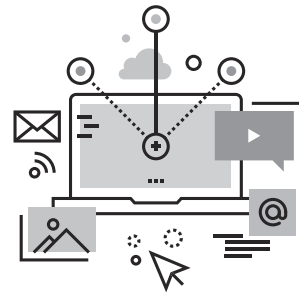


# 5G 초저지연 서비스 구현을 위한 NFV 기반 MEC 기술

이문원 다산네트웍솔루션즈 전략마케팅실 상무



## 1. 머리말

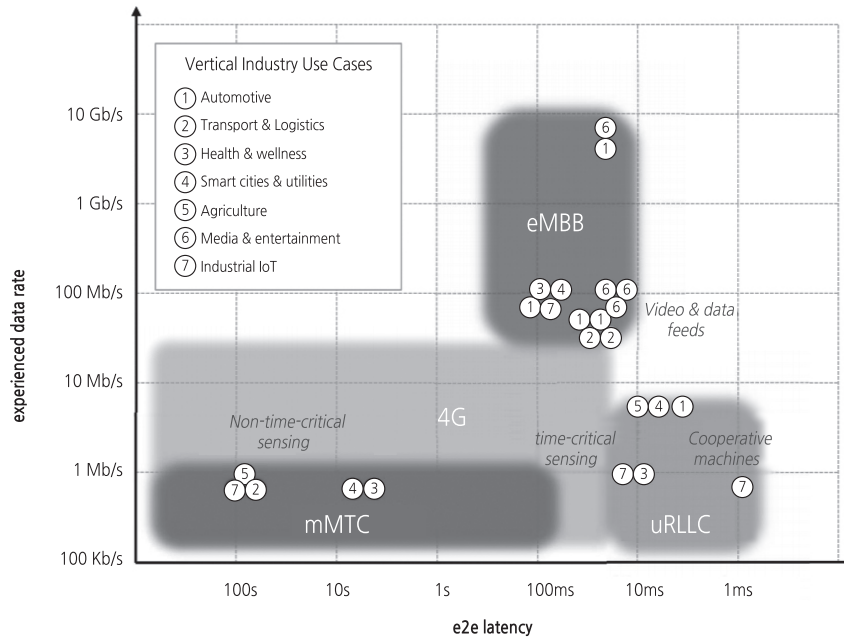
5G 무선통신 및 4차 산업혁명의 핵심 애플리케이션으로 자율주행차, 공장 자동화, 원격 진료, VR/AR과 같은 고신뢰 초저지연 통신(URLLC, Ultra Reliable and Low Latency Communication)을 요구하는 서비스들이 급부상하고 있다. 그러나 무선사업자망의 전통적 클라우드 구조는, 애플리케이션 서비스 포인트가 코어망 외곽에 위치하기 때문에, 통상 RTT(Return Trip Time) 10ms 이하의 초저지연 요구사항을 만족시키는 것이 물리적으로 어렵다. 반면 2세대 MEC(Multi-Access Edge Computing) 기술은 무선 액세스망의 S1 링크상에 애플리케이션 서비스를 위치시키는 분산 클라우드 컴퓨팅 구조를 채택하고, 네트워크 가상화(NFV, Network Function Virtualization)에 의한 프로세싱 지연을 최소화하는 하드웨어 기반 가속 기술을 적용함으로써 이같은 한계를 극복한다. 본고에서는 MEC 기술 트렌드를 분석하고, URLLC 요구사항을 만족시킬 현실적인 MEC 플랫폼 구현 방안을 제시해 본다.

## 2. MEC 급부상의 배경

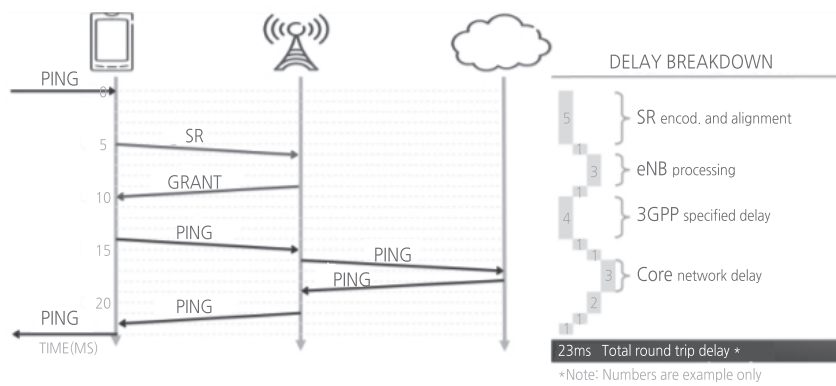
5G 무선통신 기술의 도입을 견인하는 애플리케이션들은 현 세대인 4G-Advanced 기술로는 지원이 불가능한, (단방향 End-to-End 연결 기준) 10-100ms 수준의 저지연과 10Mbps-10Gbps 수준의 초고속 전송을 요구하는 eMBB(extreme Mobile Broadband) 계열과 1-10ms 수준의 초저지연을 요구하는 URLLC 계열로 구분할 수 있다[1]. 이 중 MEC 호스팅의 대상으로는 자율주행차, 스마트 시티/공장/농장, 가상/증강 현실(VR/AR) 및 360도 비디오 애플리케이션 등을 대표적으로 꼽을 수 있다 ([그림 1] 참조).

특히 제4차 산업혁명의 도입을 추진하는 진영에서는 초연결 지능(Connected Intelligence) 확산을 위한 무선 통신망의 (단방향 E2E 기준) URLLC 요구사항을 좀 더 구체적으로 나열하고 있다. 자율형 교통수단 5ms, 공장 자동화 1ms, 고속열차 10ms, 원격 의료 서비스 1ms, 스마트 그리드 8ms, 가상/증강 현실 7ms[2].

그러나 통상 1ms 이하의 URLLC는 무선 단말과 기지국 간 패킷 전송이 이뤄지는 셀룰러(Cellular)



[그림 1] Diverging requirements – many capabilities beyond 4G performance, NGMN

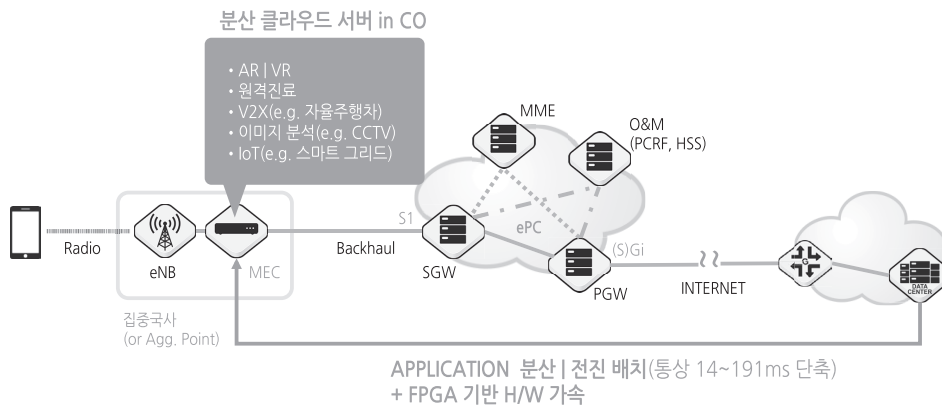


[그림 2] Latencies in 4G LTE Today, Ericsson, EuCNC 2016

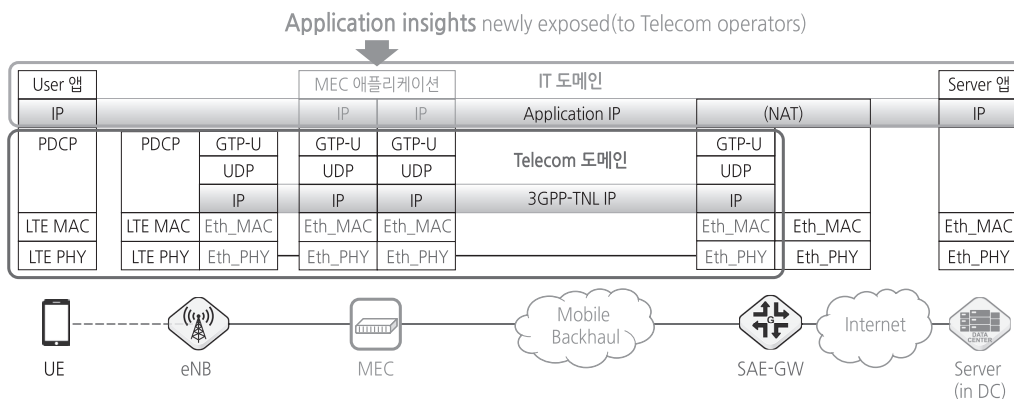
아키텍처로는 물리적인 수용이 어렵기 때문에, 대안 기술인 포그 컴퓨팅(Fog Computing) 커뮤니티에서 관련 연구가 진행 중이다. 전반적인 5G URLLC 요구사항을 논의하는 IEEE 및 NGMN 기술 그룹은 ①단방향 E2E 지연 5ms, ②전송 신뢰도 99.999%, ③이동성 지원 500Km/h를 대부분의 URLLC 서비스를 수용할 현실적인 합의점으로 제시

하고 있다[3].

이러한 애플리케이션별 지연 요소를 (현 LTE 네트워크의) 전송 구간별로 측정해 보면 두 가지 분석 결과를 도출할 수 있다. 첫째, E2E 네트워크 전송 지연보다 미들웨어, 애플리케이션 계층의 처리 지연이 지배적이며, 둘째, 기지국 처리 이후 백홀 및 코어망 지연이 최소 60% 이상(통상 14~191ms) 기여



[그림 3] MEC를 이용한 애플리케이션의 전진 배치



[그림 4] MEC에 의한, 무선통신 사업자의 애플리케이션 계층 정보 획득 및 처리

한다[4](그림 2 참조).

따라서 ITU-R 연구 그룹5의 'IMT-2020 요구사항 최종 권고(초안)'에 따라 5G 무선망의 User Plane 지연이 1ms 수준으로 줄어들 경우[5], URLLC 단방향 E2E 5ms 요구사항을 만족 시킬 현실적인 방안은, '애플리케이션 서비스 지점을 기지국과 인접하도록 물리적으로 전진 배치함으로써, 백홀망 이후의 지연 요소를 제거하는 것'이라는 결론에 이르게 된다. 이러한 전진 배치를 실현할 기술적 대안은, 적어도 현재는, MEC가 유일하다(그림 3 참조).

### 3. MEC 핵심 기술 - S1 트래픽 오프로드

MEC 기능의 핵심은, 탭(Tap) 또는 DIP(Deep Packet Inspection) 장비처럼, 모바일 백홀 구간 of S1 트래픽을 인라인(In-line) 방식으로 오프로드 한다는 점이다. 2세대 탭핑(Tapping) 장비인 NPB(Network Packet Broker)처럼 오프로드된 데이터에 추가적인 내부(on-premise) 프로세싱을 수행했으며, 최초로 적용된 애플리케이션은 캐싱(Transparent Caching)이었다. 그러다 NPB 기반 위에 다양한 애플리케이션을 호스팅하려는 시도가 범

<표 1> S1 트래픽 오프로드의 주요 기능 요소

Category	Description
U-Plane Support	<ul style="list-style-type: none"> <li>• <b>U-Plane interception</b> <ul style="list-style-type: none"> <li>-GTP-U header reconstruction(w/ PDCP SN &amp; GTP-U SN persistence)</li> <li>-TNL IP header reconstruction(w/ DSCP persistence)</li> </ul> </li> <li>• <b>Fast IP rerouting based on link supervision(e.g. w/ BFD)</b></li> <li>• <b>DSCP(in TNL IP)-based traffic bypass</b></li> </ul>
C-Plane Support	<ul style="list-style-type: none"> <li>• <b>Passive tapping on RRC &amp; NAS messages</b> <ul style="list-style-type: none"> <li>-Signaling and data radio bearers(SRBs &amp; DRBs), S1-AP, NAS messages</li> </ul> </li> <li>• <b>Monitoring on Cell Trace(LTE644 &amp; LTE459) interface</b></li> </ul>
Traffic Offload Policy (TOP)	<ul style="list-style-type: none"> <li>• <b>eRAB admission policy decision</b> <ul style="list-style-type: none"> <li>-Subscriber profile ID filter</li> <li>-QCI(Quality Class Indicator) filter</li> <li>-eRAB ARP filter</li> </ul> </li> <li>• <b>Packet admission policy decision</b> <ul style="list-style-type: none"> <li>-3-tuple filter</li> <li>(*Application-specific filters are off the TOP scope)</li> </ul> </li> </ul>
Security Support	<ul style="list-style-type: none"> <li>• <b>IPSec-based Backhaul Security</b> <ul style="list-style-type: none"> <li>-IPSec in tunnel mode, w/ Encapsulated Security Payload(ESP)</li> <li>-Encryption(e.g. AES-128-CBC), integrity protection(e.g. HMAC-SHA)</li> <li>-Connection negotiation(e.g. IKEv1), authentication(e.g. X.509)</li> </ul> </li> </ul>

용 애플리케이션 플랫폼 형태로 발전하게 되었다.

이러한 트래픽 오프로드 기능의 가장 본질적인 가치는, 무선통신 사업자로 하여금 그동안(GTP 트렁크에 둘러 싸여) ‘블랙박스’와 다름 없었던 S1 트래픽 내부의 IP/애플리케이션 정보를 들여다 보고, 사용자 정보의 무결성을 해치지 않는 선에서 추가적인 컴퓨팅/저장/네트워킹 처리할 수 있도록 지원한다는 점이다(그림 4) 참조).

S1 트래픽 오프로드 기술은 GTP 터널이 업링크, 다운링크에 서로 다른 TEID(Tunnel Endpoint Identifier)를 주소로 이용하기 때문에 상태유지(Stateful) 세션 관리 구현이 까다롭다. 핵심은, GTP-U 헤더를 복구하기 위해 PDCP SN(Packet Data Convergence Protocol Serial Number)과 GTP-U SN 간 매핑, 그리고 TNL(Transport Network Layer) IP 헤더를 복구하기 위해 DSCP(Differentiated Services Code Point) 매핑을 구현하는 것이다. S1 트래픽 오프로드 기능을 구성하는 주요 요소를 정리하면 <표 1>과 같다.

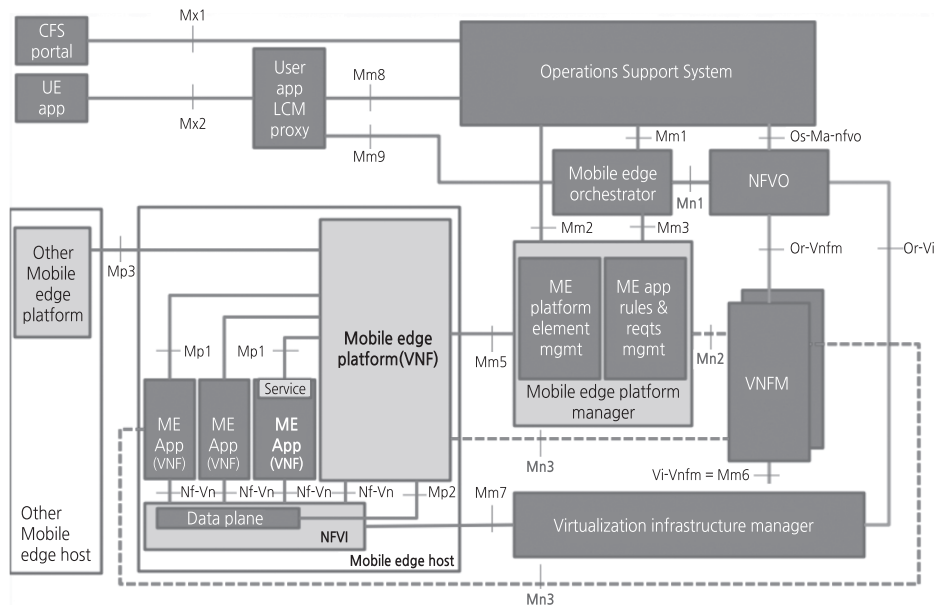
#### 4. MEC-NFV 간 아키텍처 통합

초기의 MEC 플랫폼은 애플리케이션 미들웨어로써 리눅스 호스트 OS상에 직접 설치되고, KVM/Xen 등 하이퍼바이저 기반 가상머신(VM)에 포팅된 OVA(Open Virtual Appliance) 포맷의 애플리케이션과 연동하는 베어메탈 서비스 형태였다.

그러나 유럽 표준화 단체인 ETSI 산하 MEC ISG(Industry Specification Group)에 의해 MEC 레퍼런스 아키텍처가 정의되고[6] ETSI NFV 표준과의 호환성이 강조됨에 따라, MEC 플랫폼 기능은 여타 애플리케이션들처럼 VNF(Virtual Network Function) 형태로서 VNFM(VNF Manager)의 통합관리를 받는 것이 일반화되고 있다[7].

#### 5. NFV 데이터 플레인 가속과 SDN 지원

MEC-NFV 간 호환성을 확보한 후에는, ① S1 오프로드 트래픽 처리 시 지연의 최소화, ② SDN(Software Defined Network) 컨트롤러와의



호환성 확보의 두 가지 숙제가 남는다. 이것을 동시에 해결하는 가장 일반적인 접근은 DPDK(Data Plane Development Kit) 기반 OVS(Open vSwitch) 가속이다. 구현을 위해서는 다음과 같은 요소가 필요하다.

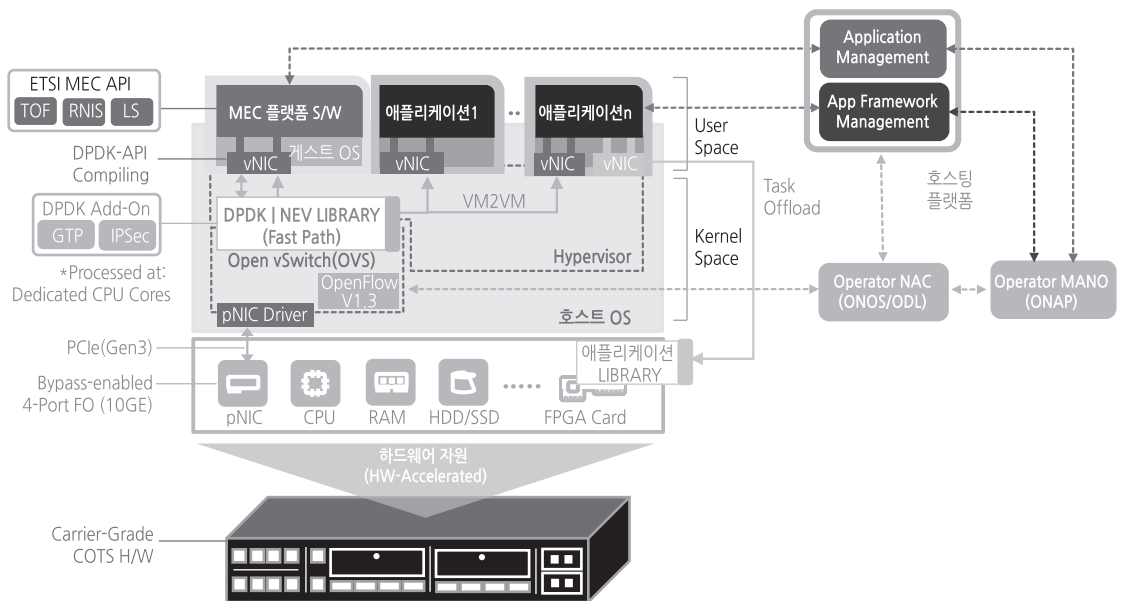
- ①DPDK 기능이 활성화 된 OVS
- ②DPDK 드라이버를 지원하는 NIC(Network Interface Card)
- ③PMD(Poll Mode Driver) 및 VT-d(Virtualization Technology for Directed I/O)를 지원하는 다중 코어 인텔 CPU
- ④DPDK 라이브러리를 이용하도록 컴파일링 된 애플리케이션

각 요소의 설치 및 설정은 다음 레퍼런스를 참고하도록 한다.

- CPU & BIOS settings, N/W configurations, Core pinning &

- Isolation[8]
- Linux for Real time – Installation, Reference & Tuning Guide[9]
- Open vSwitch – Installation, Running & Test[10]
- DPDK – Installation, Running & Test[11]

DPDK 기반 OVS 가속 방식은, 애플리케이션의 vNIC(virtual NIC), OVS 그리고 pNIC(physical NIC)이 모두, 커널 스페이스 드라이버 대신, 유저 스페이스의 DPDK 드라이버 및 메모리 영역을 공유하게 함으로써, ‘커널-호스트OS-하이퍼바이저-게스트OS’로 구성된 버추얼 패브릭을 완벽하게 회피한다(PMD가 활성화되면, DPDK 드라이버가 지정된 CPU 코어를 이용하도록 설정함으로써 성능을 더욱 높일 수 있다). 동시에 OpenFlow를 포함한 여타 SDN 프로토콜을 지원하는 OVS의 다양한 L2/L3 기능을 이용하는 한편, DPDK 라이브러리에 GTP,



[그림 6] NFW 기반 MEC 플랫폼 E2E 아키텍처

IPSec 처리 등 애드-온 기능을 추가함으로써 차별화를 추구할 수도 있다(그림 6 참조).

그러나 DPDK 기반 OVS 가속은, pNIC과 vNIC을 DMA(Direct Memory Access) 방식으로 연결하는 SR-IOV(Single Root I/O Virtualization)와는 달리, OVS 레이어의 프로세싱 지연으로 인해 베어메탈 수준의 'Wire-speed'를 달성하기가 쉽지 않고, 비용이 많이 들며, 특정 실리콘(인텔) 플랫폼에 종속될 수 있다는 단점이 있다. 대안으로 ARM 계열의 OFP(Open Fast Path) 오픈소스[12]를 이용하거나, 보다 기술 중립적인 CCIX(Cache Coherent Interconnect for Accelerator)[13]와 IOMMU(I/O Memory Management Unit)를 활용하는 방안을 고민해 볼 수 있다.

## 6. URLLC 지원을 위한, 추가적인 고려사항들

URLLC 요구사항인 단방향 E2E 5ms를 지원하


기 위한 NFW 기반 MEC 플랫폼의 데이터 플레인 성능(Latency) 목표는 100-400um 정도로 설정할 수 있으며, 이는 베어메탈 하드웨어에(가상화 기반 없이) MEC 플랫폼 S/W만을 설치했을 때 달성 가능한 이상적인 수치를 벤치마킹한 것이다. MEC 전문 벤더들은 이러한 성능에 근접하기 위해 DPDK 보다는 SR-IOV 채용이 유리하다는 공통적인 의견을 보였다[14](단, SR-IOV는 OVS 레이어를 회피하므로, SDN 컨트롤러와의 연동은 VDP(VSI Discovery and Configuration Protocol) Extension과 같은 비표준으로 구현한다는 가정).

데이터 플레인 가속을 최적화 한 후에도, 추가로 애플리케이션(VNF) 레이어의 처리 지연을 극복해야 한다. 이는 Assisted-Driving 이나 실시간 CCTV 응용 서비스와 같이 초저지연의 이미지/빅데이터 분석이 요구되는 경우에 특히 중요하다. 관련해서, MEC 커뮤니티 일부에서는 애플리케이션 프로세싱을 FPGA 기반 하드웨어로 오프로딩 하는 연구를

진행하고 있는 반면, 인텔은 NEV(Network Edge Virtualization) SDK를 출시하고 (값비싼 FPGA 카드 대신) 특정 CPU 코어를 활용하는 방안을 제시하고 있다[15]. 또한, NEV는 S1 트래픽 통계 정보를 추출해서 앱(VNF)에 제공하는 라이브러리를 포함한다.

## 7. 맺음말

결론적으로, 단순히 현 시점의 데이터 플레인 속도가 아닌 전체 개발 생태계의 양과 질, 그리고 그에 따른 기술 혁신의 속도와 경제성을 고려했을 때, URLLC 요구사항을 만족할 MEC 플랫폼 구현 방안은 ①DPDK 지원 가능한(x86) NFV 하드웨어 플랫폼을 선별하고, ②하드웨어와 리눅스 OS/하이퍼바이저를 Real-time 가이드라인에 따라 설치&튜닝하고, ③오픈소스 OVS 및 DPDK/NEV 라이브러리를 설치 & 튜닝하고, ④DPDK-컴파일링된 MEC VNF를 VM에 포팅하는 접근이 현재로서는 유리해 보인다.

이때, MEC용 NFV 하드웨어는 통신사망에 요구되는 (L2 Forwarding Consistency, Latency, Fast Link Fault Detection, High Availability와 같은) ‘Carrier-grade’ 기능[16]을 포함하되, C-RAN 집중 국사와 같은 ‘Near Edge’ 환경에 최적화된 용량과 확장성을 갖춰야 한다. 그리고 남은 문제는 오직 ‘비용’이 될 것이다. 

※이 논문은 2017년도 정부(과학기술정보통신부)의 재원으로 ‘범부처 Giga KOREA 사업’의 지원을 받아 수행된 연구입니다[No. GK17P0400, (초저지연-총괄/1세부)저지연 융합서비스를 위한 모바일 엣지 컴퓨팅 플랫폼 기술 개발].

## [참고 문헌]

- [1] ‘NGMN 5G Vision for Vertical Industries’, NGMN, 2016 Nov.
- [2] ‘5G Systems – Enabling the Transformation of Industry and Society’, Ericsson White Paper, 2017 Jan.

- [3] ‘5G Ultra-Reliable and Low Latency Communication’, IEEE Communications Theory Workshop, Ericsson, 2016 May.
- [4] ‘Ultra-Reliable and Low Latency 5G Communication’, European Conference on Networks and Communications, Ericsson, 2016 Jun.
- [5] ‘Minimum requirements related to technical performance for IMT-2020 radio interface(s)’, Draft New Report ITU-R M., 2017 Feb.
- [6] ‘Mobile Edge Computing Framework and Reference Architecture’, ETSI GS MEC 003 v1.1.1, 2016 Jun.
- [7] ‘MEC Integration with NFV’, ETSI MEC ISG, SDN NFV Congress, 2016 Oct.
- [8] Intel Network Builders – Techniques to maximize the performance of your NFV infrastructure on Intel Architecture: [builders.intel.com/university/networkbuilders/course/how-to-maximize-performance-of-nfv-on-intel-architecture](http://builders.intel.com/university/networkbuilders/course/how-to-maximize-performance-of-nfv-on-intel-architecture)
- [9] Red Hat Enterprise Linux for Real Time – Installation, Reference & Tuning Guide: [access.redhat.com/documentation/en-us/red\\_hat\\_enterprise\\_linux\\_for\\_real\\_time](http://access.redhat.com/documentation/en-us/red_hat_enterprise_linux_for_real_time)
- [10] Open vSwitch Documentation: [docs.openvswitch.org/en/latest/](http://docs.openvswitch.org/en/latest/)
- [11] DPDK Documentation, Performance Report & Training: [www.dpdk.org/doc](http://www.dpdk.org/doc)
- [12] Open Fast Path Foundation: [www.openfastpath.org](http://www.openfastpath.org)
- [13] CCIX Consortium: [www.ccixconsortium.com](http://www.ccixconsortium.com)
- [14] ‘DZS internal survey at MEC Congress Berlin’, 2017 Sep.
- [15] Intel NEV: [networkbuilders.intel.com/network-technologies/nev](http://networkbuilders.intel.com/network-technologies/nev)
- [16] ‘Carrier grade performance and Reliability in Network Virtualization(Tolly Report)’, WindRiver Whitepaper, 2016 Oct.: [events.windriver.com/wrcd01/wrcm/2016/12/tolly-report-hpe-nfv-system.pdf](http://events.windriver.com/wrcd01/wrcm/2016/12/tolly-report-hpe-nfv-system.pdf)

## [주요 용어 풀이]

- 포그 컴퓨팅(Fog Computing): 사용자 단말에서 발생한 데이터를 클라우드로 전송하는 대신, 로컬의 엣지 컴퓨팅(Edge Computing) 단말이 빠르게 처리하는 분산 클라우드 컴퓨팅 아키텍처. 엣지 컴퓨팅 단말은(주로 패킷 포워딩이 주 역할인) IoT 게이트웨이와 비교했을 때, 고성능의 스토리지, 통신, 제어 및 관리 기능을 갖는다는 특징이 있다.