RD) deployed as network termination at customer premises.

4.2.1.1 Netfactory-RD

RMK-RD manages the Remote Device at customer premises which performs various functions. This is done through the Net Factory RD system based on Xyna Factory by vendor GIP. With the transition to dynamic handling of parameters for GK Access, not only the configuration of the service creation point at BNG is dynamic but also the complete autoconfiguration of RDs, even for those RDs that are connected to a "static" provisioned BNG access. There are no changes planned for NF-RD in this context of RD fibre@A4.

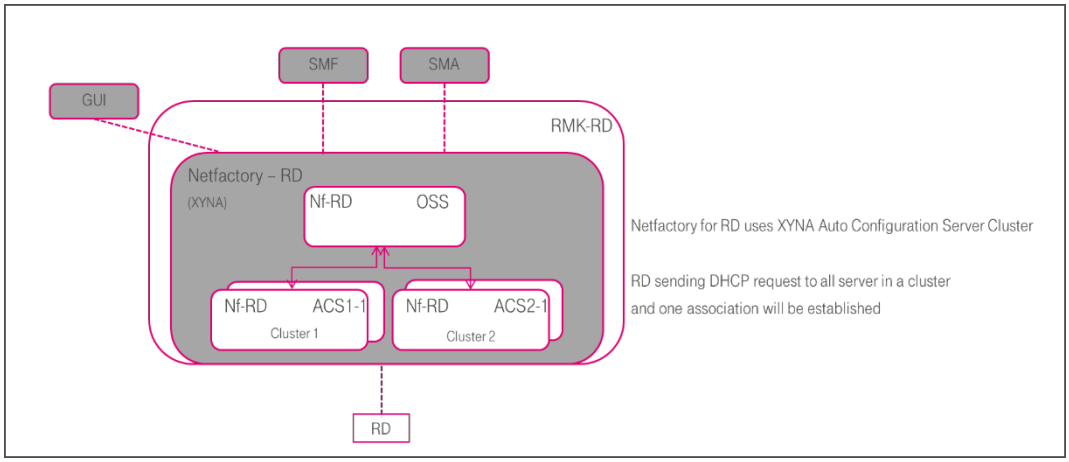

Figure 55 Netfactory-RD

4.2.1.2 Dynamic Production (dBNG)

Before being sent to the customer, the RD is given a basic configuration in a DT service center. This is a general configuration that is not specific to the customer. After saving the basic configuration, the "Factory RD" becomes a "Telekom RD" through the initialization process. In this state, the RD can only establish a PPPoE session in VLAN 7 and get an IP address via DHCP in VLAN 3.

Keys and credentials are applied in both directions (from RD to RMK-RD and from RMK-RD to RD) to uniquely identify the RD when it connects to any logical BNG port, and to set up an SSH session for further configuration steps.

The LINE ID is a universal key used to identify the logical connection from the network termination point (APL) to the subscriber circuit endpoint on the BNG.

The format is standardized as Countrycode.Carriercode.LineCode (DEU.DTAG.1A2B3C4E), which is created and managed by RMK-A as part of the access profile.

The PPPoE session is used to identify the line and customer's product/service, and also to monitor that the RD is functioning (LCP). If the PPPoE session terminates, both the logical port on the BNG and the RD will lose their configuration.

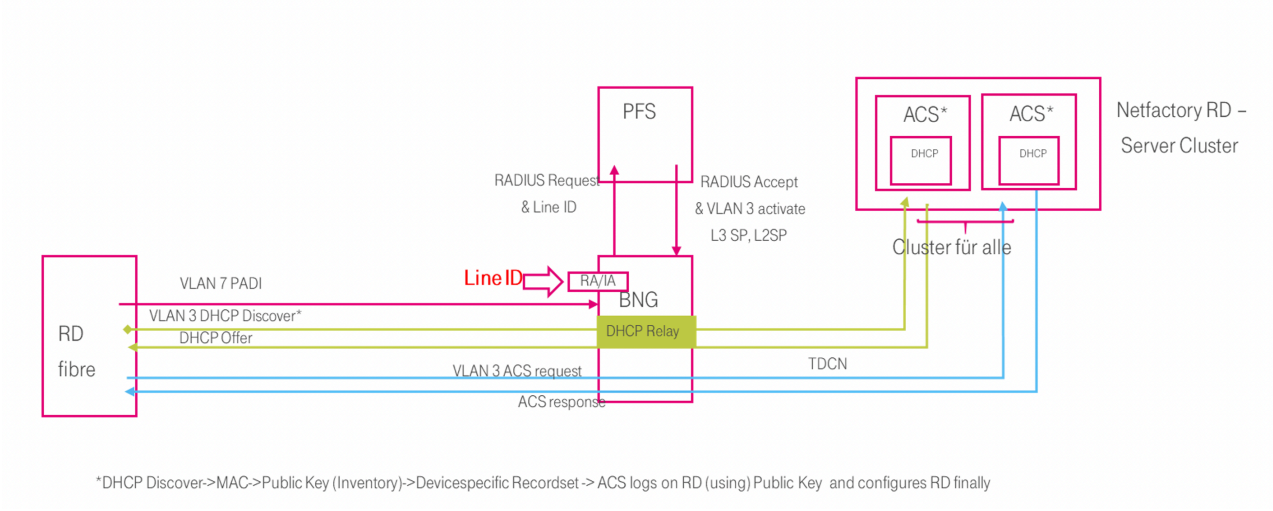

Figure 56 RD Dynamic Production (Management and Control)

After the RD powers up, the BNG receives a PADI message and sends a RADIUS request with the LINE ID to the PFS. The PFS has the parameters for the corresponding access and data of the right product, which are then sent back to the BNG via another RADIUS message.

Meanwhile, the RD sends a DHCP request to the BNG in VLAN 3, which is forwarded to all Netfactory instances via DHCP proxy and TDCN after a successful PFS lookup. Based on the DHCP request, the Netfactory recognizes the RD.

The NF RD checks the RD's state, reads/matches inventory data, checks whether FW/BL/BaseConfig-Updates need to be executed, and then triggers the RD to fetch a configuration file which is dynamically created to fulfill the services defined by the LineID. Then service creation at RD is done and IM services are completely active.

The VLAN range for IM (Individual Market) services is 3001-4001 (so-called "T-Tag"), which is reserved for the actual services of the product. (Note: for DCIP, it is always 3001).

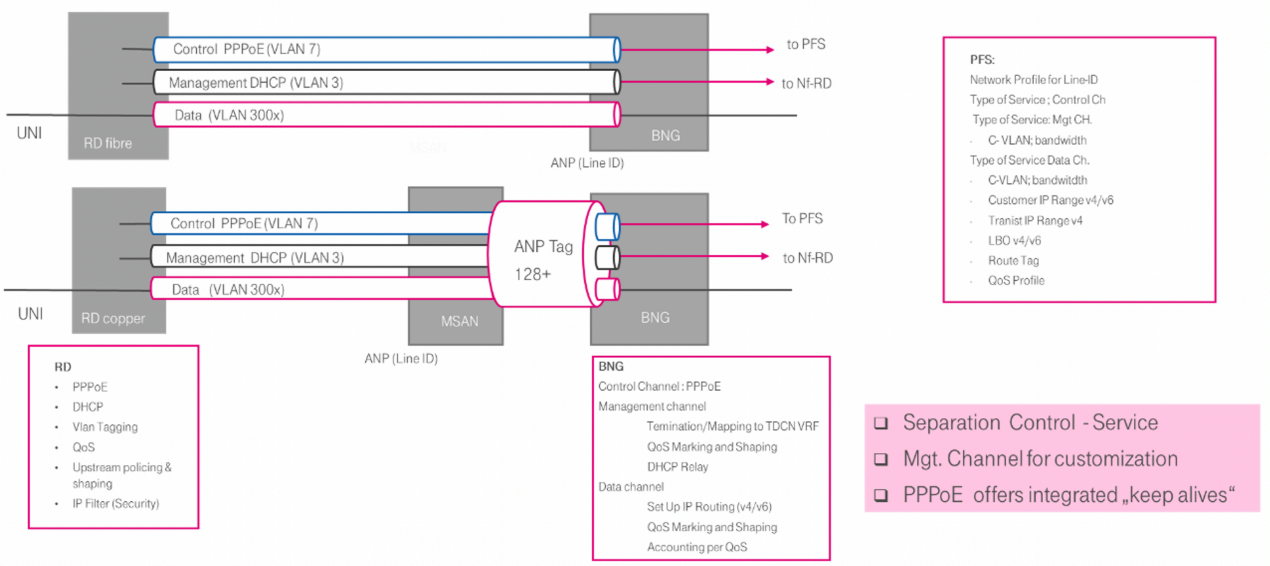

Figure 57 RD Dynamic Production (Management and Control)

4.2.2 L3-Services - DCIP

DCIP (Deutschland LAN IP-Connect) covers basically all L3 Business Service produced on the BNG. It appears in the documentation also as BIP or BPA. All L3 Services are based on DCIP and it is good enough for L3 to have a detailed analysis only on DCIP.

4.2.2.1 DCIP High-Level Capabilities

Find here the basic capabilities of the L3 DCIP services:

- Deployment of a UNI for customer connectivity
- VLAN-Tagging between RD and BNG
  - C-VLAN ID#7 für PPPoE (Control Channel)
  - C-VLAN ID#3 für DHCP (Management Channel
  - C-VLAN ID# i.e. 3001 für Data Channel (public Internet)
- PPPoE Client for the setup of the control channel
  - CHAP based authentication
  - Defined PADI behavior during nonresponsive network
- DHCP client fort he RD management network establishment (TDCN- Access)
  - Defined DHCP- Discovery/Release behavior
- QoS Support
  - Mapping to DTAG QoS scheme
  - Remarking IP-Precedence (upstream and downstream)
  - Shaping
  - 4 WF and Priority Queue for voice
- OAM Support
  - Service monitoring
  - Test and diagnosis
- SNMPv3
- Firmware Update Service
- RD configuration service
- Security Support (Filter (IP und Port), VRFs, SSH, etc.)
- Routing
- Redundancy such as VRRP
- ACL

4.2.2.2 DCIP and RD Addressing

As with IP-based mass-market products, the DCIP product provides access to the public internet. While with IP-based mass-market products, the customer's home gateway is assigned an IPv4 and/or an IPv6 address, with DCIP, the customer can be provided with several IPv4 and/or IPv6 address ranges (so-called IP aggregates or prefixes) whose traffic is routed through the Deutsche Telekom IP network. In this case, a DCIP customer can use both provider-independent (PI) address ranges and provider-aggregated (PA) address ranges, which are briefly explained below:

- PI addresses are IP addresses that do not originate from the AS3320 address range of Deutsche Telekom. The customer has either registered these IP addresses with RIPE or they come from the IP address range of another service provider who has registered them with the customer.
- PA addresses are IP addresses from the AS3320 address range of Deutsche Telekom. The corresponding PA prefixes are assigned to the BIP customer by Deutsche Telekom and registered with them.

Regarding the IP prefixes that can be provided to the customer (UNI/LAN), the DCIP product is implemented with the following scope of services:

- A maximum of 20 IP prefixes are supported. Of these, a maximum of 10 prefixes can be used for IPv4 and a maximum of 10 prefixes for IPv6.
- The respective 10 IPv4 and IPv6 prefixes can be PI and/or PA address ranges of the corresponding IP protocol version.

For PA address ranges, the following IP aggregates are assigned to the customer by default, which can also be provided as smaller or larger address ranges upon customer request:

- For IPv4: /28 prefixes (14 host addresses)
- For IPv6: /48 prefixes (with this, the customer can create 2^16 IPv6 /64 networks, with each /64 network supporting 2^64 host addresses). If the BIP customer requires a larger prefix than /48, they must demonstrate the additional address requirements to RIPE.

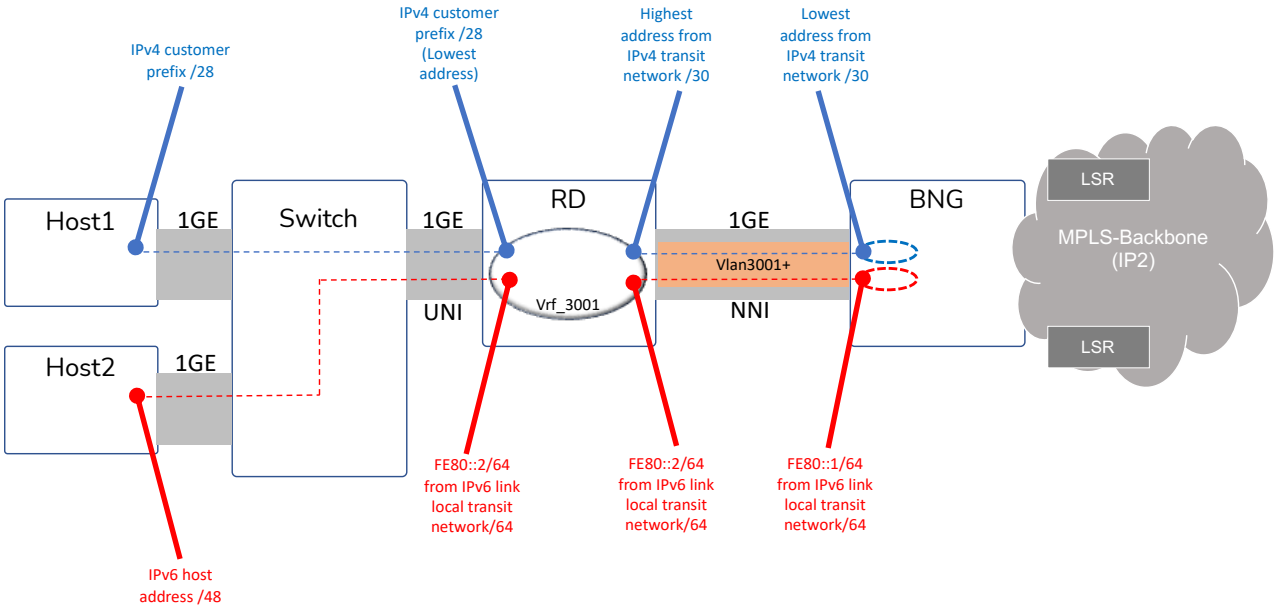

Figure 58 RD DCIP IP-Addressing

NOTE: DCIP is not a L3VPN service or product, it may be terminated in a VRF on the BNG, but the customer gets simple public IPv4 and v6 internet access.

4.2.2.3 DCIP Routing

Find here are diagram regarding the routing from BNG to the RD and to the customer UNI. It is all static routing upstream and downstream. Downstream on the RD we assume a default route.

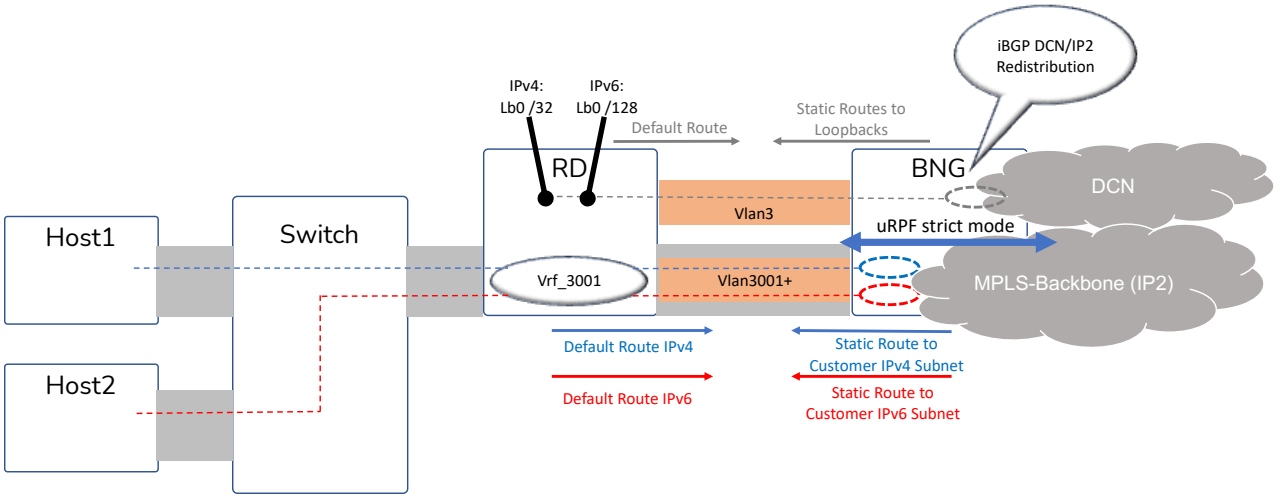

Figure 59 RD DCIP Routing

NOTE: There is contradicting information about the customer prefix level, in some documents they are /28 in some other they are /29 for IPV4. The same is for IPv6 with /64 versus /48. This doesn't affect the architecture since this information is automatically provisioned and should not be an issue for A4. The transport links between RD and BNG are /30 and /64 in all the documentation.

4.2.2.4 DCIP Ethernet-OAM

For DCIP Ethernet-OAM (Y.1731 with CCP: Connectivity check Protocol) is used only for Service VLAN (3001) Continuity Check between the RD and the BNG by processing Continuity Check Messages (CCM). IEEE 802.1ag is largely identical with ITU-T Recommendation Y.1731. Both the RD and BNG are representing a Maintenance association End Point (MEP). The MEP run a constant reachability heartbeat in vlan 3001, so that actions can be taken when a defect is recognized. In the case of DCIP the BNG (and RD?) will withdraw its static routes (the RD routes and prefixes), so that the reachability of a specific RD is withdrawn from the internet routing table. There is no other application for EOAM in conjunction with DCIP and L3.

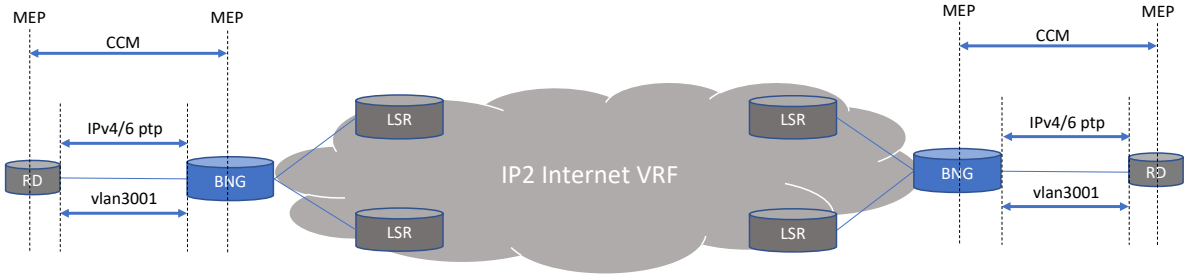

Figure 60 DCIP Ethernet-OAM

4.2.2.5 DCIP Quality of Service

The following measures are applied regarding QoS for DCIP. On the RD:

- Mapping of the subscriber QoS marking to the DTAG marking scheme (802.1p and DSCP/IP-Precedence) upstream and downstream
- Remarking of the subscriber QoS marking to the DTAG marking scheme (802.1p and DSCP/IP-Precedence) upstream and downstream
- H-QoS Shaping upstream towards the BNG with:
    - 4 WFQ (Low Delay, Low Loss, Best Effort und Management/Control Traffic) plus Strict Priority Queue für Voice with rate limiter

On the BNG

- H-QoS Shaping downstream/egress towards the RD with:
    - 4 WFQ (Low Delay, Low Loss, Best Effort und Management/Control Traffic) plus Strict Priority Queue für Voice with rate limiter
- H-QoS Shaping upstream/ingress from RD with:
    - 4 WFQ (Low Delay, Low Loss, Best Effort und Management/Control Traffic) plus Strict Priority Queue für Voice with rate limiter

DTAG provides standard profiles for DCIP:

Class	IP-Prec. UNI	IP-Prec. Uplink	Profile 0 BE-Only	Profile 1 Standard	Profile 2	Profile 3	Profile 4	Profile 5
Multimedia	4	4	0%	45%	90%	0%	45%	0%
Critical	4	3	0%	45%	0%	90%	0%	45%
Best-Effort	0,1,2,6,7	0	95%	5%	5%	5%	50%	50%
Control	N/A	6	5%	5%	5%	5%	5%	5%
SUM	N/A	N/A	100%	100%	100%	100%	100%	100%

Figure 61 DCIP QoS Profiles

Whereas the class:

- “Multimedia” means - Low-Delay
- “Critical” means - Low-Loss

Qualitätsklasse	One-Way Delay [ms]	Jitter [ms]	Loss [%]
Low Delay	25	5	0,1
Low Loss	40	-	0,01

Figure 62 DCIP Low Delay and Loss SLA

NOTE: A subscriber may have manually adopted specific and deviating from the DTAG profiles QoS settings.

4.2.2.6 DCIP SLA's

There are defined some more SLA's regarding throughput and QoS:

IP-Throughput [M Bit]						
Bandwidth	**50M**	**100M**	**200M**	**300M**	**600M**	**1000M**
Packet Size [Byte]						
128	40.441	80.882	161.765	242.647	485.294	808.824
256	45.076	90.152	180.303	270.455	540.909	901.515
384	46.684	93.367	186.735	280.102	560.204	933.673
512	47.500	95.000	190.000	285.000	570.000	950.000
1024	48.740	97.481	194.961	292.442	584.884	974.806
1526	49.148	98.296	196.592	294.889	589.777	982.962

Figure 63 DCIPThroughput per Packet Size

Delay [ms] einfach	
Low Delay	25
Low Los	40

Figure 64 DCIP Delay

Jitter [ms]	
Low Delay	5
Low Los	-

Figure 65 DCIP Jitter

Packet Loss [%]	
Low Delay	0,1
Low Los	0,01

Figure 66DCIP Loss Rate

4.2.2.7 DCIP Probe Based Monitoring

The "DCIP_Detailed Design_2014 01" refers to various methods of probing for DCIP and the RD. The assumption for the moment is, that probing is completely overlay and does just make use of CFM and i.e., PPPoE keep alives.

4.2.2.8 DCIP IPFIX Arbor DDoS prevention

TBD

4.2.2.9 DCIP redundant RD's

DTAG offers to their RD customer a redundant access with 2 RDs per customer site. A single RD has always only one uplink, there no dual homing offered for RD. Redundancy is always implemented based on 2 different RD's with 2 different access line being terminated at 2 different BNG's.

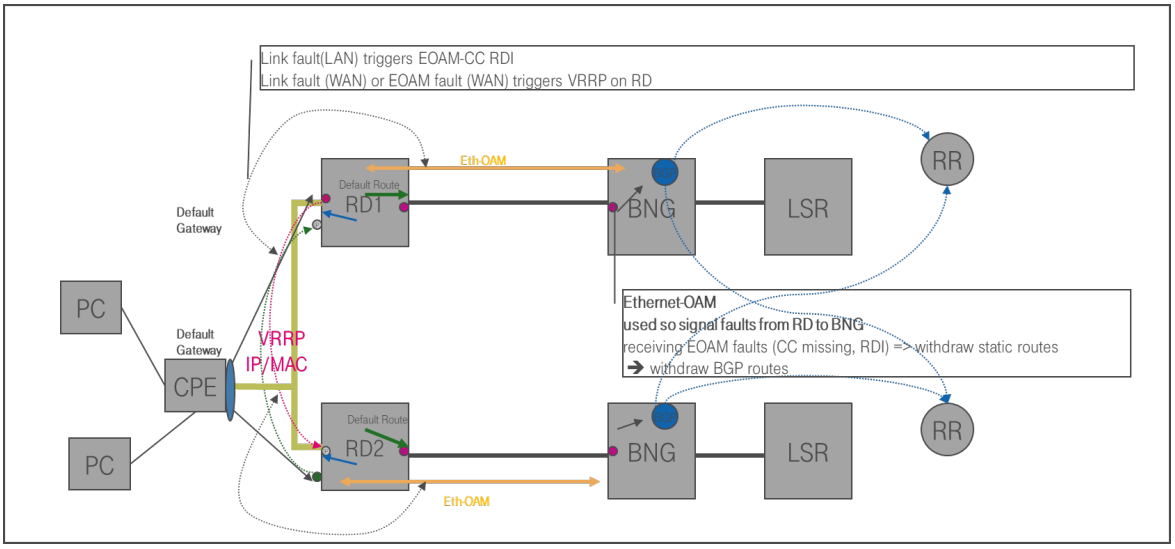

Figure 67 Redundant RD

If the primary connection fails, the ETH-CC (Ethernet-OAM) system implemented by RD and BNG will detect it. In order to enable a failover to the secondary connection, RDs must exchange status information via the customer LAN using the VRRPv3 protocol and turn down the VRRP Master on ETH-CC down event.

However, if the UNI on the RD is disrupted, VRRP implementation on the RD triggers the appropriate actions, selecting a new VRRP master and setting the RDI bit within the ETH-CC frame towards the BNG. Consequently, the BNG detects an error in the ETH-CC session, sends an alarm and removes the routing entries for the customer's subnets.

Similarly, if the BNG detects an error on the still-active line, it sets the RDI in the ETH-CC towards the RD, which triggers the RD to change VRRP mastership towards the other RD. In the L3 redundancy setups (Hot stand-by or "Active/Passive"), in either case of failure, whether the RD or BNG can indicate via RDI or if also connectivity is disrupted, automatic failover to the "secondary" path is triggered both on the BNG and the RD.

Finally, when the primary access connectivity is restored, the connectivity is automatically "switched-back" to the primary access by BNG and RD independently.

4.2.2.10 DCIP L3 Lawful Intercept

The Lawful Interception (LI) architecture is depicted in the following Figure. The LI environment is used to fulfill legal requirements for the extraction of user data based on a court order. A common LI architecture for all public IP services is used for the BNG. The order to extract traffic is issued from the network platform perspective via the GFOS or LIMS system and is passed to the platform controller via the RMC Legal. The RMC legal transfers the line ID that represents the monitored access. The identification of the line ID is carried out in GFOS/LIMS and is passed as a key for the order to the platform controller.

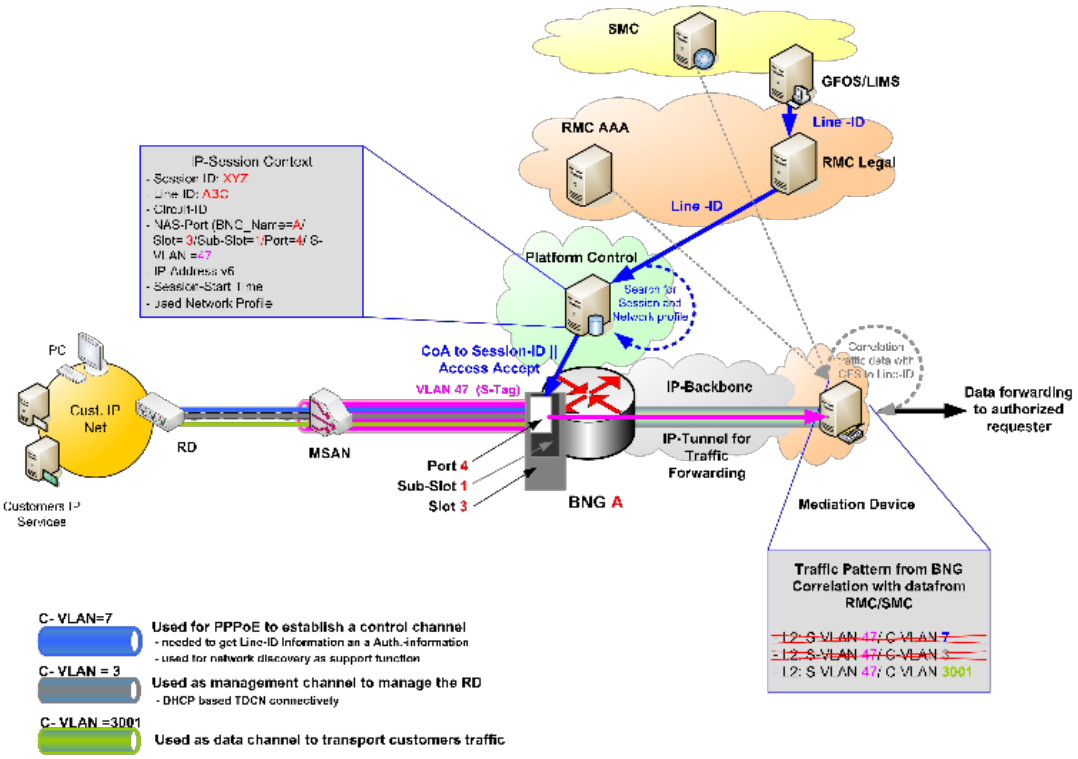

Figure 68 DCIP L3 Services – Lawful Intercept

The platform controller determines the session (session ID) that is held at runtime in the session context of the platform controller for the given line ID in order to address a change to an existing session. In any case, the measure is stored at the network profile of the line ID until revoked, as it must be ensured that the extraction can also be restored after the termination of an existing session with the reinitialization. The platform controller activates the extraction of traffic on the BNG using the RADIUS protocol, for an existing session using a so-called Change of Authorization (CoA) request for the existing session or directly with the establishment of the connection.

In the case that traffic is to be executed for an existing session, the activation of the extraction must be possible without interrupting the session. The CoA process allows for uninterrupted activation.
The extraction of traffic applies to the entire virtual customer interface, which is represented by an S-VLAN on a BNG interface and is documented in the session. Technically speaking, the extraction of traffic duplicates all IP traffic in upstream and downstream direction, which is then forwarded in addition to the "monitoring port" of the BNG.

The transport of the monitored traffic from the BNG to the Mediation Device (MD) as a transfer point for further processing of the traffic is done via a UDP-based IP tunnel that is already preconfigured between the BNG and Mediation. On the Mediation Device, the extracted traffic can then be filtered for relevant portions as necessary. This may be particularly necessary if multiple services are provided over an RFS access (represented by the line ID). If this is the case, the Mediation must be able to filter the traffic at the C-VLAN level, as it cannot be ruled out that both a BIP (e.g. in C-VLAN 3001) and an L2VPN service (e.g. in C-VLAN

3002) are produced within the S-VLAN. If this is the case, the traffic on the Mediation Device may need to be separated. In order for the Mediation to accomplish this, it may require additional information from the SMC or the RMC AAA

4.2.3 L2-Services – EVPL and EPL

RD fulfillment is the same as for DCIP, with vlan 7 for PPPoE and vlan 3 for DHCP setup. Instead of L3 on vlan 3001, vlan 3001 is now used for L2 services, so no IP configuration.

4.2.3.1 EVPL High-Level Capabilities

DTAG provides a family of L2 connectivity services for their customers, some of them grown over decades, i.e. Ethernet over SDH. The idea is/was to concentrate all L2 products under one RD production model, the main 3 EVPL Products are:

- EthernetConnect2.0: EVPL/EPL, UNI only
- CFV-Ethernet 2.0: Point2Point EPL only
- Wholesale Ethernet VPN 2.0: UNI Hub&Spoke to NNI (Wholesale)

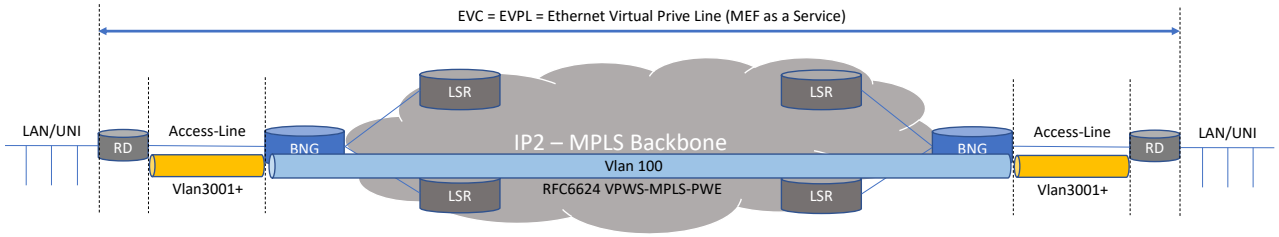

Figure 69 EVPL Transport

Each EVC maps to an EVPL or EPL and is Point2Point only. There is no bridge or MAC-learning, packets are forwarded transparently (please refer to chapter: EVPL Protocol Transparency)

Between the BNG and the RD the service vlan starts with 3001 (up to vlan 4000). Between the BNGs the EVC is mapped to a RFC6624 VPWS Tunnel and a normalisation vlan 100. Vlan tags are swapped always on the BGN (down and upstream), vlan tags will not be transported transparently.

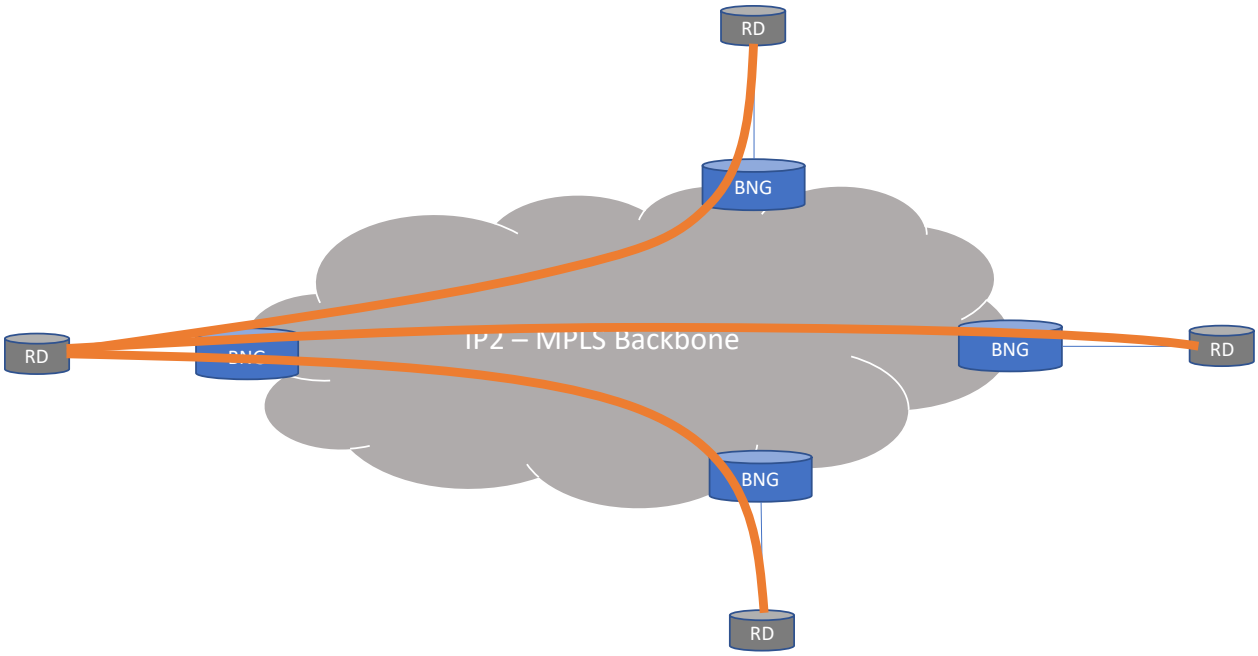

Figure 70 EVPL Hub&Spoke Topology

Based on Point2Point EVPL Services any Topology can be build, such partial/full-mesh, hub&spoke or a ring. It is important to differentiate between UNI and NNI. UNI based EVPL Services are produced on the dBNG (Dynamic Production Model) and NNIs are produced on the sBNG (Static Production Model) whereas:

- UNI = N x physical Ethernet Port on RD with each one or multiple EVCs/EVPLs
  - Max. 200 EVCs per RD are supported on 1GE (Uplink).
  - Max. 1000 EVCs per RD are supported on 10GE (Uplink).
- NNI = N x physical Ethernet Port on RD with each one or multiple EVCs/EVPLs
  - Max. 1000 EVCs per RD are supported on 10GE (Uplink).

4.2.3.2 EVPL Transport

The transport of the EVCs/EVPL services is based on a Juniper specific PWE transport implementation of VPWS (Virtual Private Wire Services), RFC 6624: https://www.rfc-editor.org/rfc/rfc6624.html

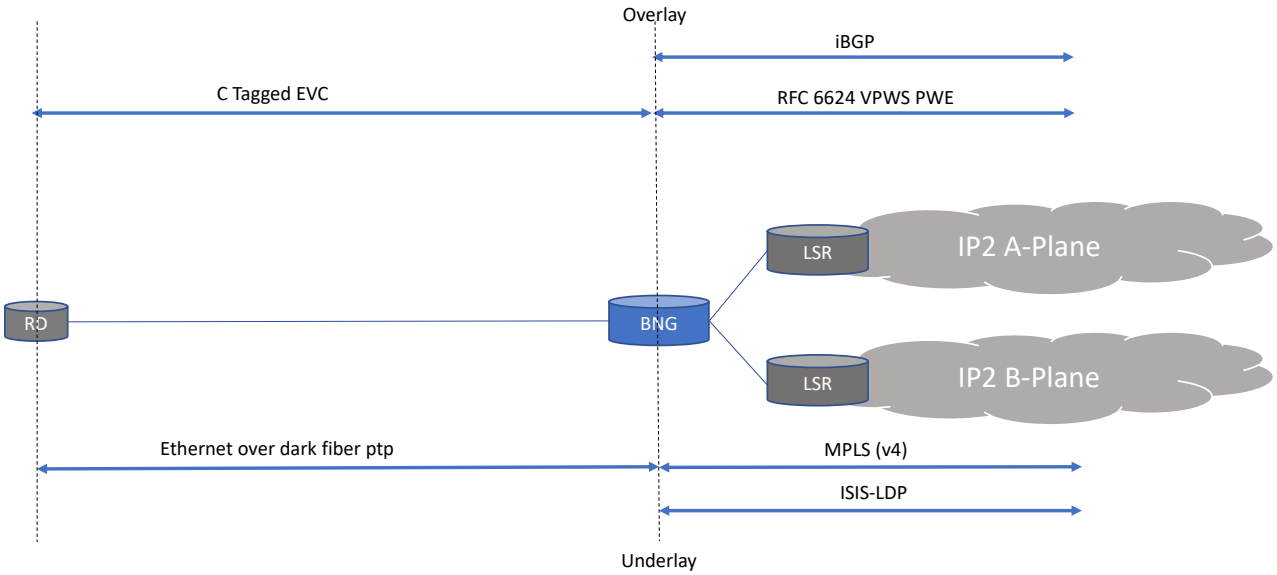

Figure 71 EVPL VPWS Transport

RFC6624 defines iBGP for VPWS auto-discovery and PWE signalling. Auto-discovery enables automatic discovery of VPWS neighbours and signalling is required to map a peers service endpoint (virtual L2 interface) to a PWE label.

RFC6624 is based on RFC4761 which specifies so-called VPLS, Virtual Private LAN Service. RFC6624 doesn't describe further details on the iBGP implementation of VPWS, but is assumed, Juniper applied the same measures as in RFC4761, but just point2point with no MAC-Learning. Find here a VPLS configuration example: https://www.juniper.net/documentation/us/en/software/junos/vpn-l2/topics/example/vpls-bgp-configuring-detailed-solutions.html

General BNG configuration (always on):

- IP/MPLS on the backbone links
  - LDP
  - IGP
- Loopback interface
- iBGP VPLS/VPWS
  - local-address (BNG loopback)
  - Address family l2vpn
  - iBGP neighbours (RR)

The BNG requires the following parameters for the VPWS setup:

- RD facing Vlan-tagged Ethernet interface
  - MTU
- Routing instance
  - instance-type (RD facing)
  - EVC Attached ethernet interface
  - Route-distinguisher RD
  - vrf-target target (import/export)
  - site range
  - site name
  - site identifier

The next chapter describes how the VPWS PWE configuration for the BNG is derived and created.

4.2.3.3 EVPL VPWS PWE Provisioning (BNG)

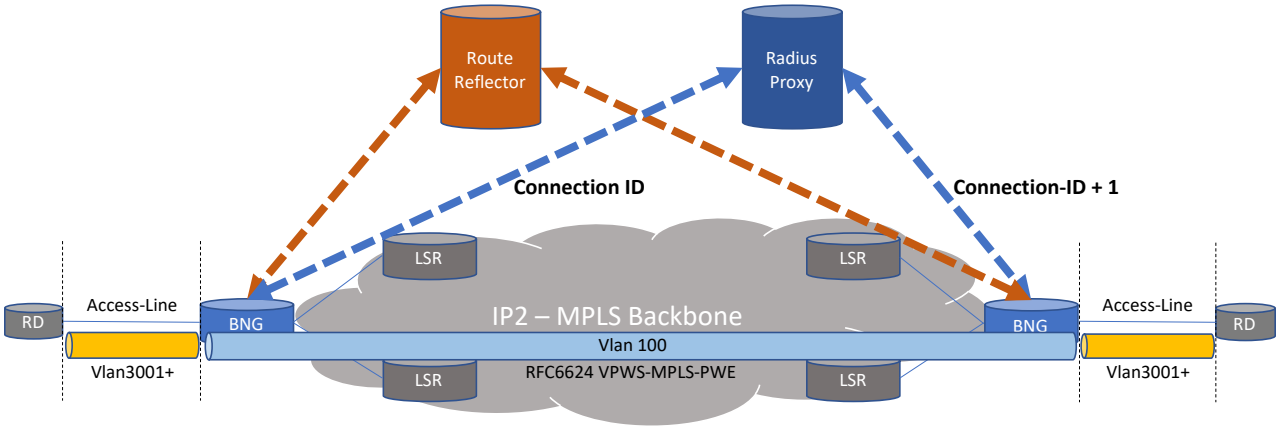

Figure 72 EVPL VPWS PWE fulfillment

The sequence of the BNG service creation is the following:

- BNG receives PPPoE request from RD (with Line-ID) and forwards it to the Radius-Proxy
- Radius-Proxy sends a service string with all EVPL parameters to the BNG (refer to table below)
- The BNG resolves the Connection-IP (Endpoint-IDs with Source-ID and Target-ID) into Junos configuration.
    - Parameters such as Route-Target and Route-Distinguisher
        - Instance-type (RD facing)
        - EVC Attached ethernet interface
        - Route-distinguisher RD
        - vrf-target target (import/export)
        - PWE-ID
        - site range
        - site name
        - site identifier
    - NOTE: Connection-ID=Source-ID of the service is de-composed (first 2 bytes => Route-Target=Route-Distinguisher, last 2 bytes => Site-ID).
    - NOTE: JUNOS EVPL configuration is a volatile configuration and not copied to the startup config
- The same happens on the peering BNG of that EVPL connection
    - NOTE: VPWS PWE are uni-directional and need to be setup from both side
- Based on iBGP Auto-Discovery and Signaling the EVPL VPWS tunnels are established

Topic	Variable and Definition (Schnittstelle Plattformsteuerung – BNG	Always Present
Servicename	ethp2p	
VLAN-ID (T-Tag)	vid <integer>	Yes
Endpoint Ids	sourceID, targetID <integer> ConnectionID = EndpointID-1 ConnectionID + 1 = EndpointID-2 Both EndpointIDs are given as sourceID and target ID (decimal numbers)	Yes

Service Bandwidth	Service-BW <integer> L2 service-Bandwidth in kbit/s	Yes
Access Bandwidth	Access-BW <integer> L2 Access-Bandwidth in kbit/s Optional Parameter. All services with a given Access-BW use the same Access Bandwidth policer. All Access bandwidth must be the same AccessBWL2. Hirachical Policing with strict priority must be implemented: 1st Priority: Voice, 2nd Priority LowDelay, LowLoss, EthernetOAM, 3rd Priority BestEffort If no AccessBWL2 exists there is no Access Bandwidth Policer	No
QoS Class used and Bandwidth	**BW-LD, BW-LL, BW-VO, BW-BE** <integer> L2 service-Bandwidth in kbit/s Definition of the QoS Class inside the service. not used classes are handled as best effort.	No
Opt-Profile	optpro	No
Description	SDscr	No

Figure 73 EVPL Service Strings PFS to BNG

The connection-ID is associated with the vlan tag between the RD and the BNG plus the BNG interface towards the RD.

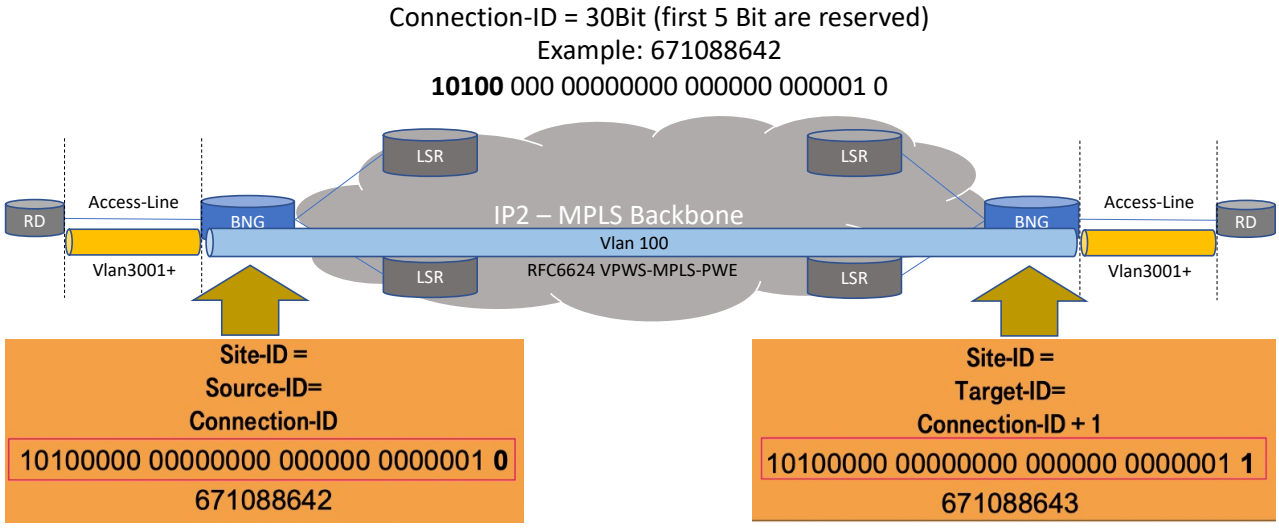

Figure 74 EVPL Connection-ID

Please find here an example on how the VPWS specific parameter are derived from the Connection-ID:

Figure 75 EVPL Retrieving Route-Target and Route-Distinguisher

In the context of dynamic production, the BNG constructs the parameters Route-Target, Route Distinguisher, and Site-ID for the service configuration on the BNG based on the transmitted Endpoint IDs using scripts.

The BNG (Site A) splits Source-ID 671088642 (Connection-ID) into Route-Target 40960 and Site ID 2, as well as the Target-ID into Route-Target 40960 and Site ID 3. The functional condition (plausibility check in the BNG) that both Route-Targets are identical is met.

- Connection-ID=Source-ID 671088642 = 1010 0000 0000 0000 0000 0000 0000 0010
- Connection-ID=Source-ID of the service is de-composed (first 2 bytes => Route-Target=Route-Distinguisher, last 2 bytes => Site-ID).
  - 1010 0000 0000 0000 = 40960
- Target-ID=Connection-ID+1=Source-ID+1

The BNG communicates the L2VPN route with: Site-ID=2, Route-Target=40960, Route-Distinguisher=40960, Nexthop=Loopback address-BNG(A), MPLS-Label to Route-Reflector. The Route-Reflector communicates this information to all BNGs, including BNG(B).

The BNG (Site B) splits SourceID 671088643 into Route-Target 40960 and Site ID 3, as well as the Target-ID into Route-Target 40960 and Site ID 2. The functional condition (plausibility check in the BNG) that both Route-Targets are identical is met. The BNG communicates the L2VPN route with: Site-ID=3, Route-Target=40960, Route-Distinguisher=40960, Nexthop=Loopback address-BNG(B), MPLS-Label to Route-Reflector. The Route-Reflector communicates this information to all BNGs, including BNG(A). Both BNGs now have the complete routing information, and the connection is established.

4.2.3.4 EVPL Protocol Transparency

The RD EVPL subscriber requires a certain level of protocol transparency. I.E. by default any logical and physical Ethernet interface would punt STP packets (based on Destination MAC Multicast address: 01:80:C2:00:00:00) to the control plane and process the packet(s) accordingly. But in the case of a EVPL Service most of the control plane packets should be forwarded, but not all. I.E. CFM and Y.1731 packets should be processed at the edge of the network. Protocol Transparency is defined by MEF with L2CP. Find here the table for DTAGs RD and EVPL services:

L2CP - Protokolle			Comment DT
Cisco - Protokolle	Ethertype/Subtype	Destination Address(es)	
Cisco Port Aggregation Protocol (PAgP)			passed
Cisco Uni Directional Link Detection (UDLD)			passed
Cisco Discovery Protocol (CDP)			passed
Cisco VLAN Trunking Protocol (VTP)			passed
Cisco Dynamic Trunking Protocol (DTP)			passed
Cisco Inter Switch Link (ISL)			passed
allgemeine L2CP-Protokolle (verschiedener Hersteller)			
Per VLAN Spanning Tree Protocol (PVST+)			passed
Spanning Tree Protocol (STP)			passed
Link Aggregation Control Protocol (LACP)	Subtypes: 0x01, 0x02	01-80-C2-00-00-02	discarded
Link Aggregation Control Protocol (LACP)	Subtypes: 0x01, 0x02	01-80-C2-00-00-03	passed
Ethernet Local Management Interface Protocol (E-LMI)	0x88EE	01-80-C2-00-00-07	peered (if enabled); discarded (if disabled)
Ethernet Synchronization Messaging Channel (ESMC)	Subtypes: 0x0A	01-80-C2-00-00-02	peered (if enabled); discarded (if disabled)
Link Layer Discovery Protocol (LLDP)	0x88CC	01-80-C2-00-00-03	passed
Precision Time Protocol Peer-Delay (PTP)	0x88F7	01-80-C2-00-00-0E	passed
Protocol (VDP)	0x8940	01-80-C2-00-00-00	passed
Port-Based Network Access Control	0x888E	01-80-C2-00-00-03	passed
802.3 MAC Control: PAUSE		01-80-C2-00-00-01	discarded
802.3 MAC Control: Priority Flow Control (PFC)		01-80-C2-00-00-02	discarded
802.3 MAC Control: Multipoint MAC Control	Subtype: 0x0002-	01-80-C2-00-00-01	discarded
802.3 MAC Control: Organization Specific Extensions		01-80-C2-00-00-01	discarded
Rapid/Multiple Spanning Tree Protocol (RSTP/MSTP)	LLC Address: 0x42	01-80-C2-00-00-08	passed
Shortest Path Bridging (SPB)	LLC Address: 0xFE	01-80-C2-00-00-2F	passed
Multiple MAC Registration Protocol (MMRP)	Ethertype: 0x88F6	01-80-C2-00-00-20	passed
Multiple VLAN Registration Protocol (MVRP)	Ethertype: 0x88F5	01-80-C2-00-00-0D	passed
Multiple Stream Registration Protocol (MSRP)	Ethertype: 0x22EA	01-80-C2-00-00-0E	passed
Multiple ISID Registration Protocol (MIRP)	Ethertype: 0x8929	01-80-C2-00-00-00	passed
Generic Attribute Registration (GARP)			passed
Connectivity Fault Management (802.1ag) *[D-MAC=01:80:C2:00:00:33 - 01:80:C2:00:00:37 => Freigabe der Domain-Level 3-7			nein, transparent ab MEG-Level 4-7, siehe Leistungsbeschreibung
OAM - Protokolle			
Link-OAM (EFM)			discarded
Link Loss Down Forwarding			nein
Ethernet Frames mit Ethertype			
0x0600)			passed
Internet Protocol, Version 4 (IPv4)	0x0800		passed
Internet Protocol, Version 6 (IPv6)	0x86DD		passed
Address Resolution Protocol (ARP)	0x0806		passed
Reverse Address Resolution Protocol (RARP)	0x8035		passed
AppleTalk Address Resolution Protocol (AARP)	0x80f3		passed
EEE 802.1Q-Tagged Frame	0x8100		passed
MPLS Unicast	0x8847		passed
MPLS Multicast	0x8848		passed
PPPoE Discovery Stage	0x8863		passed
PPPoE Session Stage	0x8864		passed
EAP over LAN (IEEE 802.1X)	0x888E		passed
MAC Security (IEEE 802.1AE)	0x88E5		passed
VLAN Double Tagging (IEEE 802.1ad)			passed

Figure 76 EVPL L2CP Protocol Transparency RD

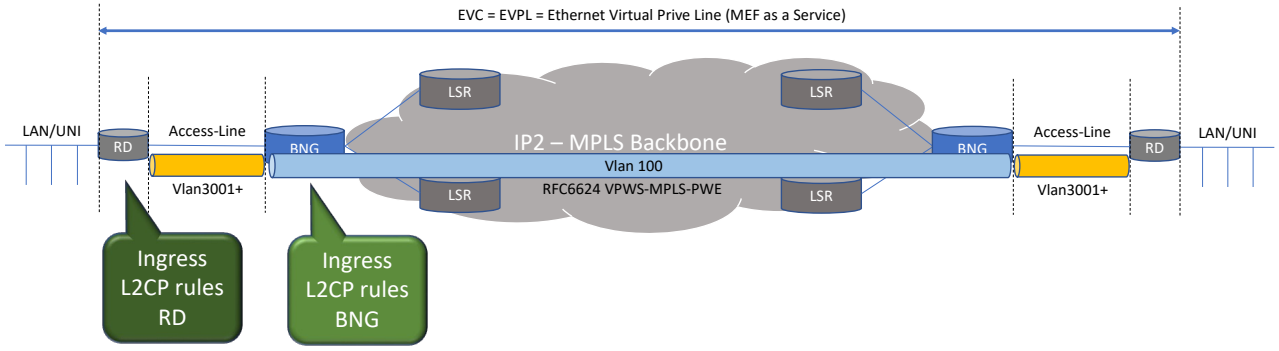

Figure 77 EVPL L2CP Enforcement points

4.2.3.5 EVPL Ethernet-OAM

For EVPL Y.1731 is used end2end between the RD's for continuity checking of the access plus the MPLS PWE. RD–to–RD Continuity Check and Performance Measurements means proactive ETH-CC, ETH-DM and ETH-LM measurements are performed steadily between the two Maintenance association End Points (MEP) configured for each EVC-Endpoint at each RD.If one of the two remote devices (RD) detects through ETH-CC testing that the opposite RD is no longer responding and therefore a local ETH-CC alarm is present, the customer will be informed via Ethernet E-LMI (Ethernet Local Management Interface).

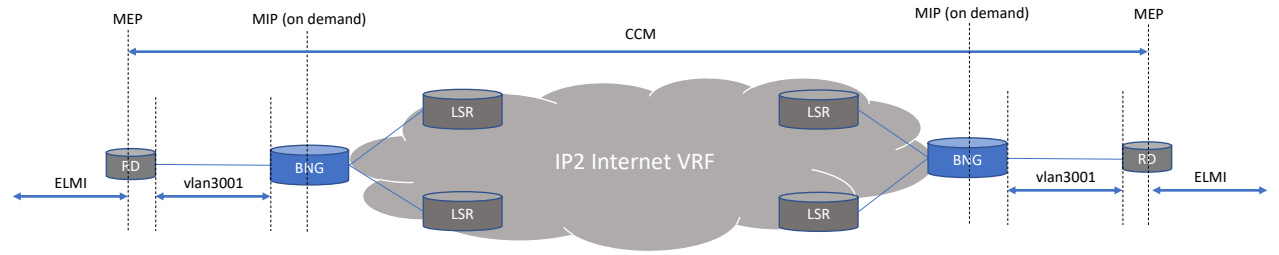

Figure 78 EVPL EOAM MEPs and MIPs

In order to inform the customer which Ethernet service is affected, an identifier is needed within the E-LMI. This identifier is communicated to the RD via RMK-RD, as well as to the customer via BSS.

Furthermore, the following states can be communicated to the customer via E-LMI:

- Ethernet service is added
- Ethernet service is cancelled/deleted
- configured Ethernet service is
    - active -> without alarms
    - inactive -> alarm is present

For test and diagnostics maintenance domain Maintenance Intermediate Points (MIP) can be activated on demand on the BNG.

4.2.3.6 EVPL offered bandwidth

EVC Bandwidth (kbit/s)	Access Bandwidth						
	2.000	4.000	8.000	20.000	980.000	9.840.000	Remark
1.000	X	X	X	X	X	X	
2.000	X	X	X	X	X	X	
4.000		X	X	X	X	X	
8.000			X	X	X	X	
10.000				X	X	X	
12.000				X	X	X	
20.000				X	X	X	
40.000					X	X	
50.000					X	X	
60.000					X	X	
80.000					X	X	
100.000					X	X	
200.000					X	X	
400.000					X	X	
600.000					X	X	
800.000					X	X	
980.000					X	X	
1.000.000						X	

Figure 79 EVPL offered RD Bandwidth

4.2.3.7 EVPL Class-of-Service

An EVC supports one or maximum 4 classes of service, Voice, Low-Delay, Low-Loss and Best-Effort.

- Premium (Voice)
- Priority (Low-Delay)
- Critical (Low-Loss)
- Standard (Best-Effort)

Class-of-Service	Qualität					
	One-Way Frame Delay		One Way Frame Delay Variation		One Way Frame Loss Ratio*)	
Voice (p-bit: 5)	≤ 20 ms	Guaranteed	≤ 2 ms	Guaranteed	≤ 0,1 %	Target Value
Low Delay (p-bit: 4)	≤ 25 ms	Guaranteed	≤ 5 ms	Guaranteed	≤ 0,1 %	Target Value
Low Loss (p-bit: 3)	≤ 40 ms	Target Value-			≤ 0,01 %	Guaranteed
Best Effort (p-bit: 0)	≤ 110 ms	Target Value-			-	-

Figure 80 EVPL offered CoS SLA's

For the Class-of-Service profile of an end2end EVC a so-called CoS-Identifier is specified. CoS-Identifiers are unique for each EVC and per Class-of-Service. For each CoS-Identifier there is Egress-Bandwidth-Profile, an Ingress-Bandwidth-Profile and a Class defined. Class-of-Service definitions per EVC and EVPL are symmetrical and bi-directional.

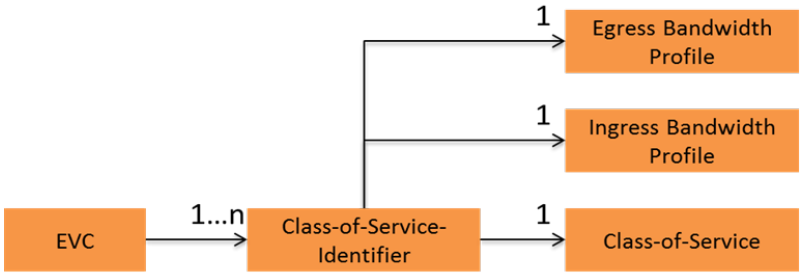

Figure 81 EVPL EVC Class-of-Service-Identifier

In regards to CoS, Classification, Marking, Queueing and Shaping to the Access-Bandwidth the RD is the central service creation or enforcement node.

NOTE: As of today the BNG also implements an Ingress H-QoS Shaper in order to be able to deliver best in class services and SLAs even if RD configurations are wrong/misconfigured and not matching ordered DTAG CoS profiles.

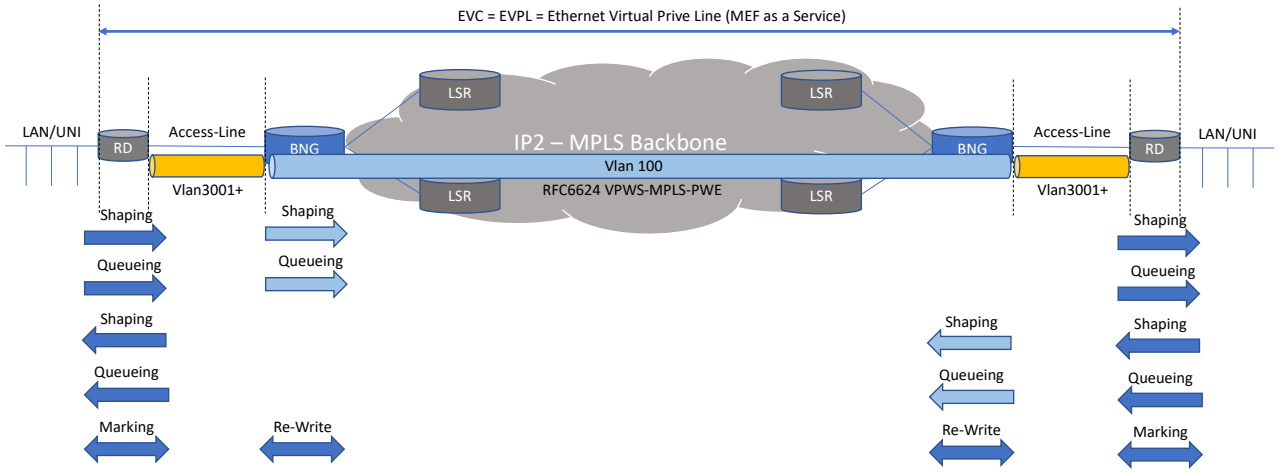

Figure 82 EVPL CoS Enforcement points

NOTE UWE: RD does also ingress police

Find here as a reference an example of an RD interface and QoS configuration:

```
************ Interfaces ***********

interface gigabitethernet 0/2/3.1102
trust 8021p
trust upstream UNI_Mixed_LD_LL_BE
qos-profile qos-p_EVC_206_100000k_NoM inbound
traffic-policy tp_EVC_206_100000k_NoM inbound link-layer
quit

interface gigabitethernet 0/2/5.3201
qos-profile qosp-L2-Egress-SPQ outbound group L2-Egress-SPQ
trust upstream default
trust 8021p
undo qos phb disable
```

```
quit


interface gigabitethernet 0/2/3.1102
description "ENNI_3201(S1102)"
vlan-type dot1q 1102
quit
interface gigabitethernet 0/2/5.3201
description "NNI_3201"
statistic enable
statistics dual-cycle enable
vlan-type dot1q 3201
quit

************ Cross Connect ***********

ccc DTAG_ENNI102 interface gigabitethernet 0/2/3.1102 tagged out-interface
gigabitethernet 0/2/5.3201 tagged

************ EVC Policies ***********


qos-profile qos-p_EVC_206_100000k_NoM
car cir 100000 pir 100000 cbs 80000 pbs 80000 priority-aware priority-template
CAR_POLICING
quit

traffic policy tp_EVC_206_100000k_NoM
statistics enable
undo share-mode
classifier L2_LD behavior tb_COS_40000k_CIR
classifier L2_LL behavior tb_COS_40000k_CIR
classifier L2_BE behavior behavior_permit
classifier MATCH_ANY behavior behavior_deny
quit


traffic behavior tb_COS_40000k_CIR
car cir 40000 pir 40000 cbs 60000 pbs 60000
quit


qos-profile qosp-L2-Egress-SPQ
user-queue cir 980000 pir 980000 flow-queue fq-L2-Egress-SPQ service-template
layer2adjust
quit
```

4.2.3.8 EVPL CoS marking and remarking

QoS Class	P-Bit Access (T-Tag) Ingress	P-Bit Pseudowire Normalisation VLAN	MPLS-TC IP-Backbone	P-Bit Access (T-Tag) Egress
Ethernet CC Control	6	6	7	6
LowLoss	3	3	7	3
Low Delay	4	4	2	4
Voice	5	5	2	5
Best Effort	0 Anything not defined	0	0	0 Anything not defined

Figure 83 EVPL CoS marking and rewriting rules

4.2.3.9 EVPL Production Model

3.7.2 Systemübersicht und Schnittstellen (EVPL Light)

4.2.3.10 EVPL L2 Lawful Intercept

For L2-Services such as EVPL based connectivity no LI is required.

4.2.3.11 EVPL Synchronization

Originally RD offered Synchronization as a service to the customer.

- L3-Business Services do not have clocking services options.
- L2 Business Access currently requires SyncE on RD ports as well as T4 on a dedicated Clocking Port.

This service seems not be required any further. There are remaining SyncE requirements for MSAN connectivity, please refer to chapter: A4 POD Timing Distribution

4.3 RD Accounting

As of today some DCIP customer still have volume based accounting. The accounting data is retrieved from the BNG as a flat file accounting.

4.4 RD Fiber on A4

NOTE: Design and specification of the RD on A4 relevant items will be part of the ART process and delivered in User-Stories. As soon as the user-stories are finalized, information will be added here. ETA defined below are still a subject to change.

4.4.1 RD Service Activation

Like in the BNG case RDs and RD services are provisioned in 3 steps:

- Business Access reservation and planning
- Fulfillment
- POD service activation

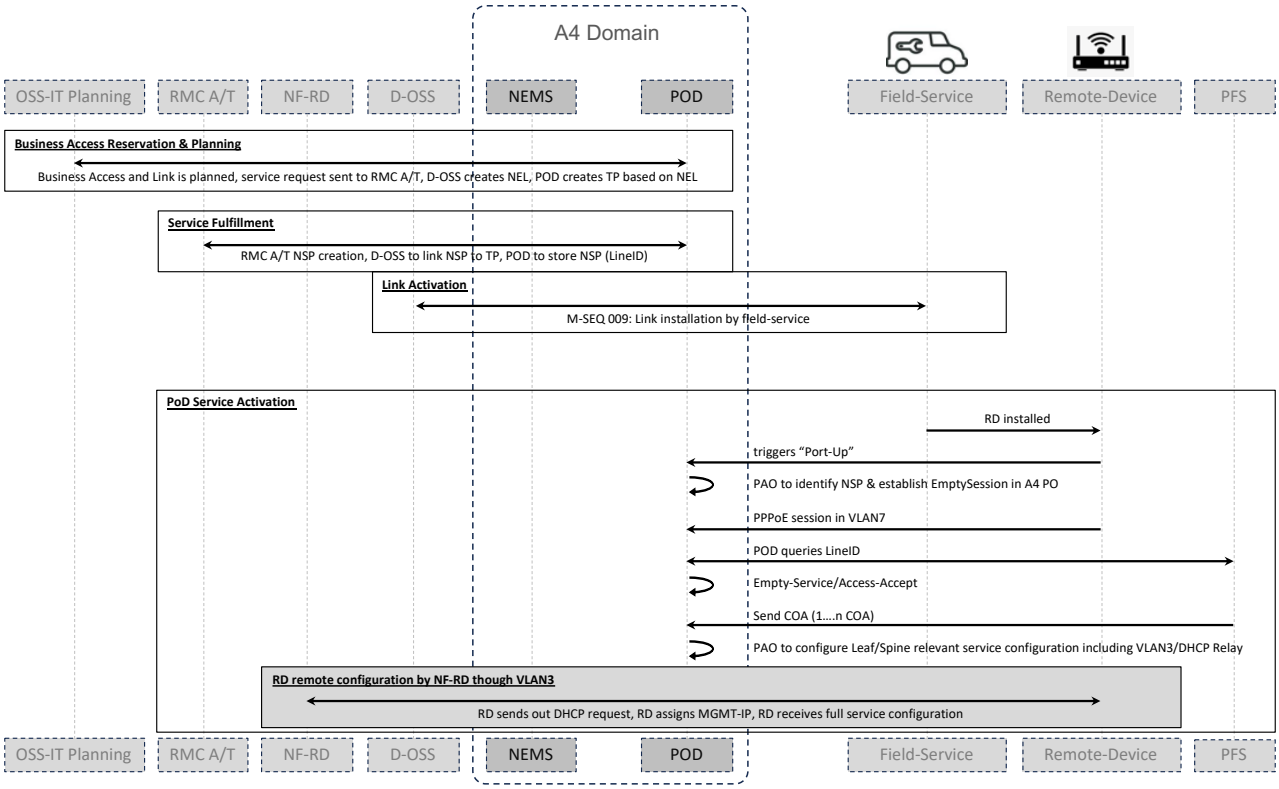

Figure 84 RD service activation

4.4.2 A4 Management Networking

Remote-Devices are managed through the Netfactory-RD, which can be reached only through the T-DCN. T-DCN is connected to the 2 redundant BOR switches. The RD itself sends and receives management traffic on VLAN3, which is terminated on the Leaf switch. But the Leaf switch is connected only through its OOB Management port to the BOR, which shall not be used for production critical connectivity. For RD it was chosen to add additional 1GE links between the BORs and the A10NSP switch, which is present in all A4 PODs.

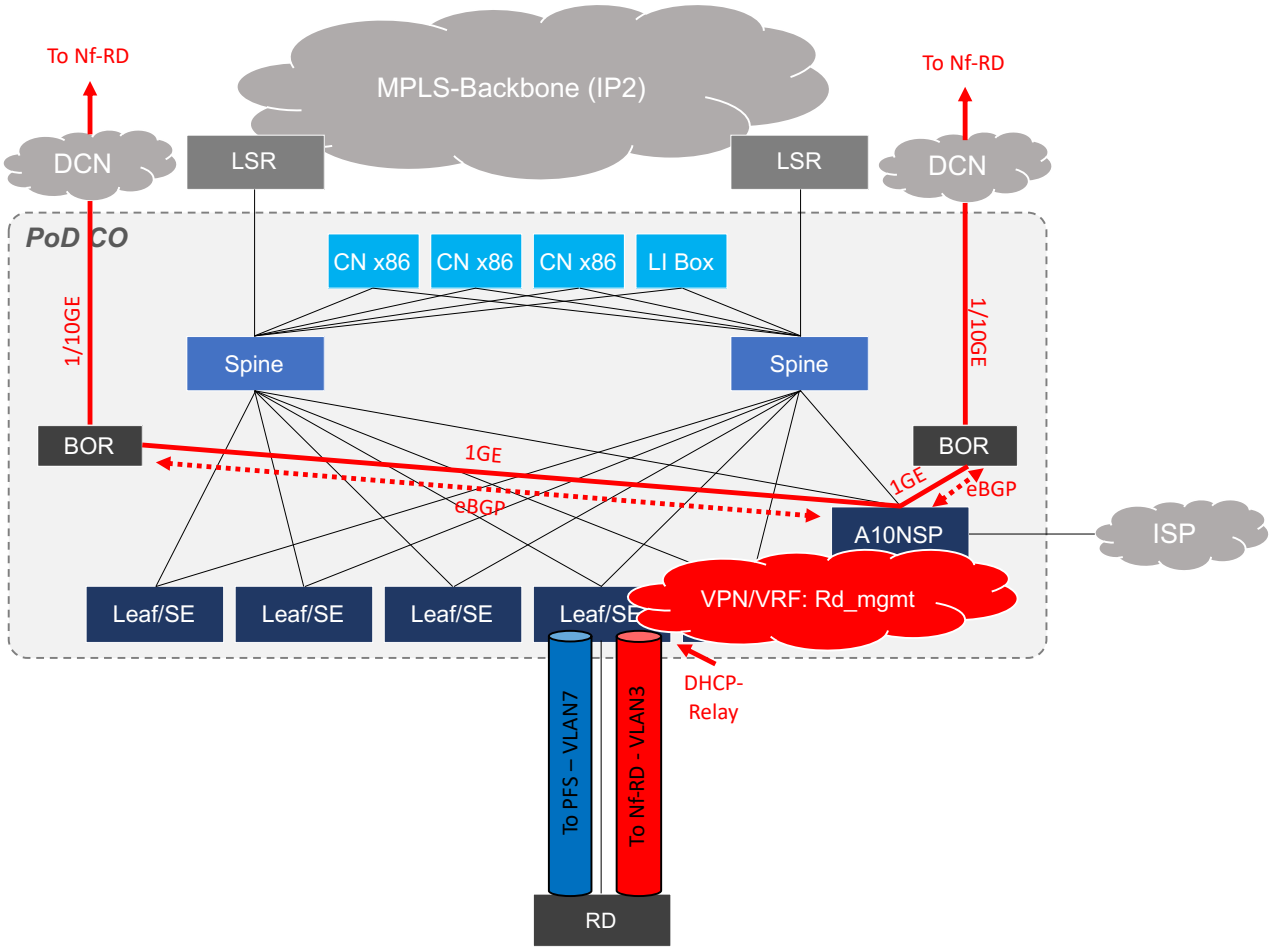

Figure 85 RD Management in A4

A POD internal VPN (Rd_mgmt) is created to forward VLAN3 traffic from the Leaf to the A10NSP switch. DHCP-Relay between the Remote-Device and the Rd_mgmt VRF is provided on the Leaf switch. The POD internal VPN (Rd_mgmt) is then being routed on the A10NSP switch towards the BOR. Between the A10NSP switch and the BOR, eBGP is being used for route exchange.

At the Leaf switches, RD Mangement networks are created using loopback sub interfaces and these networks are injected into BGP via redistribution into RD management VRF instance. At A10NSP, default route is advertised to Leaf switches in order to receive RD management traffic from downstream Leaf switches. Upstream RD Mangement traffic from the RD to A4-POD is received at Leaf and send towards A10NSP using default route in RD management VRF instance. At A10NSP, RD management routes received from downstream Leaf switches are installed into RD management VRF instance. At A10NSP, Downstream RD management traffic intended for RD access devices is forwarded using RD management VRF instance. s
The requirement is to support multiple RD prefixes (pool fragments) per Leaf switch. The first IP in the pool fragment will be assigned to Leaf and will be used as the dhcp relay agent source IP address. LineID inserted as option 82 in the DHCP frames. Since the unnumbered interfaces are used towards the RD access devices

and act as gateway and one unnumbered interface will be created for each fragment, it is must that the access interface needs to be associated with unnumbered interface which is assigned with first IP-address from the corresponding prefix pool.

4.4.3 A4 Fabric to PFS Interworking

ETA: Q3-2024

4.4.4 A4 EVPL Design

ETA: Q3-2024

4.4.5 VPWS Kompella and VPWS EPVN Interworking

ETA: Q3-2024

4.4.6 A4 RD Fabric VRF and VPN design

ETA: Q4-2024
Refer to rtbrick L2 and L3 services pdf

4.4.7 A4 EVPL QoS

ETA: Q4-2024
No Ingress QoS on Leaf for EVPL.
COMMENT: Obviously there are limitations with H-QoS. The port shaper has only two priorities/flow and will overrule the strict priority queues. This is only a problem with SDSL Ponding, when one member fails. (AndreasZ and CGiese problem statement 2106023 rtbrick meeting)

DECISION: Start RD with the 2 flow shaper implementation and clarify with BRCM the feasibility of a 4 flow shaper implementation.

4.4.8 A4 EOAM Implementation

ETA: Q4-2024
No MIP for EVPL required (Fabian statement 2106023 rtbrick meeting)

MEP is required on Leaf for DCIP. Ethernet Loopback should be support, Linktrace doesn't make sense.
Scale 2000 CFM and session interval 1000ms -> 2000pps

4.4.9 A4 DCIP

ETA: Q1-2025

4.4.10 A4 Lawful Intercept

ETA: Q1-2025
RADIUS received by PaO and then how is LI activated on the logical interfaces (non PPPoE)?

4.4.11 RD Accounting

ETA: Q1-2025

Replace flat file accounting by Prometheus streaming (tbd)
TBD

4.4.12 A4 Clocking Design

ETA: Q1-2025

4.4.13 A4 Arbor IP-Fix Implementation

ETA: Q1-2025

4.5 A4 RD Copper (MSAN)

ETA: Q2-2025
ANCP OAM - Blueprint_RebuildEdge_20181026_e01_TH.pdf – Page 14

5 A4 POD Diagnosis Framework

The Diagnosis Framework allows A4 POD external systems (e.g., OSS-IT), via the NEMO Diagnosis API, or the NEMO GUI to retrieve from the A4 POD the actual status of a service and the underlying resources or the status of dedicated resources for incident management and troubleshooting purposes.

A diagnosis can also be triggered internally by components to ascertain the working state of a resource such as for a link test.

From a high level the mechanism of either requesting status from the NEMO Diagnosis API or the NEMO GUI is illustrated in Master Sequence 20: A4-M-SEQ-020: POD Diagnostics - Access 4.0 - Telekom Wiki and the additional detailed sequences: Detailed Sequences Diagnostics - Access 4.0 - Telekom Wiki

The legacy NEMO Diagnosis API has one GET endpoint per diagnosis type. Each endpoint defines specifically which parameters are received as input and what the response contains as specified in a Swagger file. The plan moving forward is to use a POST diagnosis on NEMO unless there is a specific need for GET. These modifications are described here:
https://wiki.telekom.de/display/ACC4/Diagnosis+specific+endpoints

The EMS API defines one generic POST endpoint for each diagnosis. This endpoint is kept abstract to avoid frequent updating of the API itself when new diagnosis types are defined. Instead, the diagnosis types are defined using schemas, along the lines of the approach that is taken in the case of TMF639 APIs.

In the A4 POD the Diagnosis framework is realized through Business Event Management (BEM) API and Actors, as shown in Figure 86.

An perceived limitation of Diagnosis requests is that a single Diagnosis can only gather results from a single Responder, however, this is not really a limitation. The intention is to allow a single controller to resolve a diagnosis. The controller is free to collect information from multiple required systems. In the end, the controller can build the Diagnosis Response.

A4-seq-diagnosis-02: Execute POD-Diagnosis and get response

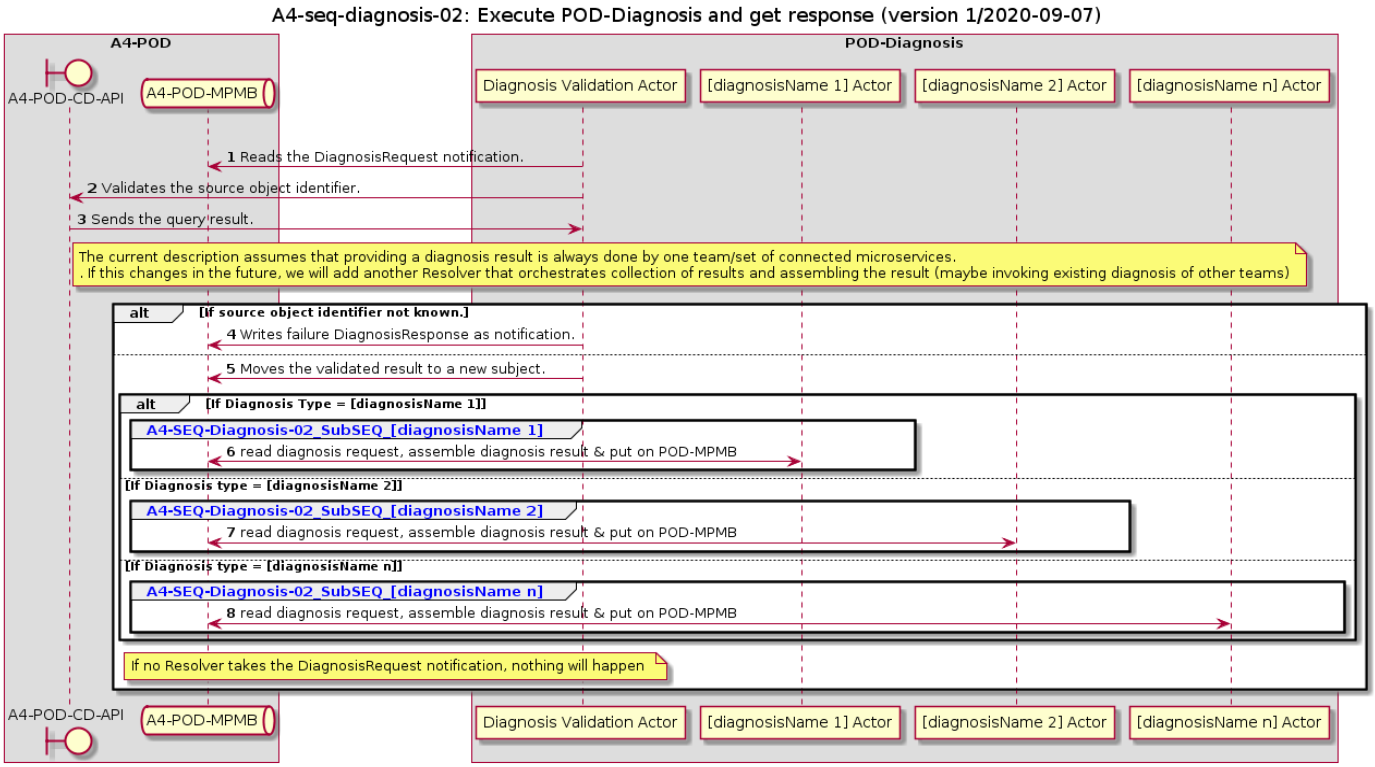

Figure 86 A4 Diagnosis

6 A4 POD Telemetry Framework

The existing A4 POD Telemetry Framework is described here:
Metrics Architecture 2.0

The A4 POD Monitoring framework describes 'Observability' in the A4-POD' describing support for:

- Logging
- Metrics
- Tracing

6.1 Logging

The Logging architecture of the A4 POD follows Device Controller specific paths:

- Switch-C Device Controller handles logging for all RTBrick Switch based Network Elements
- PON-C Device Controller handles logging for all OLT Network Elements
- DPU-C Device Controller handles logging for all DPU Network Elements
- LI-C Device Controller handles logging for all LI Box Network Elements
- BOR-C Device Controller handles logging for all BOR Network Elements

The interfaces between the Network Elements and the Device Controllers is defined by the '**IF-A4-DC-Device**' interfaces in Section 'Architectural Interface Definitions'.

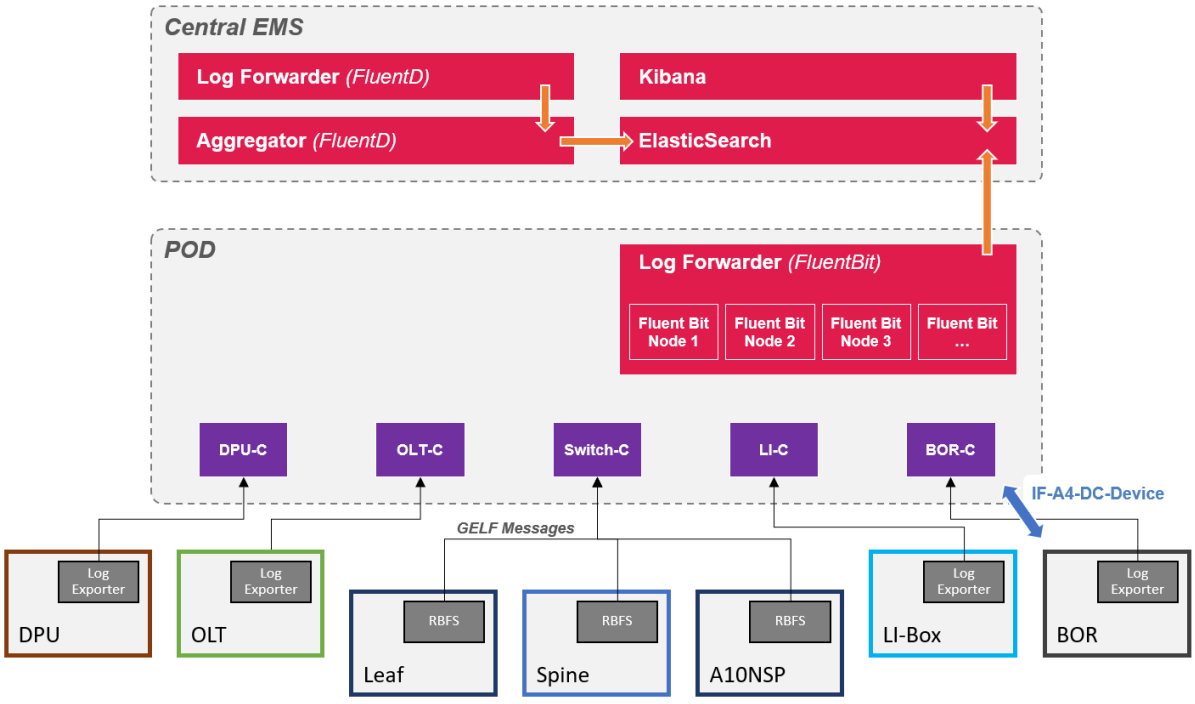

Figure 87 Generic A4 POD Logging Architecture

Each Network Element supports a specific Log exporter mechanism to its Device Controller which passes them to the Fluent Bit Log Forwarder. On the Central site FluentD is used to aggregate the logs and pass to Elastic Search.

6.2 Metrics

In the A4 POD metrics for the K3s Cluster and the Network Elements are collected.

The Metric Collection architecture of the A4 POD follows Device Controller specific paths:

- Switch-C Device Controller handles Metric Collection for all RTBrick Switch based hardware Network Elements
- PON-C Device Controller handles Metric Collection for all OLT Network Elements
- LI-C Device Controller handles Metric Collection for all LI Box Network Elements
- BOR-C Device Controller handles Metric Collection for all BOR Network Elements

The Metrics are collected and stored locally on the A4 POD in the Prometheus Time Series Database they are then aggregated and exported to the Central EMS. This is done by 'Metricbeat' which fetches the data from Prometheus transforming/Normalizing the data into JSON elastic format for storage in Elastic in the Central EMS.

Metrics rules are used to configure the thresholds to raise the metrics-based alarms. On the scraped metrics, Prometheus shall apply the metrics rules and raise/clear the alarms and dynamic configuration of the alert rules needs to be supported to enable the monitoring of a new NE, or new alarm.

This mechanism is well defined for the RTBrick switch based NEs and described below.

The current model for the Leaf/Spine and A10NSP switches is to support Prometheus as part of the inbuilt Operating System so it is deployed when the Switch Software is installed as described in the RTBrick documents below:

- Metric monitoring and Sampling - RTBrick
- https://documents.rtbrick.com/21_6_1/rbms/rbms_metrics_mgmt_guide_online.html
- Time Series Database for Switches

Within the A4 POD the OLT-C and PON-C Controllers support Broadband Forum based NetConf/Yang models and mechanisms for Metric Retrieval for Platform Hardware.

Metrics for the DPU are defined as part of Broadband Forum Yang models. The DPU-C will use the Netconf/Yang interface to retrieve the DPU Metrics and then support an interface for Prometheus to scrape the Metrics.

The Metrics Collection for the PON-C Access Controller is described here: PON Access Devices Manager.

SUSE Manager will additionally provide collection of Metrics for the POD Cluster and will provide an interface where they can be examined. Currently no plan to integrate and export to EMS.

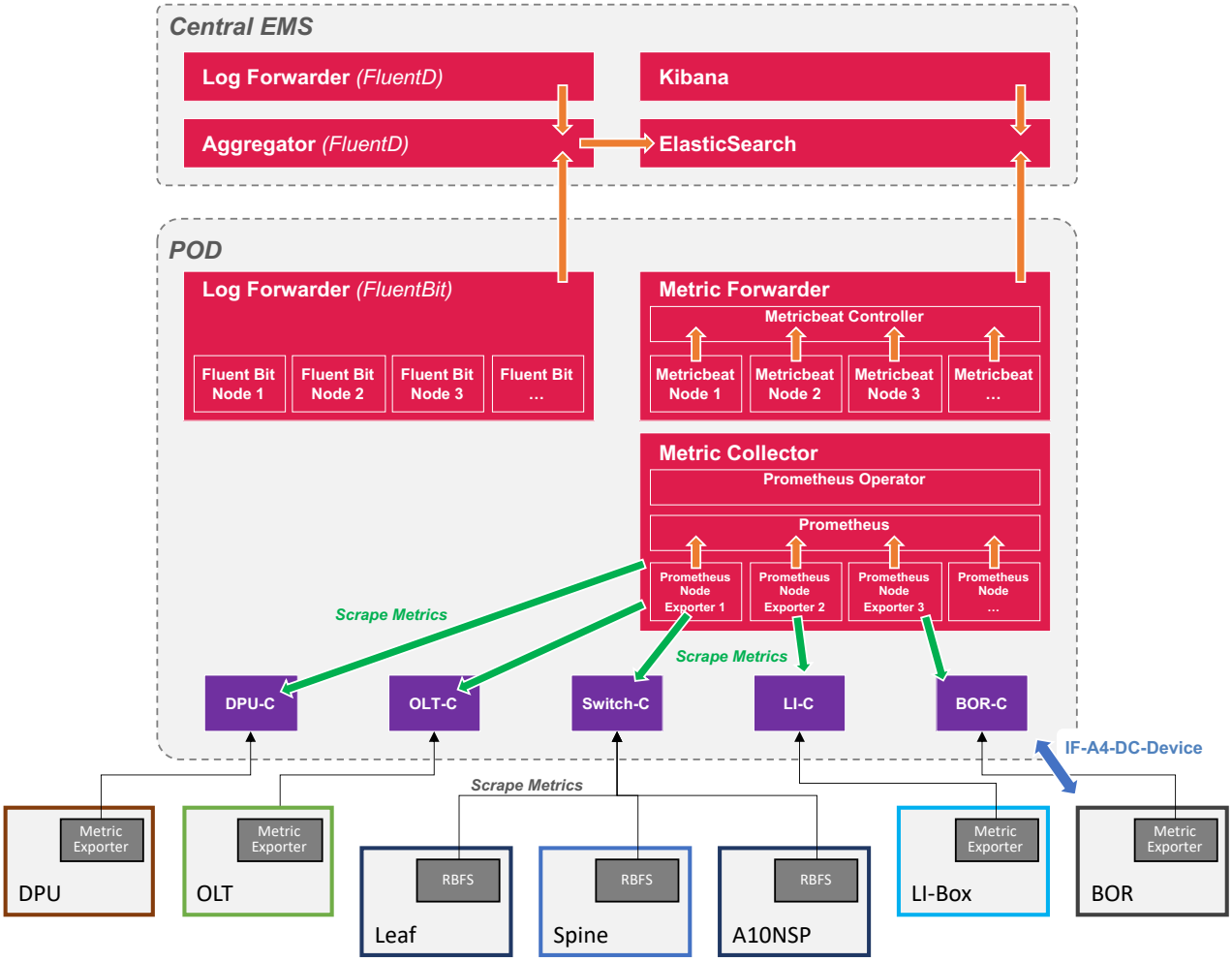

Figure 88 Generic A4 POD Metric Collection Architecture

6.3 Tracing

Traceability in transaction processing software implies use of a unique piece of data (e.g., order date/time or a serialized sequence number) which can be traced through the entire software flow of all relevant application programs. Messages and files at any point in the system can then be audited for correctness and completeness, using the traceability key to find the particular transaction.

Currently the A4 POD does not have a program wide architectural model for Traceability, however there are several references to the opentelemetry project from CNCF and general concept.

There is an approved W3C Recommendation available which describes requirements for a distributed tracing model described as 'Trace Context' which is described here https://www.w3.org/TR/trace-context/. This provides a standardized way of defining a format for the exchange of trace context propagation data i.e., a 'Trace Context'.

This standard has been implemented by several well know tracing stacks including among others:

- opentelemetry
- Elastic
- Jaeger

See the trace context implementation report https://github.com/w3c/trace-context/#reference-implementations for a better idea of the support.

The proposal would be to evaluate the outlined solution in the context of the 'Trace Context' support, the implementation available in PAO (https://gard.telekom.de/gardwiki/display/ACC4/Tracing) and the General approach to Tracing defined here : https://wiki.telekom.de/pages/viewpage.action?pageId=2091691480 and shown in Figure 89.

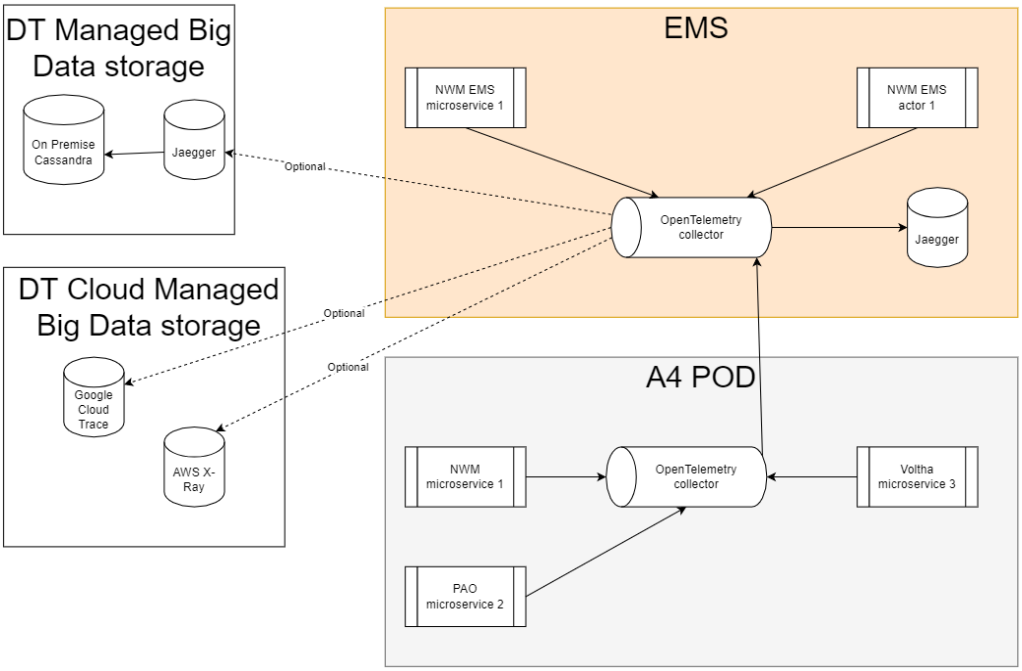

Figure 89 General Approach to Tracing

7 A4 Edge Cloud Platform

Deutsche Telekom is creating a disaggregated network access solution based on commodity components for software and hardware. The solution depends on a container-based infrastructure in an edge cloud environment. Each Edge cloud consists of a 3 node Kubernetes cluster as control and management plane. Several network elements are handling the customer traffic.

This concept describes how the 3 node Kubernetes cluster can be orchestrated and connected to the central management layer. Kubernetes will be provided at a carrier grade level, the concept will cover the installation, the basic management, and the operation of the service. The concept is based on standard products of SUSE, all customizations are outlined.

The automation is based on a three bare-metal server setup and combines the Kubernetes master and worker roles on each node. This is orchestrated by the POD Bootstrapper Service utilizing the SUSE based Provisioner that interacts with SUSE Manager and Rancher.

The installation needs to be available in a multi-home environment and fit in with the network requirements of Deutsche Telekom. The auto-installation must be independent from the used hardware and must be able to support different HW types.

SUSE Manager provides the required features to auto-detect and auto-provision the servers inside the POD and configure all parts based on central rules. SUSE Manager offers interfaces which can be used to integrate

with process automation and the system management via an API endpoint to a leading layer - in this case the A4-EMS . The Access 4.0 project has already defined a central management layer and several workflows, which are used to deploy and manage POD servers. The servers of the POD are handled like a network element so that the generic workflow can be implemented. The concept has designated a POD Bootstrapper Service which interfaces to the EMS service and drives the SUSE based orchestration.

The SUSE solution is based on standard products:
#1: POD Environment:

- SUSE Linux Enterprise-Micro
- K3S Kubernetes distribution
- Longhorn storage

#2: SUSE Central Management Layer (CeML):

- SUSE Manager
- Rancher Multi-Cloud Management (MCM)
- SUSE Tools
- SUSE Monitoring

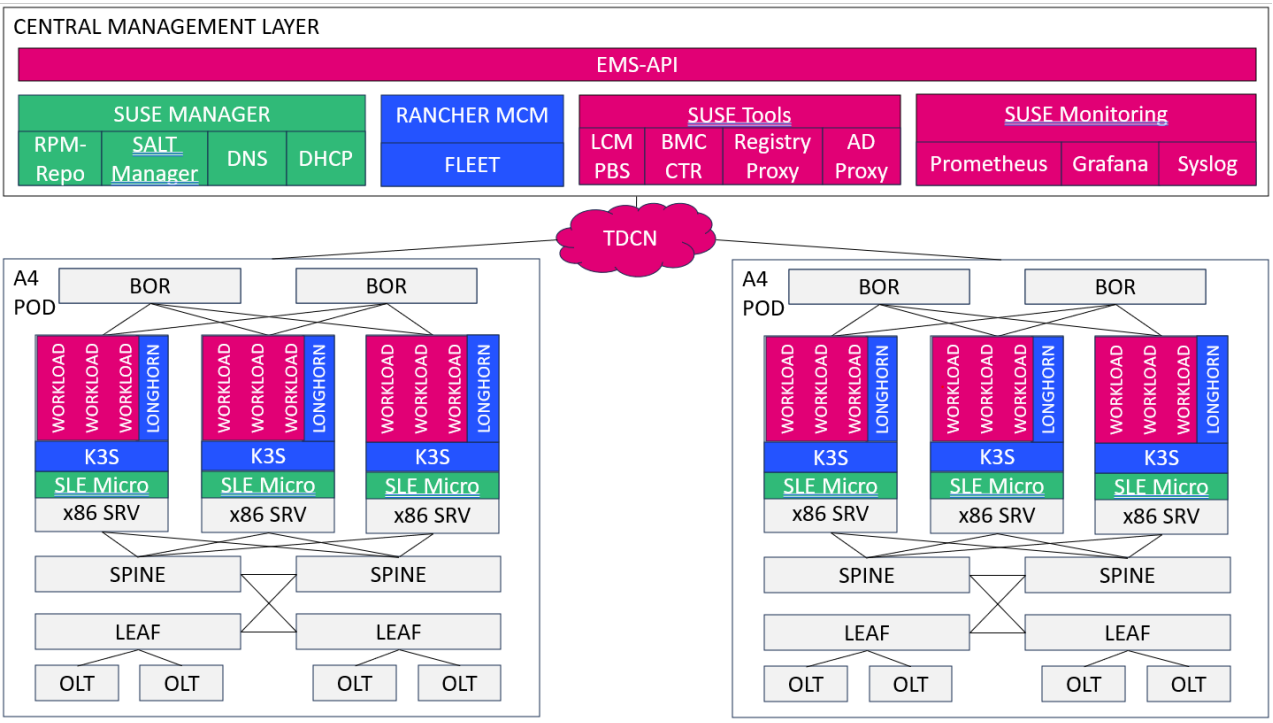

Figure 90 DT A4 Target Architecture

Figure 90 is an overview of the target architecture. The Central Management Layer (CeML) also include a few more components that are based on open-source solution and are utilized for process automation namely, SUSE Tools platform hosting the POD Lifecycle Manager and proxy services, and SUSE Monitoring platform providing the Alerts, Metrics and Logging solution for the A4 platform.

7.1 Architecture Components

The following products from SUSE form the solution architecture:

- **SUSE Manager**
- **SUSE Manager HUB**
- **SALT**
- **Rancher**
- **Longhorn**
- **Prometheus**
- **K3s**
- **Load Balancers**
- **SUSE Linux Enterprise - Micro**

7.1.1 SUSE Manager

SUSE Manager is responsible for controlling the network-based OS installation of all POD servers. All services (like DNS / DHCP / SW-Repository) which are needed for this setup are controlled via SUSE Manager.

Pre-conditions require that the hardware of a new POD node is already configured in such a way that the boot process will start on a PXE enabled interface.

Key features of SUSE manager:

- AUTOMATION API
  driven automation for Linux server provisioning, configuration, and patching
- ASSET MANGEMENT
  Inventory hardware and software systems Create reports for physical, virtual machines and cloud instances, assign subscriptions and identify over- or under-utilization
- PROVISIONING
  Provision unattended bare-metal systems via AutoYaST/Kickstart/PXE booting; virtual guests as easily as physical instances; new servers identical to a running server or predefined configuration; and SUSE Studio™ images directly.
    - Track server changes and return to a previous version or configuration if required Provision and start/stop/configure virtual guests Support first-time installation with rapid setup of network installation environments (create Cobbler systems records)
- SOFTWARE AND PACKAGE MANAGEMENT
  Collect and distribute custom software packages into manageable groups. Centrally push software by grouping servers, easing the burden of manually managing individual servers. Create customized repositories for the delivery of operating system packages or RPM Packet Manager-based (RPM based) applications and content. Migrate SUSE Linux Enterprise to new service packs directly from the SUSE Manager user interface.
    - Use the SUSE Manager application programming interface (API) to create custom scripts for easily automating many tasks- Provision RPM-based applications to automatically deploy complete, integrated software stacks.
    - Search operating system instances by packages, patches, or system specifications to reduce administrative overhead. Remove unnecessary system packages and freeze the current configuration to avoid package installations by mistake.

- PATCH MANAGEMENT
  Receive notifications when the latest Linux server updates are available. Connect to SUSE Customer Center to easily access updates, security patches and service packs Plan maintenance windows ahead of time by scheduling updates. Apply role-based controls so administrators have authority to manage each system. Significantly reduce the time to patch hundreds, even thousands, of servers via real-time configuration and monitoring
- REAL-TIME ORCHESTRATION AND CONFIGURATION MANAGEMENT
  Salt-based configuration management enables fast and secure deployment of tens of thousands of systems. Manage configurations over time to track and manage configuration drift.
  Centralize configuration file management for server groups.
  Deploy and parameterize salt formulas with standardized forms via SUSE Manager. UI Develop and maintain standardized configuration profiles for servers or groups of servers to simplify initial server provisioning.
    - Easily migrate custom scripts for Red Hat Network Satellite, create new AutoYaST and Kickstart scripts or use SUSE Manager to develop new scripts based on existing installations.
- REDEPLOYMENT
  Re-deploy on the same hardware; no physical interaction is needed.
- MONITORING SUSE
  Manager includes a comprehensive monitoring solution including the Prometheus Monitoring server and Grafana as a flexible visualization solution.

7.1.2 SUSE Manager HUB

With SUSE Manager Server 4.1 there is a new functionality called SUSE Manager Hub (See Figure 91). With SUSE Manager Hub there will be SUSE Manager Server having the role of master and there are several other SUSE Manager Servers to connect as slaves to this master server. Clients (servers) never connect to the master but always to a slave. Synchronization of software and configuration files from the master to the slave will be maintained.

The SUSE API is extended to make it possible to manage clients, that are connected to a slave, from the master.

This solution has the following advantages:

- Scalability: the number of servers connected to this solution are somewhere in the millions.
- The biggest advantage is that even when the master is down, new systems can be installed, or existing systems can be managed. When a slave is unavailable, the other slaves can still do their work.

The following consequences (currently) of this solution:

- It is not possible to see all connected server from the master. They are only visible on the slave to which they are connected.
- Only the API should be used to manage the servers.

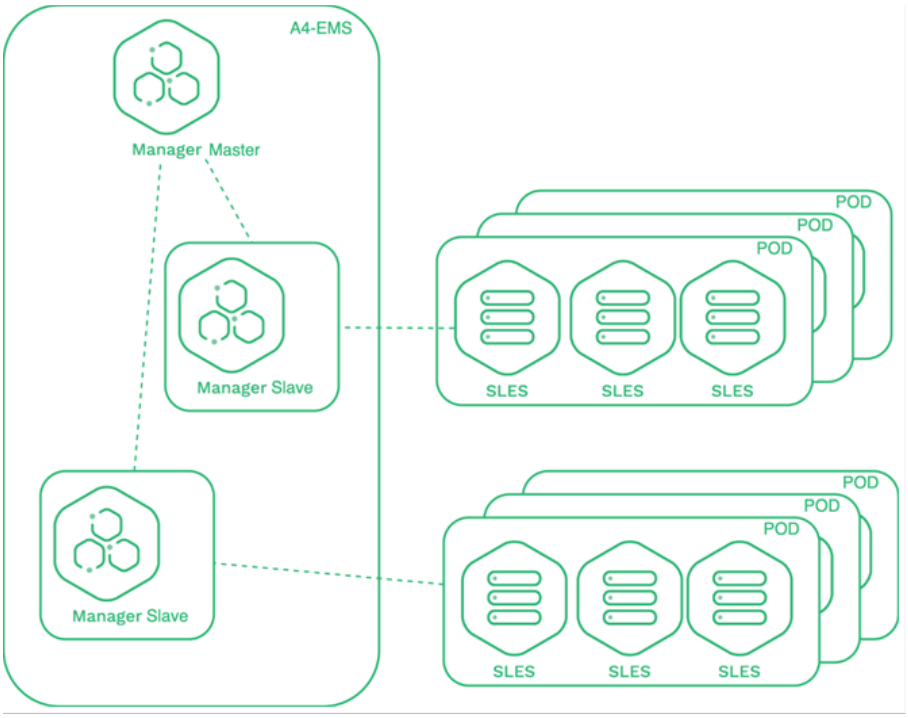

Figure 91 SUSE Manager HUB (sumam/sumas)

7.1.3 SALT as a Configuration Engine

SALT is a powerful remote automation framework. The underlying architecture of SALT is based on the idea of executing commands remotely. This could be as simple as asking a remote web server to display a static Web page, or as complex as using a shell session to interactively issue commands against a remote server. Under the hood, Salt is a more complex example of remote execution.

SALT is designed to allow users to explicitly target and issue commands to multiple machines directly and is based around the idea of a master, which controls one or more minions.

SALT here is used as the Configuration Engine to configure the base OS to fulfill the needs of the A4-POD including:

- Software Packages
- Network Configuration
- Local storage setup
- Trigger the K3S installation and register the Kubernetes cluster to the central Rancher server

7.1.4 Rancher Management Server

The central Rancher MCM service is managing all Kubernetes cluster and is responsible for configuration and updates inside of the Kubernetes environment.

Rancher itself is designed to be run atop of Kubernetes and is architected as a best-of-breed microservices-based application which extends and augments the Kubernetes API. This makes it easy for developers and operators to leverage Rancher's features, facilitating comprehensive orchestration and automation for all downstream managed clusters and their workloads.

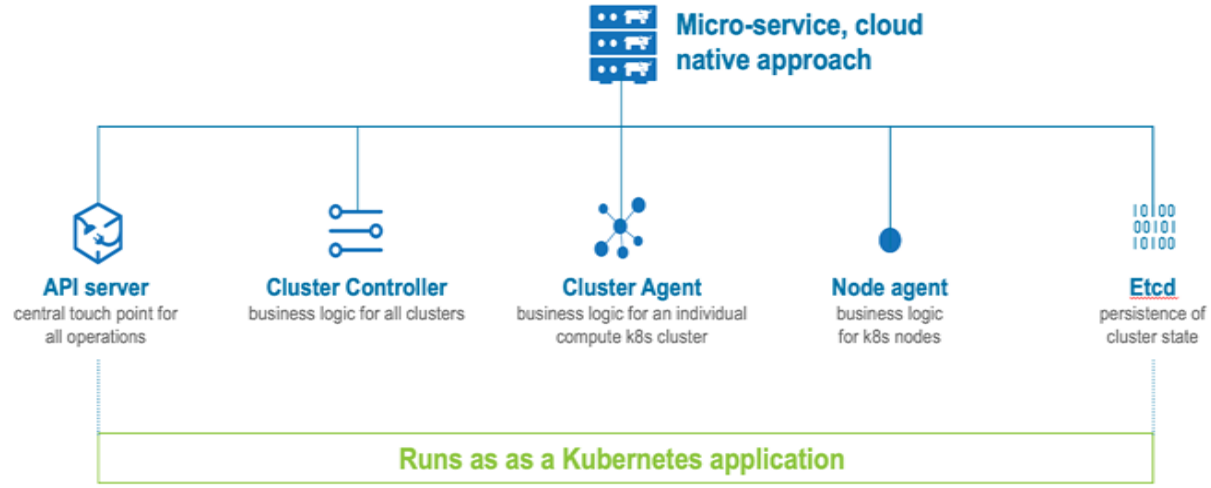

Figure 92 Rancher MCM Service

- API Server: Rancher presents a RESTful API via this server, and which is the entry point for all client interactions, including the Rancher Web UI, as well as agents running in downstream clusters.
- Cluster controller: The cluster controller contains the central logic required to manage and maintain all clusters under Rancher's control.
- Cluster agent: This agent is deployed into a given downstream cluster and contains the business logic for managing the desired state of that cluster according to the configuration stored in Rancher. It's responsible for establishing a connection back to the central Rancher server and reconciling any declared changes within Rancher itself.
- Node agent: This interacts with nodes in a cluster and is used in certain operational circumstances such as upgrading Kubernetes. It's also the fallback for the cluster agent in case that is offline for some reason.
- Etcd: This is used to persist the state (configuration) of Rancher Server.

7.1.5 Longhorn

Running stateful applications within Kubernetes can present a challenge, particularly when it comes to ensuring that a persistent storage volume can move with the pod (K3s) based on where it happens to be scheduled. To meet the persistent storage requirements for applications of this class that make up part of Access 4.0's functionality, Rancher will be introducing its cloud-native distributed block storage solution - Longhorn.

Longhorn is designed to run within any Kubernetes cluster, regardless of architecture, to provide highly available persistent storage capabilities. It is engineered from the ground-up to fit the Kubernetes persistent storage paradigms, in which persistent volumes map to a particular allocation of block storage and persistent volume claims tie those to an application running in a Pod.

A key Longhorn's design goal is that of data integrity and availability, by default it creates three replicas of every volume so that in the event of a loss of a node there is no interruption to service or loss of any data. Replicas are scheduled across Kubernetes nodes, with no more than one replica per node, resulting in a failure domain for a given persistent volume being a single node.

Architecturally, Longhorn is relatively simple, Figure 93 introduces the various Longhorn components and is not meant as a direct representation of the target deployment.

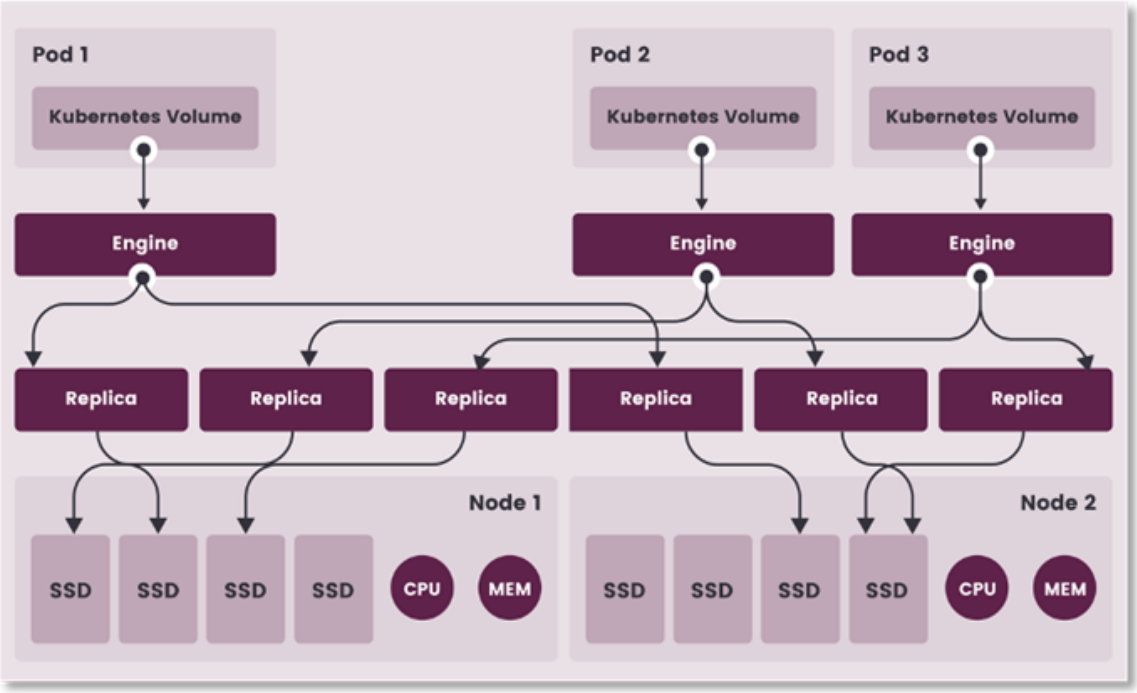

Figure 93 Longhorn Components

The key Longhorn components are as follows:

- Longhorn Manager (not displayed in above diagram): The Longhorn Manager is a custom Kubernetes controller that is responsible for managing the Longhorn Custom Resource Definitions (CRDs). It's

an application which runs on each node as a Daemon Set and is responsible for creating new volumes and managing their replicas.

- Longhorn Engine: Longhorn Engine instances are created by the Longhorn Manager. It is a dedicated volume controller, and so there is an engine instance per volume. Containerized applications running in Pods connect to the Longhorn Engine instance, and this in turn maintains connections to that volume's replicas.
- CSI Driver and Plugin: The Container Storage Interface – CSI - driver is responsible for creating a filesystem on volume and mounting it on a node, where it's subsequently bind-mounted via kubelet inside the Pod. The CSI Plugin standard implements native (to Kubernetes) persistent volume lifecycle actions, such as create, delete, attach, detach, and so on.
- Longhorn UI: The Longhorn user interface which handles presentation and interaction with Longhorn resources via the Longhorn API

7.1.6 K3s

K3s is a lightweight yet fully CNCF conformant Kubernetes distribution, ideally suited for running in Edge computing environments. K3s is packaged as a single binary of less than 40MB in size but includes out-of-the-box all the typical services and functionality required by an application stack running in Kubernetes. The CNCF conformance provides the guarantee that the Kubernetes API presented by K3s is 100% compatible with a given upstream release.

As K3s is packaged as a single binary including host dependencies such as a container runtime, it greatly simplifies the installation process, with less moving parts to worry about when it comes to automating the installation as well as upgrades. It works best in Edge-like scenarios, where it's designed to run production workloads unattended in remote locations or inside IoT appliances.

K3s was originally a Rancher Labs project. In June 2020, K3s was donated to the CNCF as a testament to the company's commitment to the open-source community. While K3s runs as a single binary on a host, its behavior can be dictated by either command-line arguments or via a simple configuration file. K3s can run in two modes, server, or agent, and within a process depending on this role the individual components are as follows:

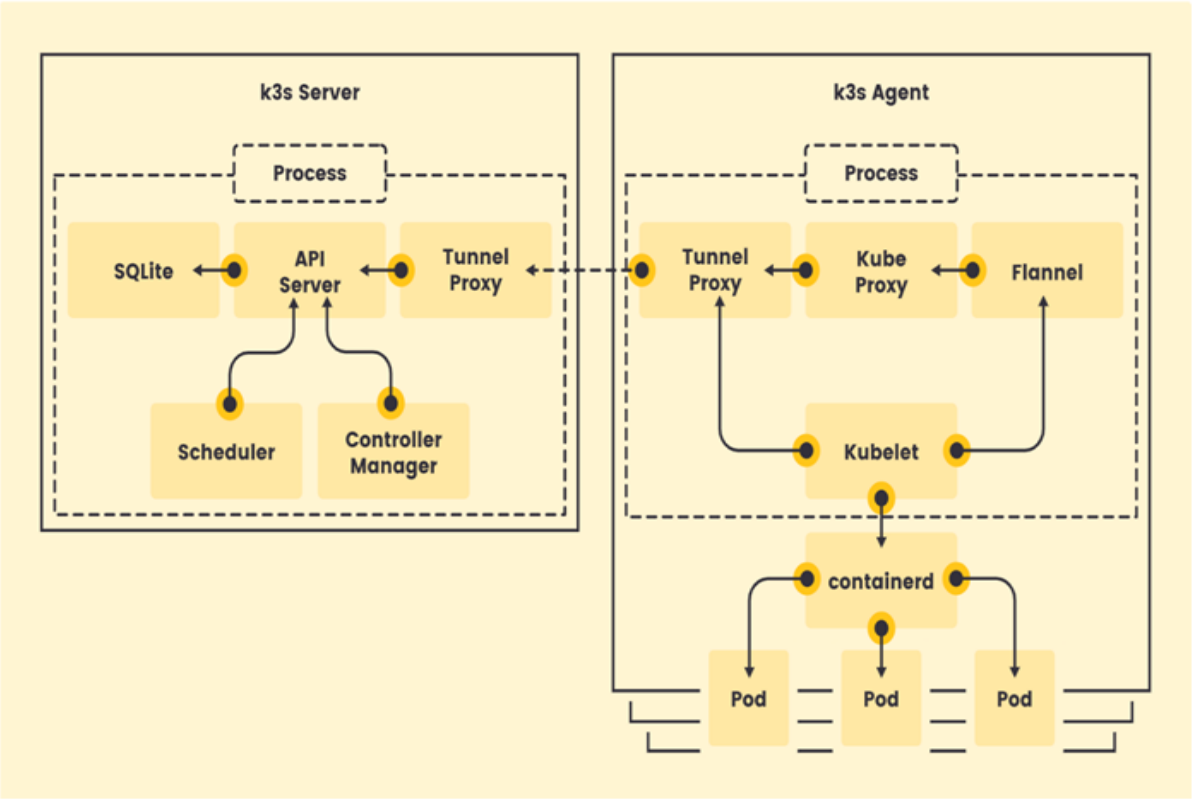

Figure 94 SUSE K3S Deployment

For deployments within A4, K3s will be deployed across three nodes in a highly available fashion using the in-built etcd distributed key-value store. Each node will be configured as both a server and an agent, to provide HA not just for Kubernetes itself but also for applications and their workloads deployed into the cluster.

Underlying Linux based OS for Worker and Server nodes (Server Cluster) is based on (SUSE Linux Enterprise) SLE-micro-OS, a transactional Operating System.

7.1.7 Prometheus / Grafana

A central monitoring server based on Prometheus will collect all node data and it is possible to forward log messages to a central syslog / logging service. As an option, Grafana can be added to provide a graphical dashboard.

Note: Monitoring architecture are under discussion, there is possibility to use Prometheus and Grafana from Rancher central components, and each A4 pod could have Prometheus running.

Data will be sent from A4 POD located Prometheus to Centrally located Prometheus.

7.1.8 Load Balancers

A new opensource project called ‘kube-vip’ is being proposed as an alternative or possibly also as a co-component alongside ‘MetalLB’.

The proposed kube-vip’ s presence is primarily used to reduce the recovery time when the employed Cilium CNI – Static Egress- feature-On fails and will automatically be re-spun by Kubernetes. The recovery time impacts re-starting of transactions from the k3s pods as the failure directly impacts the ongoing in-flight transactions.

The quicker response times stems from the fact that kube-vip plumbs the VIP to a physical interface as opposed to MetalLB that does not. The testing is still ongoing to verify the actual resolution times for MetalLB versus kube-vip. Indications to date, point to a recovery time of at least 5 mins when employing MetalLB versus kube-vip which is significantly lower (45 seconds to 2 mins, currently)

The kube-vip project is designed to provide both a highly available networking endpoint and load balancing functionality for underlying networking services. The project was originally designed for the purpose of providing a resilient control plane for Kubernetes but has since expanded to provide the same functionality for Service resources within a Kubernetes cluster.

The kube-vip service builds a multi-node or multi-pod cluster to provide high availability. In ARP mode, a leader is elected which will inherit the virtual IP and become the leader of the load balancing within the cluster whereas with BGP all nodes will advertise the VIP address.

When using ARP or Layer 2 it will use leader election.

The leader within the cluster will assume the VIP and will have it bound to the selected interface that is declared within the configuration. When the leader changes, it will evacuate the VIP first or in failure scenarios the VIP will be directly assumed by the next elected leader.

When the VIP moves from one host to another, any host that has been using the VIP will retain the previous VIP-to-MAC address mapping until the old ARP entry expires (typically within 30 seconds) and retrieves a new mapping. This can be improved by using Gratuitous ARP broadcasts when enabled (Also pending testing). The proposed kube-vip can optionally be configured to broadcast a Gratuitous ARP that will typically immediately notify all local hosts that the VIP-to-MAC address mapping has changed.

Testing will also verify that kube-vip has the capability to provide a high availability address for both the Kubernetes control plane and for a Kubernetes Service. As of v0.4.0, kube-vip implements support for true load balancing for the control plane to distribute API requests across control plane nodes.

7.1.9 SUSE Linux Enterprise - Micro

SLE-Micro (SUSE Linux Enterprise) is an ultra-reliable, lightweight operating system purpose built for containerized and virtualized workloads. It leverages the enterprise hardened security and compliance components of SUSE Linux Enterprise and merges them with a modern, immutable, developer-friendly OS platform.

SLE-Micro, facets include the following:

- **Immutable OS:** SLE Micro is a lightweight immutable OS that's optimized for edge use cases. Its immutable design ensures OS is not altered during runtime and runs reliably every single time. Further, SLE Micro leverages enterprise-hardened SLE common code base to provide enterprise-grade quality and reliability.

  SLE-Micro is immutable, which means it is a transactional OS and any changes to the OS is made by snapshots and rebooting the server for the new snapshot to be activated.

  If the new snapshot fails on reboot, the system automatically reverts to the previous working snapshot version.
- **Small Footprint and Modular Architecture**: SLE-Micro's size is optimized for small footprint installations without compromising on enterprise-grade security or quality. SLE-Micro's modular architecture maximizes developer agility and flexibility. You can start with just the Linux kernel and add required modules to create a custom image that is tailored for your application. You have full control over the footprint of the OS image.
- **Containers**: SLE-Micro is built from ground up to support containers and microservices. All applications/workloads are run as containers and separated into dedicated containers. This provides several advantages – new installation of workloads can be done without reboot, atomic updates are easier to support (create new workload, kill old workload) and it is easy to rollback when an update or configuration change goes wrong. From security perspective, workloads are isolated from the core filesystem to guard against malicious applications compromising the system.
- **Built-in Security Framework**: Includes fully supported security framework – SELinux with policies. SELinux provides a mechanism for supporting access control security policies, including United States Department of Defense-style mandatory access controls (MAC). Container runtime is adjusted to support auto-generation of SELinux policies for container workloads.
- **Secure Updates**: Updates are always security signed and verified. Additionally, the updates are easy to roll back if an update fails or is not needed.
- **Kernel Live Patching**: You can apply updates to a running kernel without the need to reboot. This helps you avoid costly downtime per device and reduce risk of cyber-attack, by applying the security updates as soon as available, without waiting for a maintenance window.
- **Certifications**: SLE Micro leverages SLE common code base, to provide FIPS 140- 2, DISA SRG/STIG, integration with CIS and Common Criteria certified configurations.

7.1.10 LI Box

LI-Box illustrations - 2 sources, currently in the process of consolidation:

- https://wiki.telekom.de/display/ACC4/Full+LI-Box+bring-up+and+update Wiki mainly Cluster bring-up
- https://wiki.telekom.de/display/ACC4/Full+POD+Stack+bring-up+and+update extended by LI-box bring-up

Decision Points to address include:

- Is a TDCN interface required for the LI Box
- Should SUSE Manager be responsible for complete Bring-up and Configuration of the LI-Box
- Should a container Application in the k3s cluster be used to configure the LI-Box, after SUSE Manager has brought up the OS of the LI-Box.

This device is not included in the POD Networking section, but we need to add it in future.

7.2 POD Cluster Networking

A4 POD server connectivity is shown in Figure 95 below

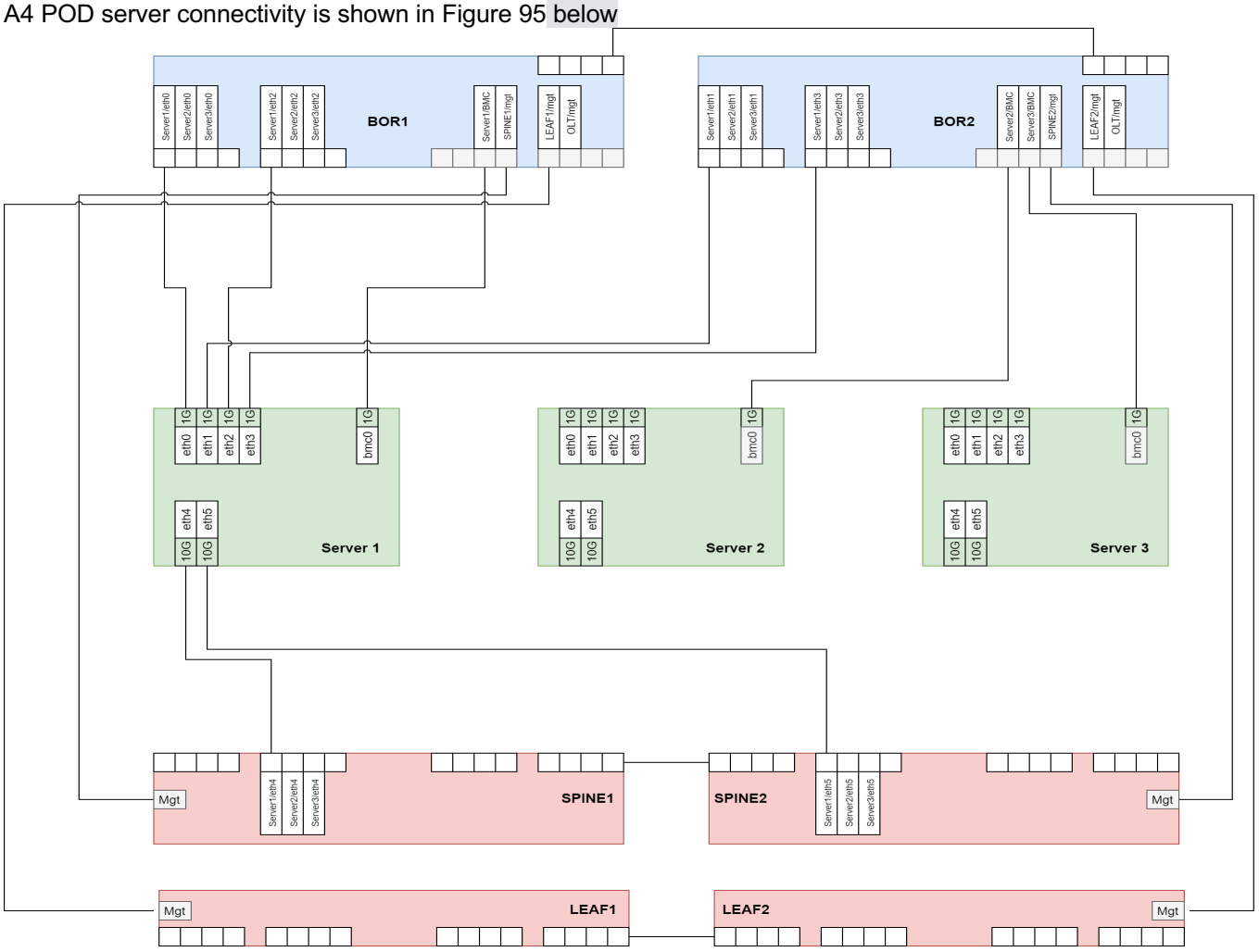

Figure 95 High Level A4 POD NE Connectivity

The A4 POD Networking conforms to standard k3s networking paradigm using the default Flannel CNI, over OOB network of the POD for running the management plane services. Access via ingress endpoints to these services are provided over the Fabric network available through secondary network ports available on the POD servers. This allows facilitating the control plane services over the Fabric network. Services the dhcp, radius utilize this network capability to provide control plane services over an independent network.

The k3s distributed applications are required to work with legacy applications north bound that are not distributed in nature: Currently Radius Client in PAO conversing with the PFS. A single IP source address per A4 POD is required when conversing between PFS and PAO Radius Adaptor. This single IP address must be Static during the entire PFS– PAO Radius Adaptor conversation allowing it to converse with the original PAO Radius Adaptor. The solution is designed to utilize certain IPs from the switch Fabric network to provide the required services over those IP.

8 A4 Bring-Up Process

The A4 POD Bring-Up process defines the procedure and the pre-requisites of POD Bring-Up from Inventory Definition and EMS Bring-Up to Inventory sync on the POD Cluster.

8.1 Inventory Definition

Inventory is defined as the textual information for the A4 NEMS, POD, NEs, NEPs, NELs and associated configuration like credentials. Ansible playbooks are used as automation for the bringup of A4 components: NEMS, POD Compute, Fabric Switch based on the component roles.

8.2 NEMS Bring-Up

EMS is deployed on Kubernetes platform provided by Das-Schiff as an API endpoint that can be used by the automation.

Ansible automation responsible for installation of services of the EMS as follows:

- Base-services like Business Event Management, Inventory, Access Manager, Secret Manager, NEMO-GUI
- Provisioner like Edge Cloud Provisioner, BOR Provisioner, Workload Provisioner
- Interface services like DigiOSS Adapter, ECP Adapters, Workload Adapter, NEMO-Adapter
- Telemetry & Notification services like ELF, Kibana, Prometheus, Grafana, Alert Manager

On completion of the EMS Bring-Up, inventory data is used in templates to create the YAML files for NEG, NEs, NEPs. Inventory YAML is then used to populate the data in the EMS.
The below Figure 96 elaborates on the NEMS Bring-Up Process

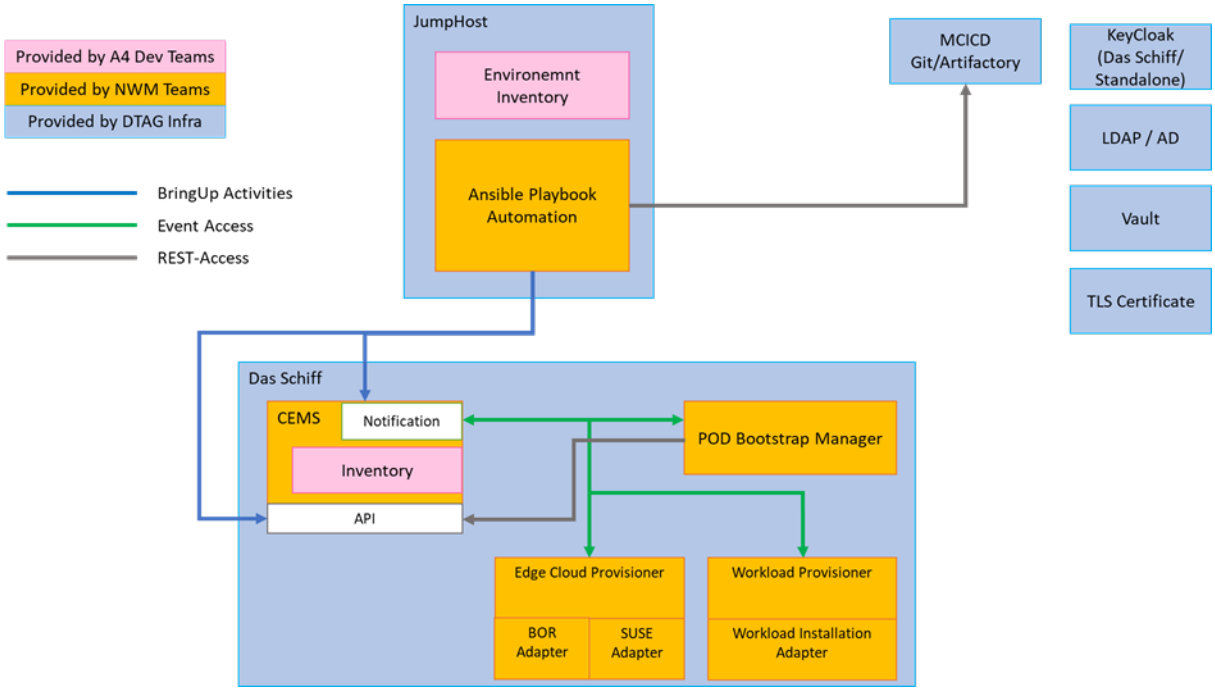

Figure 96: NEMS Bring-Up Process

8.3 SUSE CeML Bring-Up

SUSE CeML is deployed as VMs on the DT-OASIS environment, using ansible automation.

8.3.1 SUSE Manager

SUSE Manager is deployed as 2 VM in a master-slave configuration, deployed independantly or using automation and configured using SALT scripts. SUSE Mnaager is responsible for ZTP orchestration of a node based on the templates and incoming data from the PBS. It provides DHCP, DNS, HTTPBoot & SALT master services that it uses from Server Boot to Kubernetes Cluster creation.

8.3.2 SUSE Tools

This is a 3 VM deployment to run K3s cluster over SLE Micro OS. The deployment is automated using SUSE Manager and Fleet(CD). It provides services like Lifecycle Manager(LCM), POD Bring-Up Service(PBS), BMC-Controller. Acts as southbound endpoint for SUSE Adapter and utilizes the following deployments.

8.3.3 SUSE Rancher

This is deployed as 3 VM running SLE Micro OS providing the K3s cluster. It is deployed using automation template defined in SUSE Manager. It is responsible for POD Cluster Management and management of services like CD services via Fleet and Storage services via Longhorn.

The below Figure 97 elaborates on the SUSE CeML Bring-Up Process

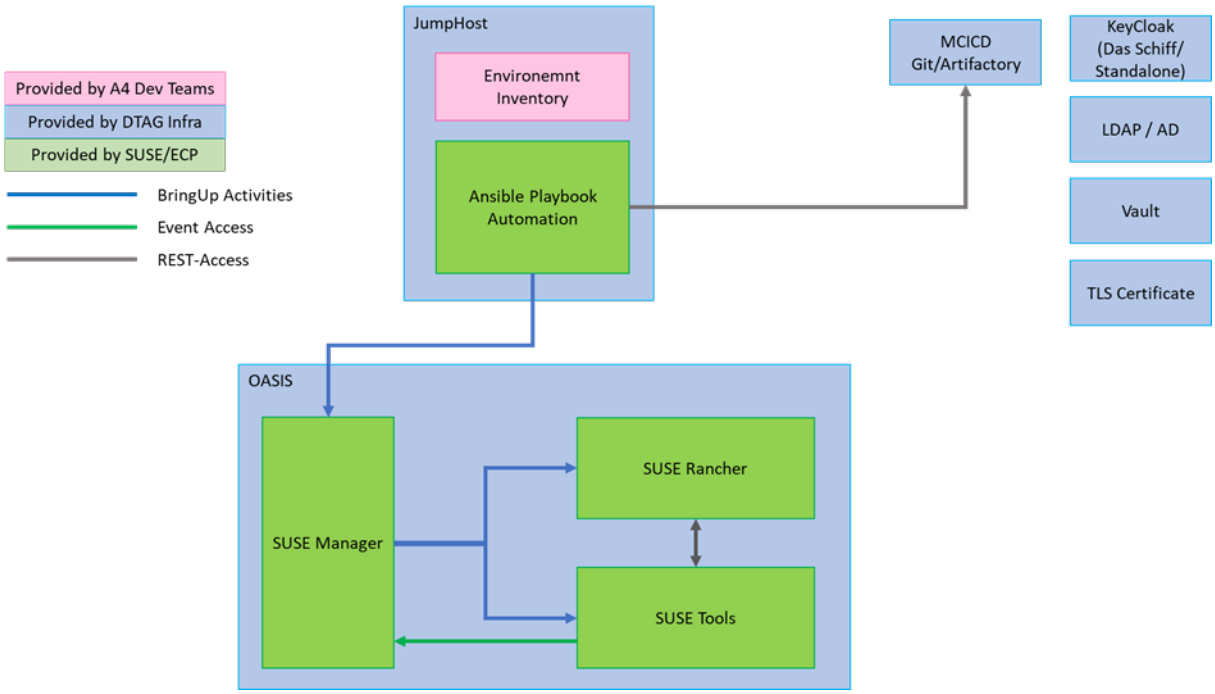

Figure 97: SUSE CeML Bring-Up Process

8.4 POD Bring-Up

BOR and POD Server are orchestrated and configured using the ECP Adapters to interact with Southbound components i.e. BOR-Adapter and ECP-SUSE- Adapter.

8.4.1 BOR Adapter

This component is responsible for populating the templates from the BOR Provisioner with the BOR inventory data. It acts as an adapter for boot-up and configuration of the BOR via Console CLI provided by terminal console server.

8.4.2 SUSE Adapter

This adapter interacts with the ECP Provisioner and communicates with the deployment in the SUSE CeML and POD Cluster to run through the POD ZTP steps. It provides POD Server Hardware information and orchestration data. The adapter communicates with the ECP LCM to initiate the POD ZTP and monitors the process using notification updates from the LCM. ECP-LCM is also responsible for Lifecycle Management of the POD cluster including OS, K3s and firmware. ECP-LCM utilizes the ECP-PBS, deployed on SUSE-Tools for Pod Bring-Up using BMC-Controller and SUSE Manager.

On completion of the POD Bring-Up, the POD Kubernetes cluster credentials are shared with the EMS, which triggers an Inventory sync. EMS uses the K8s cluster on the POD to deploy local EMS services like PAO, Switch-C, OLT-C, DPU-C, VOLTHA, etc and their associated configurations.

8.5 Fabric Bring-Up

Switch-C service on the POD provides DHCP and NOS endpoint to the Fabric switches for Bring-Up & NOS installation followed by configuration based on the

- NEL Definition: Ansible automation is used to populate the NEL in the NEMS and push the same towards the POD.
- Spine Bring-Up: Spine boots to DHCP 61 and installs the NOS, followed by configuration sync based on inventory and template.
- Leaf Bring-Up: Leaf boots to DHCP 61 and installs the NOS, followed by configuration sync based on inventory and template.

On completion of the above-mentioned procedure, the POD is considered commissioned and ready for connecting with the south bound-devices like OLT, MSAN, RD, etc.

Figure 99 provides overview of the POD & Fabric Bring-Up

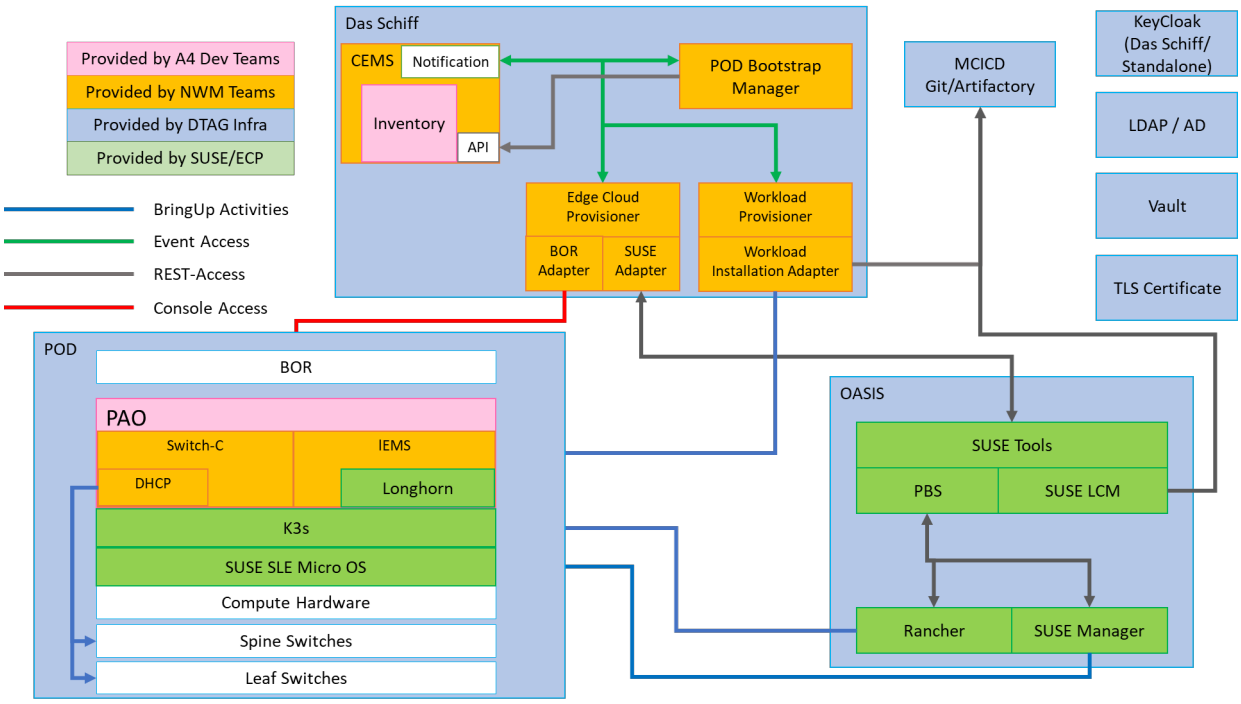

Figure 99: POD & Fabric Bring-Up

9 A4 Edge Cloud Platform Bootstrapper

The A4 POD Bootstrapper architecture allows us to decouple EMS / Inventory related code base from the actual business logic responsible for Bootstrapping an A4 POD. Bootstrapping process refers to the action of transforming a group of POD servers into a working K8S cluster on which the A4 workloads are running successfully. Before finishing the Bootstrapping process, the complete EMS Inventory is going to be synced with the newly bootstrapped A4-POD.

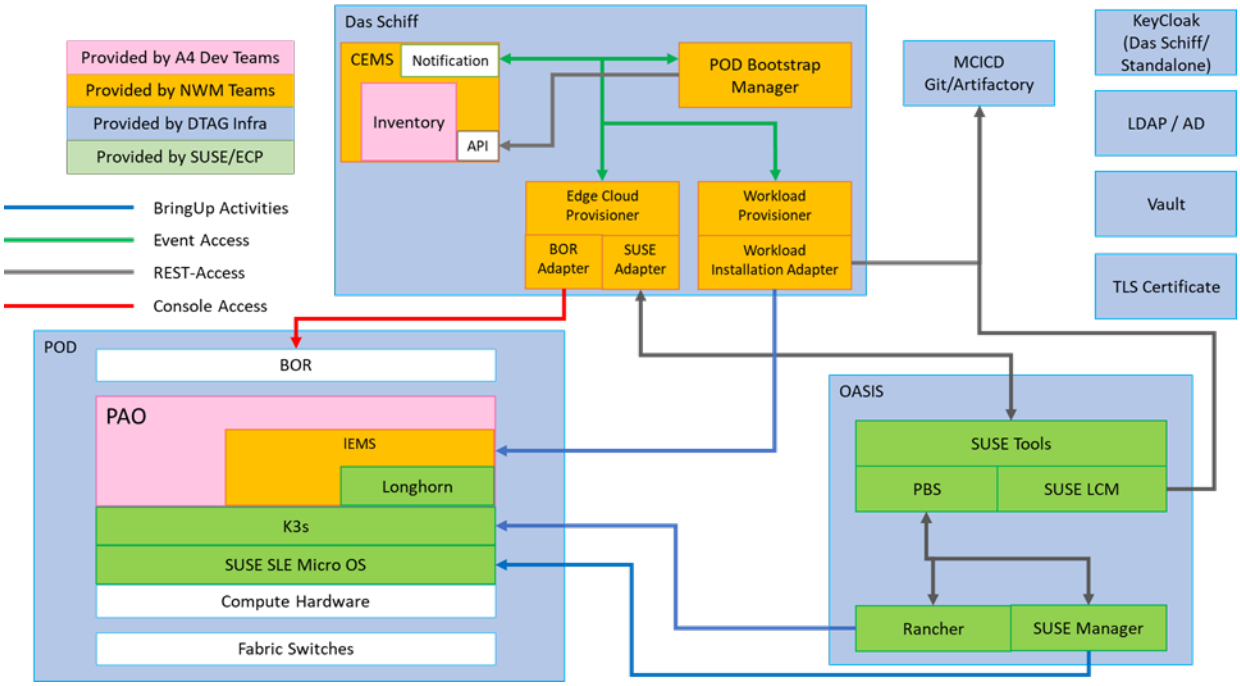

Figure 96 High Level Architectural View of the Bootstrapper Components and Execution Environments

Figure above shows how the Bootstrapper function of the Edge trapper Components and Execution Environments Cloud Platform is distributed amongst the Execution environments and their external interfaces:

"Das Schiff":

- NEMO/A4-EMS,
- Bootstrap Manager,
- Edge Cloud Provisioner (Adapter),
- Workload Provisioner

"OASIS":

- Edge Cloud Provisioner (Controller)
- SUSE Controller (DTAG API),
- SUSE Manager
- Rancher

"A4 POD Cluster"

- Application Workloads
- Infrastructure (Kafka, NATS etc.)
- K3s

- SUSE OS (SLE-Micro)
- Hardware Platforms

The following Figure shows the design components and process to implement the POD Bootstrapper.

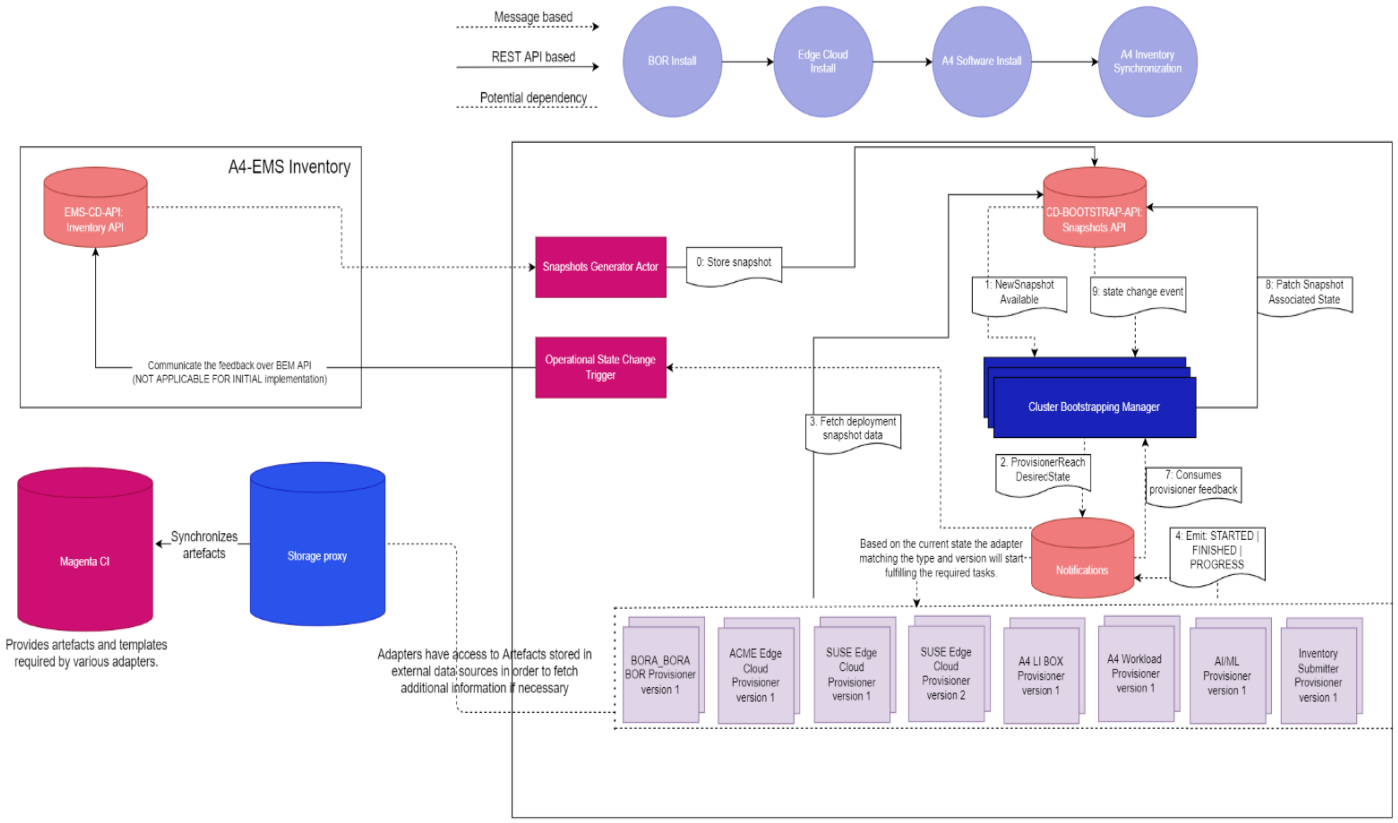

Figure 97 A4 POD Bootstrapper High Level Design and Process

To fulfil the process and design described in Figure 97 we have the following major components:

Component	Abbreviation	Description	Execution location (DT)
Snapshots Generator Actor	SGA	This is an actor responsible for generating snapshots when specific business rules apply (calculated based on A4 received notifications)	• NEMO/EMS
Pod Bootstrapper API	CBA	Provides a unified REST API for storing immutable snapshots and providing access to their individual resources using GraphQL query language. Each snapshot will maintain progress state for each resource relevant for the bootstrapping process.	• NEMO/EMS
Cluster Bootstrapping Manager	CBM	Provides a configurable workflow engine responsible to emit relevant notifications towards **Provisioners** layer. In addition, the CBM is also responsible for translating Provisioners feedback notifications into State changes stored in the **CBA**.	• NEMO/EMS
Provisioner	-	A provisioner is a logical abstraction for implementing business logic which can fulfill one or more phases configured in the CBM component. The target Cluster Bootstrapper Manager for A4 will be configured to support the following phases: • BOR installation • EdgeCloud installation • Workload onboarding • Inventory submission Conceptually a provisioner will have multiple components: See: https://gard.telekom.de/gardwiki/display/ACC4/Provisioners+anatomy for more details.	• NEMO/EMS
Adapter layer	-	The adapter layer of each provisioner is responsible to combine CBA information together is a catalogue (one or more artefacts found in Magenta CI) into a set of API calls relevant for the Controller. Generates ‘STARTED’ notification (Item 4 in above diagram)	• NEMO/EMS
Controller	-	The controller is responsible for integrating the actual third-party technology in a way compatible with the adapter layer. Compatibility means technology specific layer has a way to communicate feedback to the adapter layer which can be translated into StepProgress or StepFinished notifications (item 4 in above diagram).	• NEMO/EMS • OASIS The location depends on the actual controller being deployed.

			For instance, SUSE Controller is going to be deployed in OASIS.
Third party dependencies	-	Each provisioner has specific technologies being used to fulfill CBM phases. This is an optional dependency because some of the provisioners simply do not need additional technologies to fulfill a CBM phase (e.g., Inventory Submitter). Here are a few examples of third-party dependencies: • SUSE Manager • SUSE Rancher • SUSE Tools • Fleet	• OASIS

More specific details can be found in one of the following pages:

- https://gard.telekom.de/gardwiki/pages/viewpage.action?pageId=423953375 (OREO E2E design)
- https://wiki.telekom.de/display/ACC4/EMS+A4-POD+Bootstrapper (Overall architecture presentation)
- https://wiki.telekom.de/display/ACC4/%28BORA_BORA%29+BOR+provisioner++overview (BORA_BORA provisioner details)
- https://wiki.telekom.de/display/ACC4/Edge+cloud+provisioner+overview (EdgeCloud provisioner details)
- https://wiki.telekom.de/display/ACC4/Inventory+provisioner+overview (Inventory provisioner details)
- https://wiki.telekom.de/display/ACC4/Workload+provisioner+overview (Workload provisioner details)

10 A4 POD Timing Distribution

The A4 POD supports two types of timing distribution for two different purposes:

- Network Time Protocol (NTP)
- IEEE 1588v2 / SyncE

IEEE 1588v2 PTP protocols are used for synchronizing the hardware timing (phase, frequency and time of day) with network based highly accurate timing sources, supporting mobile fronthaul and backhaul timing requirements.

NTP is utilised as a low-cost way to distribute ToD for purposes of Billing, Generating Timestamps for Events, Security, and other applications where microsecond accuracy is not required.

While both message-based approaches depend on a low latency, symmetric transport network the IEEE 1588v2 PTP mechanisms can support the Class C 'cTE' requirements of +/- 10 nanoseconds required for 5G transport applications and 1588v2 timing accuracy is also required for Business Services.

Class C clocks must also use EEC, as noted in ITU-T G.8262.1 'The highest performance can be achieved in networks where only enhanced EEC or better clocks are implemented, see [ITU-T G.8261] for the applicable network limits.'

Synchronous Ethernet (SyncE) is based on physical layer ethernet timing. The timing reference signal from a master clock is transmitted over the ethernet interface and this is used by the slave ports to synchronize to the signal. The architecture is derived from the SONET/SDH architecture, i.e., each node in the hierarchy has an ethernet clock and synchronizes to the node above. The advantage with SyncE is the synchronization is unaffected by the traffic load on the node elements. However, it comes with a few disadvantages. All the nodes in the network must support SyncE, to enable synchronization end-to-end. Moreover, it supports only frequency synchronization and not time/phase synchronization.

The hardware enabled for IEEE 1588v2 can interwork with existing frequency synchronization mechanisms such as EEC where PTP-aware nodes operate in a "SyncE assist" mode to improve PTP performance.

10.1 A4 NTP Concept

The A4 NTP concept is described in Telekom-Wiki here:
A4 NTP Concept

NTP can usually maintain time to within tens of milliseconds over the public internet and can achieve better than one millisecond accuracy in low latency local area networks such as the DT T-DCN network. Within the DT POD the expectation of around 1 ms accuracy is expected, with some higher jitter expected for the DPU as an NTP client due to the asymmetry of the PON Network but the expectation is that the DPU ToD would be within a few milliseconds accuracy.

Figure 98shows the NTP timing distribution between the A4 POD Server NTP Clients and the DT Stratum 1 NTP Pool Servers. The SUSE Enterprise Linux is configured after installation to connect by default to the DT Stratum 1 NTP Servers with a "Key". The "Key" is required only for A4 POD external NTP communication.

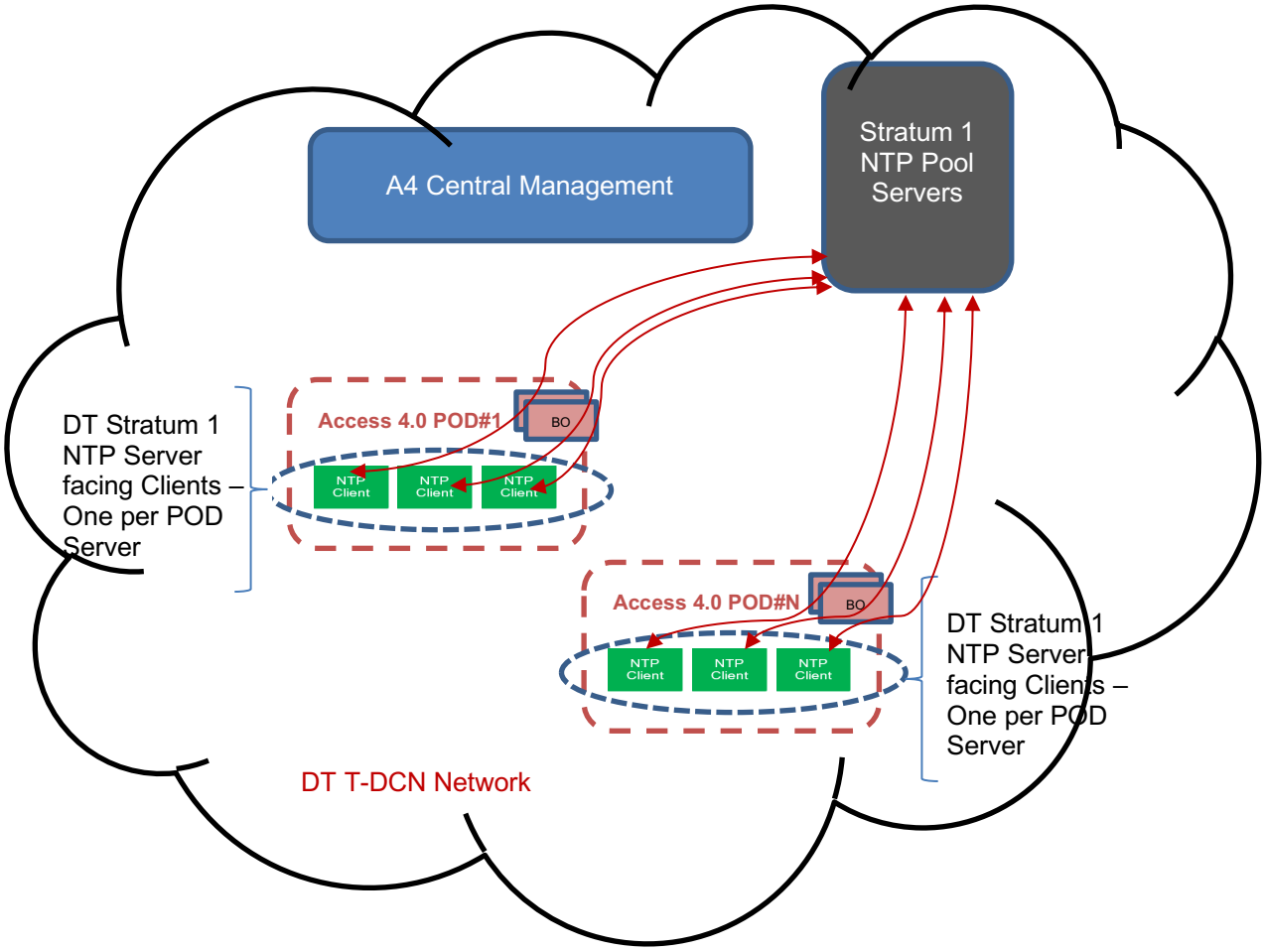

Figure 98 POD Server NTP Clients connect to DT Stratum 1 NTP Servers

The SUSE Linux Enterprise distribution is configured to provide a POD Internal NTP Server instantiated on each physical POD Server. The POD NTP servers act as a Stratum 2 NTP time Source based on the POD servers synchronizing with the POD External DT Stratum 1 NTP servers accessed via T-DCN network.

The POD switches, LI Box, OLTs and DPU provide NTP Clients which are configured to communicate with the POD NTP Servers supporting the following heirarchy of NTP time distribution withing the A4 POD as described by : A4 NTP Concept

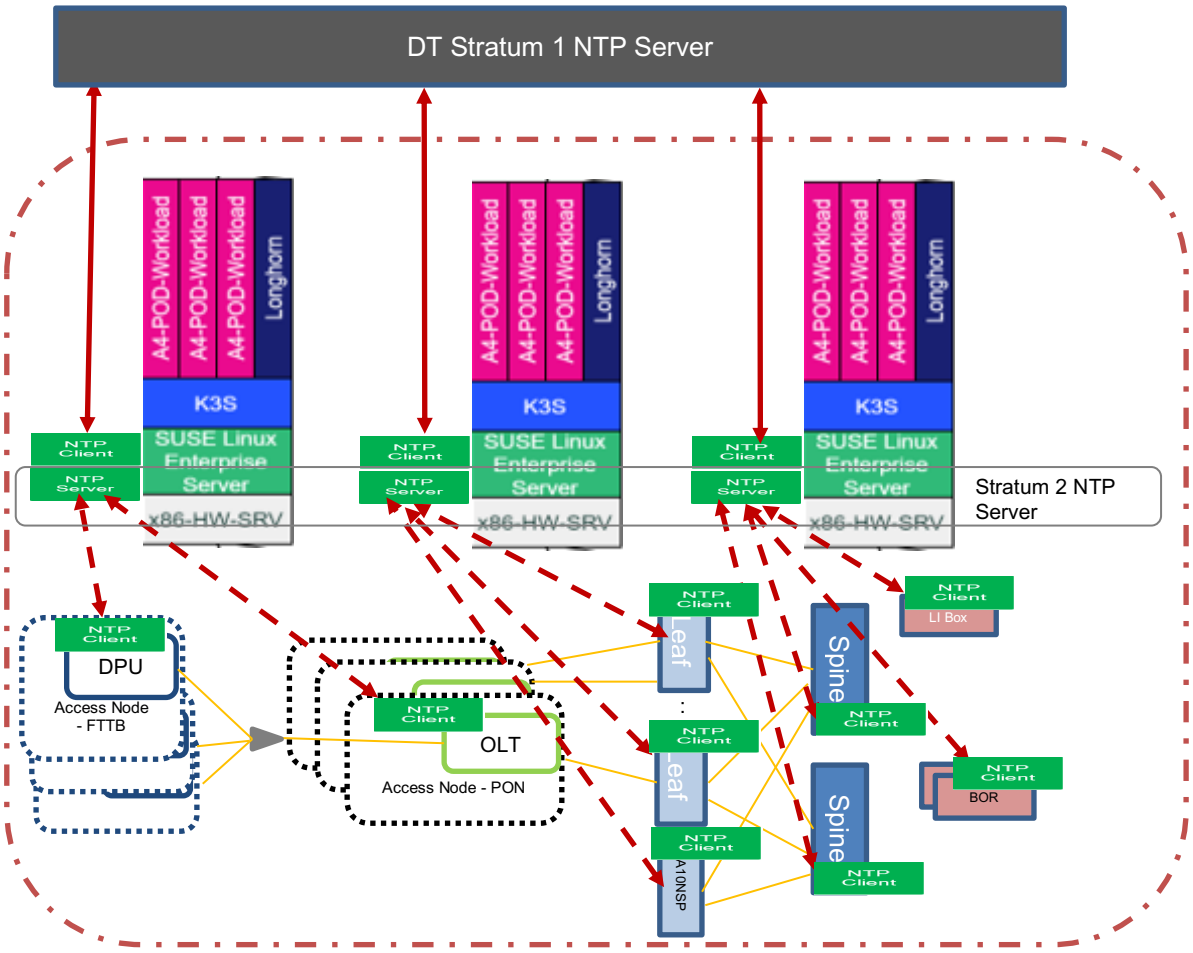

Figure 99 Intra-POD NTP Time Synchronisation

As noted in A4 NTP Concept the NTP Timing distribution hierarchy is shown in in the following Figure.

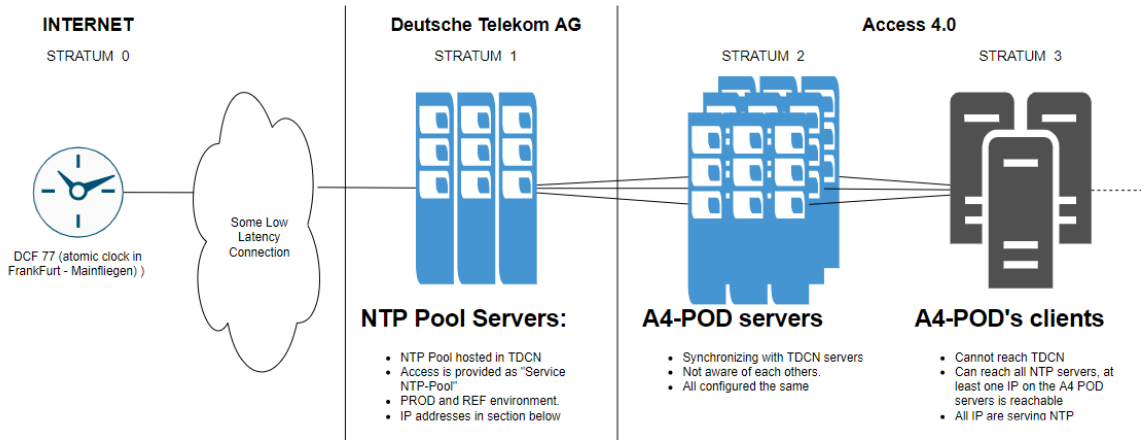

Figure 100 A4 NTP Timing Distribution Model

10.2 A4 PTP 1588v2 Concept

The A4 POD switch hardware must support the following Clocking/Synchronisation requirements for accurate phase/time synchronization:
(see Telekom-Wiki Reference: Timing Requirements Spine-Leaf hardware

- Network element must support Telecom Boundary Clock Class C according to ITU-T G.8273.2
- Network element must provide HW readiness for 10ns constant time error (cTE)
- Network element must provide support 1G interface SyncE and Precision Time Protocol with Full-Timing Support (PTP-FTS) with layer 2 encapsulation according to ITU-T G.8275.1
- Network element must support ITU-T G.8262.1 Enhanced Ethernet equipment slave clock (EEC)
- Physical interface:
    - Network element must provide two physical SyncE/PTP-FTS interfaces for clock input supporting grey optical SFPs
    - Network element must have a measure interface 1pps (1 pulse per second) with 50 Ohm / coax according to G.703
    - Network element must have a measurement interface 2048kHz with 120Ohm / RJ45 according to G.703

The A4 POD 1588v2 and SyncE timing hierarchy is shown in the Figure below:

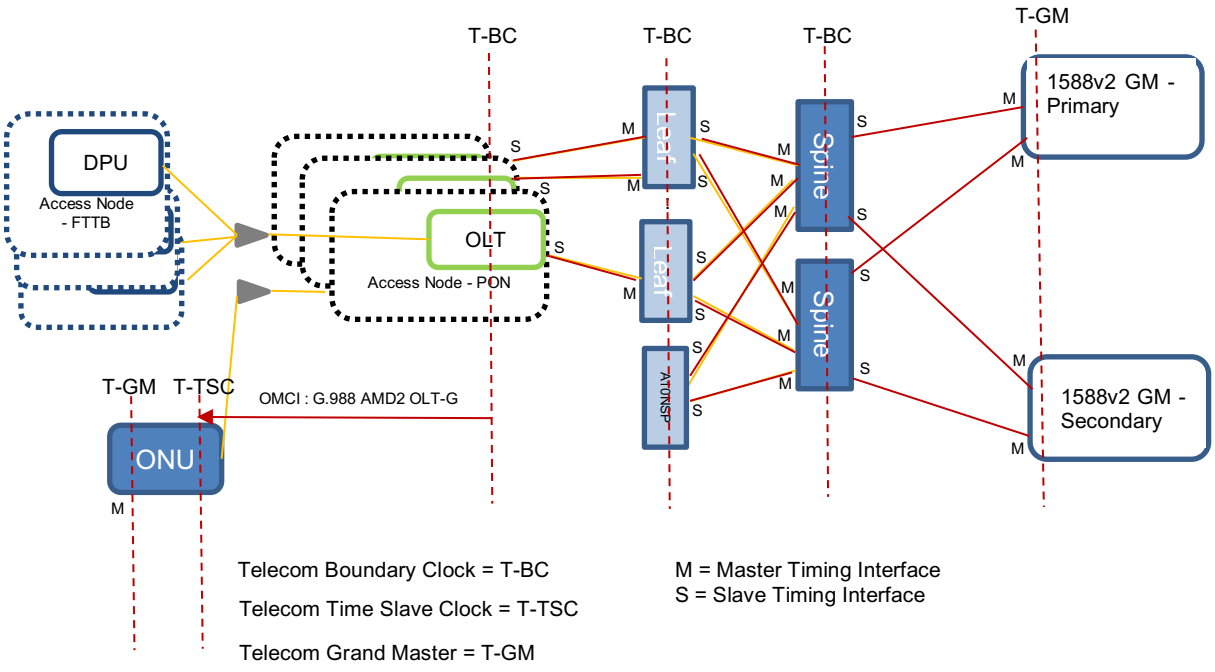

Figure 101 A4 POD PTP 1588v2/SyncE Hierarchy

As noted in the Figure above there are several types of PTP interfaces supported (ITU-T G.8275.1) in the A4 POD:

- Ordinary Clock: Ordinary clocks implement only a single copy of the protocol and behaves either as a Telecom Time Slave (T-TSC) slave or a Telecom Grand Master (T-GM) in a master-slave hierarchy. The slave clock synchronizes its clock with its master. The Grand Master synchronizes its clock to an external source like GPS.

- Boundary Clock: A Telecom Boundary Clock (T-BC), typically has many physical ports, and each port behaves like a port of an ordinary clock. One physical port behaves as a Slave Clock to the clocks above it and all the other ports behave as a master to Clocks below it in a cascaded architecture. Boundary clocks are typically enhanced network elements like switches or router with PTP support.

- Transparent Clock: A Telecom Transparent Clock (T-TC) is a special type of PTP node that behaves neither as master nor as a slave. This node transfers the incoming PTP message to the next node and in the process modifies the timestamps to compensate for the residence time in the node. This is done by addition of the residence time to the correction field of the PTP message.

In addition to the ITU 1588v2 PTP timing distribution the PON uses OMCI messages to distribute a high precision Time of Day to the connected ONUs enabled by the ‘T-REC-G.988 _201908-Amd2’ specified ME ‘OLT-G’.

Where SyncE Input is supported, the interface receiving ESMC messages will select the highest priority interface to accept the Frequency Input and where no SyncE messages are received the local provisioning will select the highest priority interface. Where SyncE Output is supported the output interface will indicate the priority of the signal based on the selected Input SyncE messages or the local priority configuration.
As defined in ITU-T G.8262.1 enhanced synchronous Ethernet equipment clock (eEEC) must be supported.

11 SLA Monitoring

11.1 Service Availability

The Access 4.0 production model foresees (currently) 4 areas of services and its SLAs:

- Mass-market / DT retail
  - 97%, Period of clearance 24h
- Mass-market / Wholesale L2BSA
  - 98,5%, Period of clearance 4h
- Business DCIP
  - 99,5%, Period of clearance 8h
- Business Ethernet Connect
  - Period of clearance 8h, penalty

These SLAs are the KPIs contractually guaranteed between Deutsche Telekom AG and its subscribers. Although they will be produced on the same platform, they have different availabilities and clearance times. The difference can be explained by:

- Difference response times to a failure
- Different processes for the restoration of a failure
- Different level of resilience of the related components

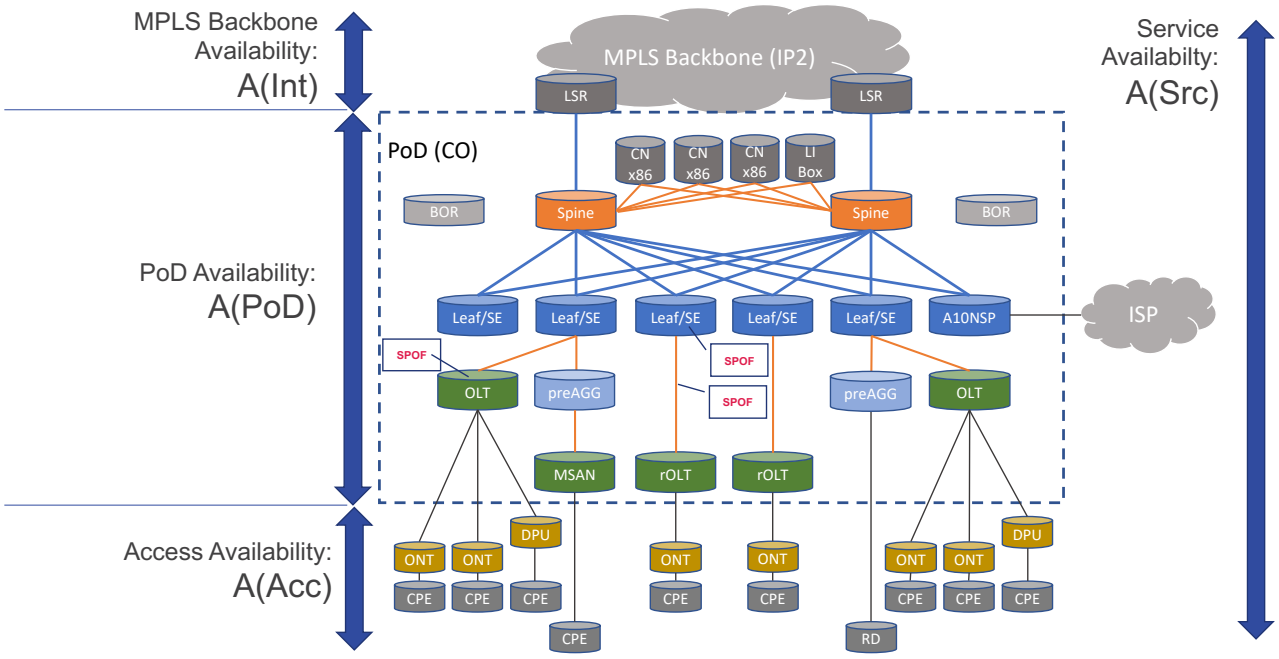

Figure 102 End-to-end view of the services availability

From a technical perspective the committed Service-Availability can be described as the product of the Access Availability, the PoD-Availability, Internet-Availability and Support (Reaction times, processes etc.).

Suitable KPIs need to be developed and a methodology for determining Service Availability developed. The current methodology is to use algorithms based on the TL 9000 standard. The A4 POD will produce KPIs which provide for monitoring the SLAs for subscribers also derived from the TL 9000 models.

11.2 POD Availability

The Access 4.0 POD (without OLTs) is an equivalent of the BNGs in production and the Access 4.0 solution in terms of the relevant KPIs should be equal or better. The POD Availability can be divided in two areas:

- Data-Plane (All that is required to keep user traffic functional)
  - Empty Session (required for all services such as retail or L2BSA)
  - PON sessions, PPPoE sessions, Fabric-Routing (Underlay plus Overlay), Internet-Routing
    - Contains networking protocols which are usually considered to be Control-Plane (i.e. BGP)
  - L2BSA and other services
- Control-Plane (All that is required to manage user sessions and operations)
  - VOLTHA, ONOS, Rtbrick Fabric-controller, PAO, Proxies

The POD Data-Plane Availability includes everything in the POD (plus OLTs) so that the subscriber has an operational session and Internet-Access.

The POD Control-Plane Availability includes everything in addition that provides production from an operations perspective:

- Activating sessions (Provisioning of subscribers)
- De-activating sessions (De-Provisioning of subscribers)
- Accounting and statistics
- Alarming and logging
- Diagnosis

NOTE: A failure of the Control-Plane should not affect the data-plane availability. Means, loss of control-plane doesn't affect established connections (PON, PPPoE, Routing).

Therefore, the overall POD Availability can be defined in 2 different ways:

- POD Availability
  - Data-Plane Availability = 99,99%
  - Control-Plane Availability = 99,999%

11.3 POD Control-Plane Availability

The following components of the PoD are contributing to the Control-Plane availability:

- Power (USV and DC to be checked)
- Cooling
- Bottom-of-Rack switches (Hardware/Software)
  - POD Cluster Networking
  - Optics, cables, transceivers
- Bare-Metal compute (x86) servers
  - Red PSU but non redundant NIC (Uwe)
- The host OS
- Kubernetes and the clustering; SLE-Micro (SUSE Linux Enterprise)
- The applications (Micro services)
  - VOLTHA
  - PAO
  - ONOS
  - Proxies
  - Switch controllers
  - Etc.

The Control-Plane components and the system is designed in a highly available manner. Compute systems are redundantly connected to redundant BoRs. They are equipped with redundant power-supplies and are optimized for the target environment temperatures.

Operating systems are hardened and operate on the latest patch levels to avoid security threats and minimize the risk of OS failures. SLE-Micro (SUSE Linux Enterprise) is an ultra-reliable, lightweight operating system purpose built for containerized and virtualized workloads, its immutable design ensures OS is not altered during runtime and runs reliably every single time. Further, SLE Micro leverages enterprise-hardened SLE common code base to provide enterprise-grade quality and reliability.

SLE Micro is immutable, which means it is a transactional OS and any changes to the OS is made by snapshots and rebooting the server for the new snapshot to be activated. If the new snapshot fails on reboot, the system automatically reverts to the previous working snapshot version.

A cluster-based Kubernetes design will assure all micro-services in the control-plane are highly available, even if a host fails.

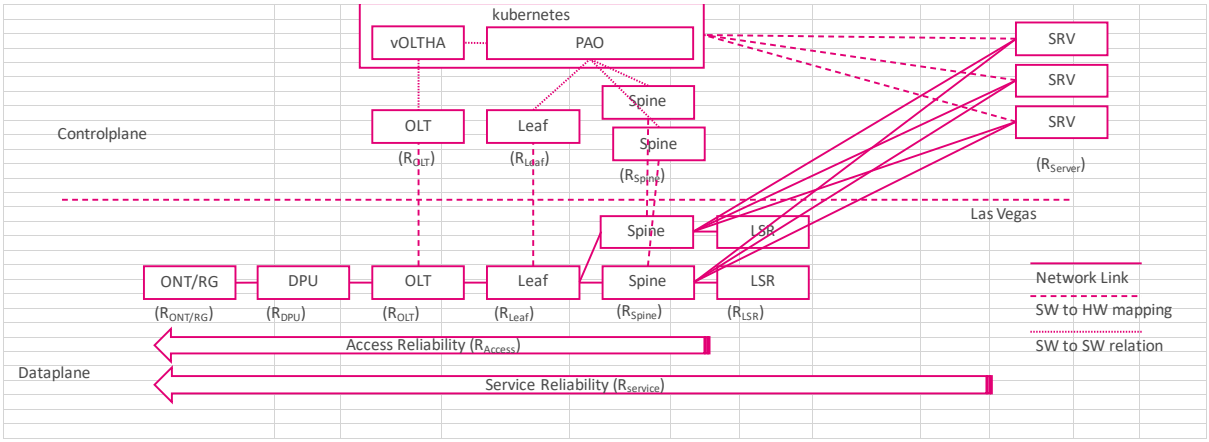

Figure 103 Control-Plane availability (DT model)

The Control Plane Availability is calculated based on the Reliability models described in 'ETSI GS NFV-REL 003 V1.1.2 (2016-07)'. It should be noted that this standard also recommends utilising the monitoring infrastructure to gather statistical data about Network Service element failures which can then be used to estimate parameters like the MTBF and MTTR more accurately.

11.4 POD Data-Plane Availability

The following components of the POD are contributing to the Data-Plane availability:

- Power
- Cooling
- Switches (hardware and software/NOS)
  - Spine
  - Leaf SE
  - A10NSP
- OLTs
  - Also, remote OLTs
- Optics, transceivers, and cables

Components such as fibre splitter, DPUs, ONUs and RGs are part of the Access and not the POD. The Internet-Backbone is high available but also outside of the POD availability considerations. Spines and Leaf Uplinks are redundant (Fabric design). A failure of a single Spine would be recovered within a certain convergence time. Single points of failures are:

- Leaf SE switches

- Uplink to the OLT (Optics, transceivers, and cables)
  - Assuming non redundant OLT uplinks
- OLTs and remote OLTs
  - Downlink Optics, transceivers, and cables

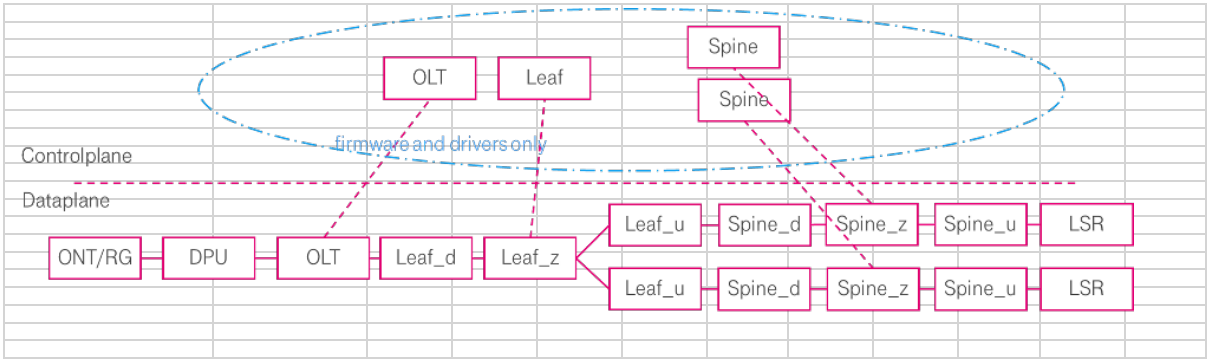

Figure 104 Data-Plane availability (DT model)

11.4.1.1 Impact of failures on POD Data-Plane availability

If power or cooling fails in a central office, then the entire POD is affected. The same is true for catastrophic damages like fire or flooding. In these cases, an MTTR of 4 hours cannot not be achieved.
A Spine failure would affect only half of the subscribers (assuming a 50%/50% load sharing between the Spines) only for the MTTR of the re-convergence. All other POD data-plane failures are single points of failures (without Dual Homing of the OLT in case of a Leaf failure), but with different impact on the number of affected subscribers:

- Leaf SE switches failure
  - All hosted and switched subscriber sessions of that Leaf will go down until MTTR
    - The impact is a percentage of the full number of subscribers per PoD and should contribute to the overall POD Data-Plane availability accordingly
- Downlink to the OLT (Optics, transceivers, and cables)
  - All subscriber sessions of that Leaf SE to OLT link will go down until MTTR
    - The impact is a percentage of the full number of subscribers of a Leaf SE switch and should contribute to the overall POD Data-Plane availability accordingly
- OLTs and remote OLTs
  - Same as above
- OLT Downlink Optics, transceivers, and cables
  - All subscriber sessions of that PON tree will go down until MTTR
    - The impact is a percentage of the full number of subscribers of an OLT or remote OLT and should contribute to the overall POD Data-Plane availability accordingly

11.5 EMS Availability

The EMS is supposed to be geo-redundant in 2 different locations, so that the EMS availability cannot be affected by catastrophic damages, such as fire and flooding. Each EMS site is considered to be highly available, based on the same concepts as the POD Control-Plane. EMS availability does not impact the POD availability. Further considerations on the EMS availability are to be done.

11.6 PoD Scalability

POD scablity is an important KPI for the over Access 4.0 business case. The more subscriber sessions per POD can be maintained on defined quantity of components, such as compute and switches, the better is the return-of-investment. The most relevant scaling KPIs are:

- Number of subscribers
  - Per POD
  - Per Leaf SE (or A10NSP)
- Number of provisioned subscribers per second
  - Per POD
  - Per Leaf SE

11.6.1 Number of subscribers per POD and Leaf SE

The maximum supported number of subscribers per POD has two aspects:

1. The software applications (PAO, ONOS, Controllers) plus their design must cope with the defined maximum number of subscribers. In addition, data-bases, system memory and defined interfaces between the applications (APIs) in the POD have to be taken into account.
2. The Leaf hardware itself. A switch has a chipset with a certain scalability (schedulers, tables, queues, memory, buffer).
   - For example, the supported maximum number of subscribers per Leaf in terms of IP or PPPoE interfaces.
   - Each type of service may require different number of resources. One example is the number of queues and their depth per service. Some residential services may require only 4 queues per subscriber, but then other service may require a higher number of queues.

The maximum supported number of provisioned subscribers per second has also multiple dimensions:

1. The software applications (PAO, ONOS, Controllers) plus their design must cope with the defined maximum number of subscribers per second.
2. CPU of the servers and the switches and of course the POD networking.
3. The interfaces to the switches, such as REST must be designed appropriately

11.7 POD Bring-Up KPIs

The KPIs regarding POD and subscriber bring-up contribute significantly to the overall quality of the system:

- The POD bring-up time is the time between turning on a fully commissioned POD and the time when all microservices, the internet routing, and the component management is fully operational (for example after roll-out or a reboot during a maintenance window).
  - BOR bring-up
  - Host OS installation and configuration
  - Kubernetes and Cluster bring-up
  - Installation of all software components and booting up
  - Switch and OLT NOS installation and configuration
- After the POD bring-up the subscriber provisioning contributes to the overall time until all subscribers are fully operational. The performance depends on the factors discussed in the previous chapter.

11.8 SLAs and KPIs of Access 4.0

Find here a list of DTAGs desired KPIs and SLAs:

Category	Criteria	Target value
Commissioning	full POD bring-up (only SW part)	< 7h

Category	Criteria	Target value
Operations	full POD reboot	< 30min, w/o subs bring-up
	Central EMS uptime	> 99,99%
	POD uptime (Data-Plane)	> 99,99%
	POD uptime (Control-Plane)	> 99,999%
Service Availability	Mass-market / DTAG retail	97%, Period of clearance 24h
	Mass-market / Wholesale L2BSA	98,5%, Period of clearance 4h
	Business DCIP	99,5%, Period of clearance 8h
	Business Ethernet Connect	Period of clearance 8h, penalty
Scale	Customers per Leaf	> 22k (Architecture goal, see details below)
	Customers per POD	> 80k
	Subscriber session set-up rate after the establishment of Empty Session (ONT PON layer is fully established using OMCI)	> 100 sessions / sec
NF/Standards and Open Source Compliance	ONF compliant	All relevant components in the offered solution are aligned with ONF SEBA/VOLTHA architecture, interfaces and data models. The components such as OLT, VOLTHA OLT adapter, OLT device manager are enrolled/embedded into ONF's Continuous Certification Program, to the extent possible, (reference: https://opennetworking.org/continuous-certification-program) including the CI/CD development pipeline cycle of SEBA/VOLTHA project, and passes continuous real-time certification against the latest ONF SEBA/VOLTHA platform software.
Software quality	according to ISO 9126	

Figure 105 Access 4.0 desired SLAs

The achievable subscriber scale is a multidimensional restriction and cannot be defined by a single number. The chipset (of the Leaf) has limits in terms of queues, counters, buffer and schedulers. DT requires services with each 5 queues, or 4 queues and even 2 queues. The current SDK and chipset supports only a partitioning of 4 and 8 queues, means a service which requires 5 queues will be implemented based on 8 queues. With that the total scale of a Leaf depends on the distribution of the different services. Find one example here with the related scale requirements roadmap:

- Scale for consumer (retail + wholesale) incl. business customers produced like retail (under the assumption: 25% with 8 queues, 75% with 4 queues).
    - Goal YE2022: 12k subscriber
    - Goal mid 2023: 16k subscriber
    - Goal YE2023: min. 20k subscriber, architectural goal: 22k

11.9 TL9000 SLA Measurement/Performance KPIs

SLAs such as availability have to be collected, measured, stored, analyzed and reported. It has been decided by DT and Radisys that the initial approach will be to Monitor the Service impact Outage (SO) KPIs as a gauge of Availability. Based on the TL 9000 R5.7 Point Release the following requirements must be met:

1. Common Measurements
   a. Number of Problem Reports (NPR)
   b. Problem Report Fix time (FRT)
   c. Overdue Problem Report Fix Responsiveness (OFR)
   d. On Time Delivery (OTD)

2. Outage Measurements
   a. Service impact Outage (SO)
   b. Network Element impact Outage (SONE)
   c. Support Service Caused Outage (SSO)
   d. Mean Time To Restore Service (MTRS)
   e. Global Service Impact (GSI)

3. Hardware Measurements
   a. Field Replaceable Unit Returns (FR)
   b. Basic Return Rate (BRR)

4. Software Measurements
   a. Software Fix Quality (SFQ)
   b. Early Software Problem Report (eSPR)

5. Service Quality Measurements
   a. Service Quality (SQ)
   b. End-Customer Complaint Report Rate (CCRR)
   c. Incident Restore Rate (IRR)

Reported measurements shall apply to products and services only during the General Availability Phase of their Lifecycle.

11.9.1 Common Measurements

TBD – MS6

11.9.2 Outage Measurements

TBD – MS6

11.9.3 Hardware Measurements

TBD – MS6

11.9.4 Software Measurements

TBD – MS6

11.9.5 Service Quality Measurements

TBD – MS6

12 Central EMS

TBD – MS6 (Will be chapter 5)