

## Toplu Ulaşım Araçları için ITxPT Standardı ile uyumlu bir Bilgi Sisteminin Geliştirilmesi

Mehmet Burak Aydın<sup>1,\*</sup>, Can Öz<sup>1</sup>, Dilek Çetin Tulazoğlu<sup>1</sup>, Geylani Kardeş<sup>2</sup>

<sup>1</sup> Kent Kart Ege Elektronik San. ve Tic. A.Ş., İzmir, Türkiye

<sup>2</sup> Ege Üniversitesi Uluslararası Bilgisayar Enstitüsü, 35100, Bonova, İzmir, Türkiye

\*Correspondence: [burak.aydin@kentkart.com.tr](mailto:burak.aydin@kentkart.com.tr)

**Özet:** Toplu taşıma filoları ücret toplama, filo yönetimi, araç takip, araç içi ağlardan araç bilgilerinin toplanması ve araç içi yolcu bilgilendirmesi gibi temel süreçleri işletmektedir. Ancak bu süreçler günümüzde çoğunlukla birbiri ile standart olmayan iletişim protokolleri üzerinden haberleşmektedir. Çoğu zaman bu işlevlerin farklı üreticiler tarafından sağlanması ihtiyaç duyulan bilgi teknolojilerinin kurgulanmasını geciktirmektedir. Farklı üreticilerin sistem ve cihazlarının birbirleri ile uyumları sağlansa bile bunların kurulum süreçleri sorunlar oluşturmaktadır. Filolara akıllı ulaşım teknolojileri sunan firmaların sözü edilen karmaşık altyapının açık standartlar bütünü ile doğru şekilde yönetimini sağlamaları ve karşılaşılan uyum sorunlarını gidermeleri amacıyla yakın zamanda Toplu Ulaşım için Bilgi Teknolojisi (ITxPT) uluslararası standartları tanımlanmıştır. Ancak ITxPT standartları yeni olduğundan tanımladığı araç içi servislerin toplu taşıma sistemleri için uygulanmasına yönelik günümüzde çok az sayıda çalışma bulunmaktadır. Yukarıda sözü edilen eksiklikten yola çıkılarak bu çalışmada ITxPT servislerinin uygulaması için bir sistem mimarisi tanıtılmıştır ve ITxPT'nin önemli servisleri olan Envanter Servisi, GNSS konum ve VEHICLEtoIP servislerinin bir toplu ulaşım aracı için tasarım ve uygulaması gerçekleştirilmiştir. Önerilen mimariye göre tasarlanan ITxPT servislerinin toplu taşıma bilgi sistemleri üreten bir firmanın araç içi birimleri üzerinde başarıyla çalıştığı gözlenmiştir. Kurulan mimarinin Avahi sistemi, UDP Multicast ve HTTP REST Servisleri temelli kurgulanmış olması ITxPT servislerinin, üzerinde Linux çalışan bu birimlerde daha kolay ve hızlı bir şekilde yüklenmesini, çalıştırılmasını ve izlenmesini sağlamıştır.

**Anahtar Kelimeler:** Toplu ulaşım, Akıllı ulaşım sistemleri, ITxPT Standardı

**Abstract:** Public transportation fleets operate fundamental processes such as toll collection, fleet management, vehicle tracking, passenger information and collection of vehicle information from in-vehicle networks. However, these processes mostly communicate with each other over non-standard communication protocols. Often these functions are provided by different manufacturers, which causes delay during the construction of the required information technologies. Even if the systems and devices of different manufacturers are compatible with each other, their installation processes are challenging. The international Information Technology for Public Transport (ITxPT) standards have recently been defined in order to ensure that companies, providing intelligent transportation technologies to the fleets, are able to properly manage the complex infrastructure in question and to eliminate the problems of compliance. However, since the ITxPT standards are new, there are few studies on the implementation of onboard ITxPT services for public transport systems. In order to contribute these efforts, a system architecture for the implementation of ITxPT services is introduced in this study and the design and implementation of important ITxPT services, called Module inventory, GNSS location and VEHICLEtoIP for a public transportation vehicle has been realized. ITxPT services, designed according to the proposed architecture, have been observed to work successfully on the units of a company producing public transport information systems. The implementation which is based on Avahi System, UDP Multicast and HTTP REST services architecture, allows ITxPT services to be installed, run and monitored more easily and quickly on these Linux-powered units.

**Key words:** Public transport, Intelligent transportation systems, ITxPT Standard

\* Corresponding author.

E-mail address: [burak.aydin@kentkart.com.tr](mailto:burak.aydin@kentkart.com.tr)

ORCID: 0000-0003-0563-761X, 0000-0003-1412-5501, 0000-0003-4507-775X, 0000-0001-6975-305X (in hierarchical order)

Received 06/08/2019; accepted 14/09/2019

Peer review under responsibility of Bandirma Onyedi Eylül University.

All rights reserved.

## 1. Giriş

Toplu taşıma işletmeleri, sahadaki operasyonlarını daha verimli ve denetlenebilir bir şekilde yönetebilmek için akıllı ulaşım sistemlerine geçiş ihtiyacı duymaktadır [1]. Toplu taşıma filolarındaki bu teknolojik dönüşüm birçok avantaj getirdiği gibi sunulan sistemlerin karmaşık ve heterojen bir Bilgi Teknolojileri (BT) yapısına dönüşmesi sonucunu da doğurmaktadır. Mevcut saha koşullarında toplu taşıma filoları ücret toplama, filo yönetimi, araç takip, araç içi ağlardan araç bilgilerinin toplanması ve araç içi yolcu bilgilendirmesi gibi temel süreçleri işletmektedir. Ancak toplu taşıma filolarının bu süreçleri günümüzde çoğunlukla birbiri ile standart olmayan iletişim protokolleri üzerinden haberleşmektedir. Çoğu zaman bu işlevlerin farklı üreticiler tarafından sağlanması ihtiyaç duyulan BT'nin kurgulanmasını geciktirmektedir ve çeşitli iletişim protokollerinin uygulanması sistemin karmaşıklığını artırmaktadır. Farklı üreticilerin sistem ve cihazlarının birbirleri ile uyumları sağlansa bile bunların kurulum süreçleri sorunlar oluşturmaktadır. Bu gibi sorunlara çözüm olması amacıyla standart protokollere ve donanım ara yüzlerine ihtiyaç duyulmaktadır [2]. Akıllı ulaşım sistemlerinin en önemli uygulama sahalarından biri olan geleceğin akıllı araçlarının [3] geliştirilmesinde de bu standart protokollerin önemi giderek artmaktadır.

Filolara akıllı ulaşım teknolojileri sunan firmaların yukarıda sözü edilen karmaşık altyapının açık standartlar bütünü ile doğru şekilde yönetimini sağlamaları ve karşılaşılan uyum sorunlarını gidermeleri amacıyla yakın zamanda Toplu Ulaşım için Bilgi Teknolojisi (ing. Information Technology for Public Transport - ITxPT) uluslararası standartları tanımlanmıştır [4]. Uluslararası Toplu Taşıma Birliği (UITP) ve dünya çapında 100'den fazla otobüs üreticisi ve toplu taşıma işletmelerinin

birlikte oluşturduğu ITxPT girişimi ve buna ait standartlar toplu ulaşım BT'lerinin birlikte çalıştırılabilmesi, kolay konfigüre edilmesi, yönetilmesi ve ölçeklendirilmesi için endüstriyel düzeyde BT'lerin geliştirilmesine ait kural ve yöntemleri tanımlamaktadır.

Avrupa Birliği tarafından fonlanan daha önceki "Geleceğin Avrupa Otobüs Sistemleri" projelerinin (EBSF ve EBSF\_2) [2, 5] üzerine inşa edilen ITxPT standartlarının ilk sürümü yakın zamanda çıktığından (2017 yılı sonlarında) ITxPT araç içi servislerinin toplu taşıma sistemleri için uygulanmasına yönelik günümüzde çok az sayıda çalışma bulunmaktadır (örneğin [1, 6, 7]). Bu çalışmalarda da ITxPT servislerinin endüstriyel boyutta implementasyonuna yönelik bilgiler yeterli düzeyde değildir.

Yukarıda sözü edilen eksiklikten yola çıkılarak bu makalede ITxPT servislerinin uygulaması için bir sistem mimarisi tanıtılmakta ve ITxPT'nin önemli servisleri olan Envanter Servisi, GNSS konum ve VEHICLEtoIP servislerinin bir toplu ulaşım aracı için tasarım ve uygulanması anlatılmaktadır. Bu ITxPT araç içi servisleri Türkiye'nin önemli toplu taşıma bilgi sistemleri üreticisi firmalarından birindeki bir BT sistemi için uygulanmıştır. Çalışma bu yönüyle de bildiğimiz kadarıyla Türkiye'de ITxPT standartları üzerine gerçekleştirilen ilk çalışma olma özelliğini taşımaktadır.

Makalenin 2. bölümünde ITxPT ve sunduğu servislerin kısa bir incelemesi yer almaktadır. 3. bölümde geliştirilen sistemin mimarisi tanıtılmaktadır. ITxPT servislerinin uygulanması 4. bölümde anlatılmıştır. 5. bölümde bu alanda gerçekleştirilen önceki çalışmalar verilmiştir. Son olarak 6. bölümde sonuçlar ve ileriye yönelik planlanan çalışmalar yer almaktadır.

## 2. ITxPT

ITxPT, toplu taşıma araçlarında araç içi tüm sistem ve servisleri düzenleyen bir standarttır [4]. Bu standart cihaz üreticilerine bakılmaksızın tüm cihazların birbirleri ile uyumlu olarak çalışmasını sağlamayı amaçlamaktadır. Bu yöntemle toplu taşıma sistemlerinin tasarım ve uygulanmasında karşılıklı problemlerin giderilmesi hedeflenmektedir. Ayrıca akıllı toplu taşıma sistemleri üreten tüm şirketler kendi cihazlarını herhangi bir değişiklik yapmadan sisteme adapte edebileceklerdir [1].

ITxPT standardı içerisinde; Envanter Servisi (ing. Inventory Service), Zaman Servisi (ing. Time Service), GNSS Konum Servisi (ing. GNSS Location Services), VCG Servisi (Araç içerisindeki tüm diğer ekipmanlar için merkez ve/veya internet iletişimi sağlayan servis), FMStoIP Servisi (Yakıt Yönetim Sistemleri Bilgisini IP ağına aktarım servisi), VehicletoIP Servisi (Araç içi sensör bilgilerini IP ağına aktarım servisi), AVMS Servisi (Araç içerisinde sefer/durak/güzergah bilgisi sağlayan servis), APC Servisi (Araca binen/araçtan inen kişilerin sayıldığı ve bu sayıların sağlandığı servis), MADT Servisi (Sürücü Bilgi Ekranı Servisi) ve MQTT Servisi (MQTT Broker IP/PORT bilgisini sağlayan servis) servisleri yer almaktadır.

Ağ üzerindeki bu servisler farklı adreslerdeki (IP/port) üzerindeki birimler üzerinden sağlanabilmektedir. ITxPT bu servislerin otomatik olarak bulunmasını tariflemektedir ve yöntem olarak standart mDNS (Multicast DNS RFC 6762) ve DNS-SD (DNS-Based Service Discovery, RFC 6763) protokollerini kullanmaktadır. mDNS İnternet'te kullanılan unicast DNS'in aksine kapalı bir ağ içerisinde sunucu olmadan çalışmaktadır. Birim isimlerinin IP adresi dönüşümlerini sağlamaktadır. DNS-SD ise birimlerin üzerindeki servislerin detaylı bilgisini sunmaktadır. Tüm birimler mDNS/DNS-SD ile kendi servislerini ağ üzerinde yayınlamaktadır. Ağa yeni bağlanan bir birim

herhangi bir ön bilgiye sahip olmadan bu protokoller ile sorgu yaparak ağ üzerindeki servislerin listesini ve bunların IP/port bilgilerini edinecektir. Böylelikle araç içerisinde tak-çalıştır (ing. plug and play) yapısında bir uyumluluk altyapısı sağlanmış olur [8]. DNS-SD protokolünün paket yapısı RFC6763'te tanımlandığı üzere "SRV" ve "TXT" alanlarını içermektedir.

ITxPT standartlarında, paylaşılan servislerin içerdikleri SRV kayıtları aşağıdaki formatta olmalıdır:

```
<instance>._<service  
name>._<type>._<protocol>.<domain>
```

Örneğin bir GNSS konum servisi uygulaması için SRV kaydı aşağıdaki gibi tanımlanabilir [8]:

```
[hostname]._gnss_location._itxpt_multicast._udp.l  
ocal 3600 IN SRV 0 0 14005 [hostname]
```

TXT kayıtlarının ise aşağıdaki şablona uygun olması beklenmektedir:

```
<instance>._<service  
name>._<type>._<protocol>.<domain> TTL class  
TXT "name=value"
```

Yine bir GNSS konum servisi için TXT kaydı aşağıdakine benzer olacaktır [8]:

```
[hostname]._gnss_location._itxpt_multicast._udp.l  
ocal TTL class TXT "name=value"
```

## 3. Sistem Mimarisi

ITxPT servislerinin toplu ulaşım araçları içerisinde hayata geçirilmesi için bir dizi yazılım / donanım modüllerinin geliştirilmesi ve bağlantılarının kurgulanması gerekmektedir. IP ağları üzerinden birbirleri ile iletişim kuracak bu modüller ve içerikleri aşağıda verilmiştir:

**Zaman Servisi:** Bu servis SNTP (RFC4330) olarak tanımlanmıştır. Araç içerisindeki birimlerin birbirleri arasında saat senkronizasyonunu sağlamak amacı ile kullanılmaktadır.

**GNSS Konum Servisi:** Bu servis UDP Multicast olarak tanımlanmıştır. Araç içerisinde diğer sistemlerin kullanması için sağlanan konum servislerini ITxPT

direktiflerine uygun olarak dağıtan modüldür. Konum bilgisi araç içi sistemlerde önemli bir veri olarak kullanılmaktadır. Mevcut sistemlerde araç içerisinde çalışan her sistem GPS konum bilgisini kendi donanım ve antenleri ile sağlamaktadır. Araç içerisine aynı amaç için kurulan çok sayıda donanımdan kurtulmak için tek bir modül ile GNSS servisi gerçekleştirilir. Bu sistem konum bilgilerinin tek bir noktadan belirlenmesine ve araç içerisinde ihtiyaç duyan sistemlere servis edilmesine imkan verir. GPS bilgisinin tek bir noktadan alınabilmesi ile aynı zamanda kurulum ve tamirat gibi konularda da kolaylık sağlar. Araç içinde kullanılan birden çok anten donanımına ihtiyaç ortadan kalkmaktadır.

*VEHICLEtoIP Servisi:* Bu servis UDP Multicast olarak tanımlanmıştır. Araç içerisindeki sensörlerden gelen verilerin (kapı açık / kapalı, odometre sayacı, vb.) toplanarak Ethernet üzerinden diğer sistemlere servis olarak sunulmasını sağlayan yazılım modülüdür. RS232, I/O veya başka iletişim arayüzleri ile sağlanabilen verilere araç içindeki farklı sistemler tarafından ihtiyaç duyulabilmektedir. Bu sistemlerin her birinin araç içerisindeki veri kaynaklarına ulaşmaya çalışması gereksiz kablolama, yazılım, maliyet, verilerde tutarsızlık ve araç içi karmaşıklığa neden olmaktadır. Örnek: kapı açık/kapalı bilgisi ücret toplama sistemi tarafından kullanılırken, filo yönetimi tarafından bu bilginin LCD yolcu bilgilendirme ekranlarında da görüntülenmek istenmesi durumunda yolcu bilgilendirme uygulamasının kapı açık/kapalı bilgisini alabilmesi için ek bir uygulama geliştirilmesine ihtiyaç vardır. VEHICLEtoIP sistemi araç içerisinde bulunan bu gibi veri kaynaklarından çeşitli iletişim arayüzleri ile verileri toplayarak, diğer sistemlere bu verileri standart servislerle sunmayı hedefleyen bir yazılım modülüdür.

*AVMS Servisi:* Bu servis HTTP Rest protokolü ile tanımlanmıştır. Araç

içerisinde; güzergâh, durak ve bunun gibi sefer bilgilerinin diğer sistemlere (yolcu bilgilendirme vb.) servis olarak sunulmasını sağlayan yazılım modülüdür. Mevcut araç içi yolcu bilgilendirme sistemleri hat, yön vb. bilgileri standart olmayan yöntemlerle merkezi bir uygulamadan almaktadır. Aynı araç içinde farklı yolcu bilgilendirme modüllerinin (LED panel, LCD panel, vb.) kendi iletişimlerini kullanması gereksiz veri akışına neden olmaktadır. Bu tekrarlanan veri akışını önlemek amacıyla ITxPT tarafından belirlenen AVMS ile yayın yapılacak ve böylece yolcu bilgilendirme sistemleri yayımlanan bu bilgileri kullanabileceklerdir.

*MADT Servisi:* Bu servis HTTP veya VNC ile tanımlanmıştır. ITxPT standardına uygun modüllerden gelen veriler, geliştirilecek uygulama tarafından denetlenip, kontrol edilebilecektir. Sürücü ekranına sağlanacak arayüzle bilgiler anlık olarak görüntülenecektir. MADT yazılımı ITxPT dokümanlarında araç içinde bulunan tüm arabirimlerin anlık durumlarının kontrol edilebildiği arayüz olarak tanımlanmıştır. Geliştirilecek olan yazılımın farklı araç ekipmanlarına göre çalışabilen parametrik yapıda olması gerekmektedir.

*FMStoIP Servisi:* Bu servis UDP Multicast olarak tanımlanmıştır. Araç içerisindeki kontrol ağlarından (CAN) alınan bilgilerin diğer sistemlere servis olarak sunulmasını sağlayan yazılım modülüdür. Otobüs CAN verilerinin anlamlı hale getirilmesi araç çeşitliliğinin çok olması nedeniyle kompleks bir işlemdir.

*Ücret Toplama:* Ücret toplama cihazları ve servisleri bu modül kapsamında yer alır. ITxPT'nin mevcut tanımlamalarında ücret toplama sistemi ve yöntemleri henüz tanımlanmamıştır. Bizim önerdiğimiz sistemde bu modül mevcut ITxPT servislerine ek olarak getirilmiştir.

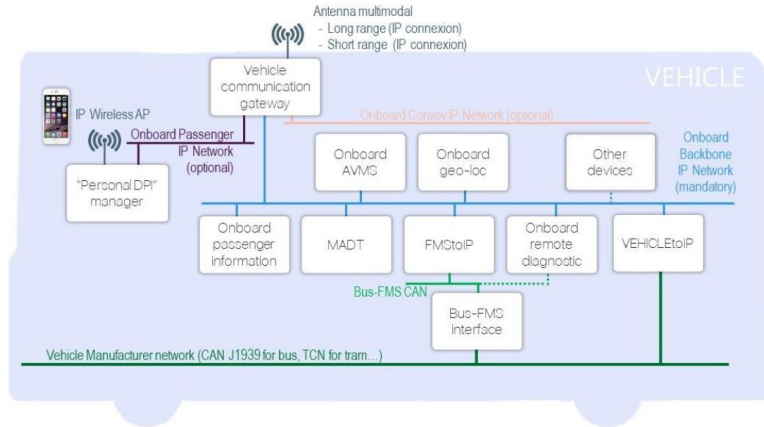
*MQTT Broker Servisi:* Bu servis MQTT Broker olarak tanımlanmıştır.

Publish/Subscribe olarak çalışan MQTT protokolü araç içerisinde hızlı akan veriler ve olay tetiklemeli uygulamalar için gerekli olmaktadır. Özellikle yolcu bilgi sistemi uygulamaları ve araç içi sensörlerden gelen veriler için bu protokolün kullanımı uygundur.

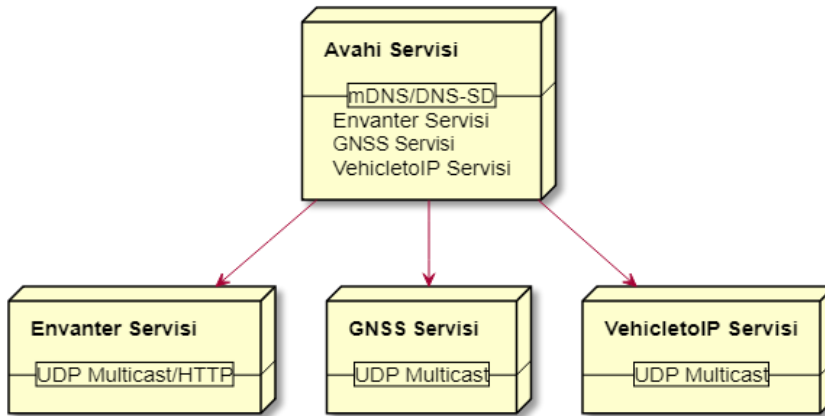
**Araç içi IP Ağı:** ITxPT araç içerisinde bir IP iletişim altyapısını zorunlu tutmaktadır. Araç içerisindeki her birim bu IP ağına bağlı olmalıdır. Şekil 1’de örnek araç içi konfigürasyon gösterilmiştir.

Bu çalışmada önerdiğimiz toplu ulaşım bilgi sisteminin ITxPT gereksinimlerini sağlaması için “multicast DNS” (mDNS) ve DNS-SD yapısını bünyesinde bulunduran Zeroconf teknolojisiinden yararlanılmıştır. İngilizce adından da

anlaşılabacağı üzere Zeroconf, bir sunucu hazırlama veya ayarlama ihtiyacı duymadan IP ağı oluşturmayı sağlayan bir teknolojidir [9]. Zeroconf teknolojisinin Linux işletim sisteminde gerçekleştirilmesi Avahi ile sağlanır. Avahi mDNS/DNS-SD protokolleri üzerinden yerel bir ağ içerisinde servislerin keşfini sağlamaktadır [10]. Avahi farklı amaçlar için birçok yardımcı yazılıma sahiptir. *Avahi-daemon* sistemde Zeroconf mimarisini gerçekleştiren ana uygulamadır. *Avahi-autoipd* uygulaması DHCP yokluğunda cihazlara dinamik IP atanması için kullanılır. *avahi-publish* servis paylaşımını sağlarken *avahi-browse* DNS-SD ile servis keşfi yapar. Şekil 2’de Avahi kullanımı üzerine kurgulanan sistemimizin mevcut servis mimarisi resmedilmiştir.



Şekil 1. IP Ağı içerisinde ITxPT modüllerinin iletişimi ([8]’den alınmıştır.).



Şekil 2. Gerçekleştirilen sistem servisleri.

Şekilde GNSS, VehicletoIP ve Envanter servisleri görülmekle birlikte bir önceki bölümde tanımlanan her bir servis modülünün mimariye dahil edilmesi Avahi uygulamaları ile mümkündür. Her bir servisin önerdiğimiz bu mimariye eklenmesi ile ilgili uygulama detayları makalenin bir sonraki bölümünde yer almaktadır.

#### 4. Uygulama

Bir önceki bölümde tanıtılan mimariye uygun olarak ITxPT servislerinin toplu ulaşım araçları için Avahi kullanımı üzerinden nasıl uygulamaya geçirileceği bu bölümde anlatılmaktadır. Örnekleme amacıyla bu çalışmada ITxPT'nin önemli servisleri olan Envanter Servisi, GNSS konum ve VEHICLEtoIP servislerinin tasarım ve uygulaması gerçekleştirilmiştir.

Sistem tasarımı ve uygulama çalışmaları T.C. Sanayi ve Teknoloji Bakanlığı'nca 2012 yılından beri akredite olan Kent Kart Ege Elektronik San. Tic. A.Ş. firmasının (kısaca Kentkart) Ar-Ge merkezinde gerçekleştirilmiştir. Kentkart otomatik ücret toplama, araç takibi ve yönetimi, gerçek zamanlı yolcu bilgilendirme, yol planlama ve araç-içi video gözetleme (ing. on-board video surveillance) gibi çeşitli alanlarda toplu taşıma bilgi sistemleri üreten, Türkiye'nin önemli bilgi teknolojileri firmalarından biridir. Kentkart'ın geliştirdiği toplu ulaşım sistemleri Türkiye'de 25 şehirde ve aralarında Amerika Birleşik Devletleri, Birleşik Arap Emirlikleri, Macaristan,

Pakistan, Polonya ve Sırbistan gibi ülkelerdeki şehirlerin de bulunduğu 10'dan fazla uluslararası lokasyonda kullanılmaktadır. Kentkart aynı zamanda toplu taşımada kullanılan ücret toplama makineleri, turnike validatörleri ve araç sürücüsü terminalleri gibi donanımları da üretmektedir. Bu çalışmadaki ITxPT servis uygulamalarında Kentkart'ın ürettiği toplu ulaşım donanımları kullanılmıştır.

Herhangi bir servisi sisteme dahil etmek için SRV ve TXT kayıtlarını içeren bir .service uzantılı dosyayı oluşturmak ve Avahi için "/etc/avahi/services" klasörüne eklemek yeterli olmaktadır. Avahi-daemon uygulaması bu uzantıdaki servisleri uygulama başlangıcında yayımlar. Servise ihtiyaç duyan cihaz avahi-browse ile tarama yapar ve istediği servisi avahi-resolve ile belirler.

Avahi-daemon yerel IP adresine kayıtları ve yerel programlar için iki adet Süreçler arası İletişim (IPC) Uygulama Programlama Arayüzü (API) sağlar. Bunlardan biri avahi-dnssconfd'dir. Bu uygulama elde edilen sunucu bilgilerini birden çok noktaya yayımlar. Diğeri ise nss-mdns'dir. Bu eklenti sayesinde isim çözümlemesi yapılır. Avahi-daemon uygulaması başlangıçta "/etc/avahi/service" klasörünün altındaki dosyaları kontrol eder ve doğru yazılan servisleri yayımlar. Örneğin GNSS konum servisi için hazırladığımız .service dosyası Şekil 3'te verilmiştir.

```
root@imx-kentkart:~# cat /etc/avahi/services/  
GNSS.service      Inventory.service  VehicletoIP.service  
root@imx-kentkart:~# cat /etc/avahi/services/GNSS.service  
<?xml version="1.0" standalone='no'?>  
<!DOCTYPE service-group SYSTEM "avahi-service.dtd">  
<service-group>  
  <name>_gnss_location.</name>  
  <service>  
    <type> itxpt_multicast._udp</type>  
    <port>14005</port>  
    <domain-name>local</domain-name>  
    <txt-record>multicast=239.255.42.21</txt-record>  
    <txt-record>version=1</txt-record>  
    <txt-record>txtversion=1</txt-record>  
  </service>  
</service-group>
```

Şekil 3. GNSS.service dosyasının içeriği.

Şekil 4'te .service dosyalarının Avahi-daemon tarafından işlenmesi sonucunda elde ettiğimiz çıktı verilmiştir. Şekildeki "Loading service file" satırlarında görüldüğü üzere "services" klasöründe bulunan GNSS.service, Inventory.service, VehicletaIP.service dosyalarının içerdikleri servisler uygulamanın başında yayınlanmaktadır. Dosyaların içeriklerinde herhangi bir hata ile karşılaşılmadığı için uygulamanın son satırlarında "Service successfully established." ibaresi bulunmaktadır.

Keşif özelliği ile birçok bilgi görülebilmektedir. Servisleri keşfetme ve keşfedilen servisleri çözme işlemleri avahi-browse tarafından yapılabilmektedir. Avahi-browse uygulaması başlatılırken birçok parametre verilebilir. Bu parametrelerden -a tüm servisleri göster, -r servisleri çöz, -l yerel servisleri gözardı et ve -p servisleri ayrıştırılabilir (";" ile ayrılmış) formatta göster anlamındadır. Şekil 5'te de görüldüğü üzere avahi-browse uygulaması -arlp parametresi ile çalıştırılmış ve servis bilgileri ";" işareti ile ayrıştırılabilir formatta servis bilgilerini

içeren SRV kaydı ve multicast IP'si içeren TXT kayıtları ile birlikte ekrana dökülmüştür.

Bu çalışmada Avahi, ARM mimarisi için çapraz derlenmiştir. Derlenen bu yazılımlar iki araç içi sürücü bilgisayar birimine (OBU) yüklenmiştir. Şekil 3'te gösterilen "GNSS.service" dosyası kayıtları hazırlanmıştır. Kayıtların yapısı makalenin 2. bölümünde anlatılan ITxPT SRV ve TXT biçimleri ile tam uyumludur. "GNSS.service" dosyası "/etc/avahi/services" klasörünün altına, avahi-daemon'ın başlangıçta yayınlaması için yerleştirilmiştir. Uygulama kodları C programlama dili kullanılarak hazırlanmıştır. GPS sensörü veri alınması için başlatılmış ve sonra iki cihaz arasında alınan TXT kayıtları ile bağlantı kurulmuştur. Veriyükü (ing. Payload) sensör verisi kullanılarak XML formatında oluşturulmuştur. Veriyükünün birden çoğa grup (ing. multicast group) şeklinde yayınlanması Şekil 6'da gösterilmiştir. Veriyükündeki *latitude*, *longitude*, *altitude* ve *time* verileri XML formatında ayrıştırılabilir veridir.

```
root@imx-kentkart:~# avahi-daemon
Found user 'avahi' (UID 104) and group 'avahi' (GID 111).
Successfully dropped root privileges.
avahi-daemon 0.6.22 starting up.
Loading service file /etc/avahi/services/GNSS.service.
Loading service file /etc/avahi/services/Inventory.service.
Loading service file /etc/avahi/services/VehicletaIP.service.
socket() failed: Address family not supported by protocol
Failed to create IPv6 socket, proceeding in IPv4 only mode
socket() failed: Address family not supported by protocol
Joining mDNS multicast group on interface eth0.IPv4 with address 139.122.100.1.
New relevant interface eth0.IPv4 for mDNS.
Network interface enumeration completed.
Registering new address record for 139.122.100.1 on eth0.IPv4.
Registering new address record for 192.168.1.10 on eth0.IPv4.
Registering new address record for 139.122.10.6 on eth0.IPv4.
Server startup complete. Host name is imx-kentkart.local. Local service cookie is 3250681428.
Service "_vehicletaip." (/etc/avahi/services/VehicletaIP.service) successfully established.
Service "_inventory." (/etc/avahi/services/Inventory.service) successfully established.
Service "_gnss_location." (/etc/avahi/services/GNSS.service) successfully established.
```

**Şekil 4.** avahi-daemon çalışma çıktısı.

```
root@imx-kentkart:~# avahi-browse -arlp
+;eth0;IPv4;_gnss_location\.;itxpt_multicast.udp;local
+;eth0;IPv4;_vehicletaip\.;itxpt_multicast.udp;local
=;eth0;IPv4;_gnss_location\.;itxpt_multicast.udp;local;imx-kentkart.local;139.122.0.64;14005;"txtversion=1" "version=1" "multicast=239.255.42.21"
=;eth0;IPv4;_vehicletaip\.;itxpt_multicast.udp;local;imx-kentkart.local;139.122.0.64;15030;"txtversion=1" "version=1" "multicast=239.255.42.21"
```

**Şekil 5.** avahi-browse parametrelerinin kullanımı.

```

Service: gnss_location.:
Payload is:
<?xml version="1.0" encoding="utf-8"?>
<GNSSLocationDelivery>
  <GNSSLocation>
    <latitude>54.098553</latitude>
    <longitude>18.764353</longitude>
    <Altitude>0</Altitude>
    <time>123523</time>
  </GNSSLocation>
</GNSSLocationDelivery>
Sending datagram message...OK
GPS_getpos okey.
Service: gnss_location.:
Payload is:
<?xml version="1.0" encoding="utf-8"?>
<GNSSLocationDelivery>
  <GNSSLocation>
    <latitude>54.099270</latitude>
    <longitude>18.765308</longitude>
    <Altitude>0</Altitude>
    <time>123533</time>
  </GNSSLocation>
</GNSSLocationDelivery>
Sending datagram message...OK
GPS_getpos okey.

```

**Şekil 6.** GNSS konum servisinin veri yükü.

Diğer cihazda işletilen programların kodlarının başında servisler keşfedilmekte ve çözülmektedir. İstenilen servisin SRV ve TXT kayıtları bir yapıya (ing. structure) atanmıştır. Atanan değerler Şekil 7’de gösterilmiştir. Program çıktısından görüldüğü üzere ihtiyaç duyulan multicast IP verisi başarıyla alınmıştır.

TXT kayıtları alınarak multicast grubu oluşturulduktan sonra uygulama GNSS konum servisi verisini almak için

beklemeye geçmiştir. Alınan veriler abone olunan cihazda ayrıştırılmıştır. Saniyede bir konum verisi alınmıştır. İlgili süreçlerin çıktıları ve ayrıştırılan verilerin içeriği Şekil 8’de yer almaktadır.

Önerdiğimiz mimariye göre uygulaması gerçekleştirilen bir sonraki ITxPT servisi VEHICLEtoIP’dir. Bu servis kullanılarak otobüs sürücüsü kapı durumu hakkında bilgilendirilebilir. Çalışmada kapı durumu anahtarlar (ing. switch) ile gerçekleşmiş ve bu bilgi sürücüye sağlanmıştır. Uygulama test ortamı Şekil 9’da görülmektedir. Ortamda 1 adet adaptör, 1 adet ayrıştırıcı (ing. splitter) ve 3 adet anahtar bulunmaktadır. Şebeke 220V gerilimini 24V’a düşüren adaptör kullanılmıştır. Düşürülen bu gerilim 3 farklı kapıyı temsil eden 24V anahtar bağlantısının gerilim hattına verilmiştir. Aradaki ayrıştırıcı ile hem anahtar bağlantısı hem de araç içi sürücü bilgisayarına güç ve Ethernet bağlantısı sağlanmıştır. 24V ve topraklaması yapılmış olan anahtarların veri kabloları microfit ile araç içi sürücü bilgisayarının farklı I/O girişlerine bağlanmıştır. Bu şekilde kapı bilgileri anahtar ile araç içi sürücü bilgisayarına aktarılmaktadır.

```

root@imx-kentkart:~# /usr/valapp/GNSS -G -sub
Selected Service:-G:
+;eth0;IPv4;_gnss_location\.;_itxpt_multicast._udp;local
+;eth0;IPv4;_vehicletoip\.;_itxpt_multicast._udp;local

selected_service:_gnss_location.:name:_gnss_location.:
Servicecount:0:
parsing_TXTRecords is finished...

txtversion:1
version:1
multicast:239.255.42.21
type:
model:
manufacturer:
serialnumber:
softwareversion:
hardwareversion:
macaddress:
status:0
xstatus:
submodules:false
path:
=;eth0;IPv4;_gnss_location\.;_itxpt_multicast._udp;local;imx-kentkart.local;139.122.0.64;14005;"txtversion=1"

```

**Şekil 7.** Çözülen GNSS konum servisi ve kayıtları.



```

wait for response!!!
Received: <?xml version="1.0" encoding="utf-8"?>
<GNSSLocationDelivery>
  <GNSSLocation>
    <latitude>54.100574</latitude>
    <longitude>18.767150</longitude>
    <Altitude>0</Altitude>
    <time>123554</time>
  </GNSSLocation>
</GNSSLocationDelivery>

MMessage:<?xml version="1.0" encoding="utf-8"?>
<GNSSLocationDelivery>
  <GNSSLocation>
    <latitude>54.100574</latitude>
    <longitude>18.767150</longitude>
    <Altitude>0</Altitude>
    <time>123554</time>
  </GNSSLocation>
</GNSSLocationDelivery>

Data
  :
Latitude      : 54.100574
Longitude     : 18.767150
Altitude      : 0
Time          : 123554
Date          :
Speed_over_ground : 0
Signal_quality :
Satellites_number : 0
Type          :
Coordinate_system :
wait for response!!!
Received: <?xml version="1.0" encoding="utf-8"?>
<GNSSLocationDelivery>
  <GNSSLocation>
    <latitude>54.100613</latitude>
    <longitude>18.767225</longitude>
    <Altitude>0</Altitude>
    <time>123604</time>
  </GNSSLocation>
</GNSSLocationDelivery>

```

**Şekil 8.** Saniyede bir paylaşılan konum verileri ve bu verilerin ayrıştırılması.

Uygulama çalıştırıldığında simüle edilen otobüs kapısının durumu güncellenmiş ve dinamik olarak XML formatında birden çoğa (ing. multicast) yayınlanmıştır. Birden çoğa yapılan yayında bulunan veri örneği Şekil 10'da görülmektedir. Her kapı için *Doorstatus* verisi anahtar durumuna göre anlık olarak güncellenmektedir. Her kapı durumu için karşılık gelen bir *Doorstatus* değeri bulunmaktadır. Bu *Doorstatus* değeri dinamik olarak değişmektedir. Kapılardan biri açık (ing. open) durumuna geçtiğinde bu değer *DoorOpen* olmaktadır ve sürücüye "açık bulunan kapı bulunmaktadır" bilgisi verilmektedir. Tüm kapılar kapalı ise *DoorClose* durumundadır.

Gerçekleştirilen diğer bir ITxPT servisi ise *Envanter Servisi*'dir. Bu servis çalışmamızda hem HTTP hem de TXT tabanlı olarak uygulamaya geçirilmiştir. Web sunucusu olarak basit, hafif ve kolay gerçekleşmesi özelliğinden dolayı *lighttpd* seçilmiştir. ARM cihazlarımızda *lighttpd* çapraz olarak derlenmiştir.



**Şekil 9.** VEHICLEtoIP servisi uygulaması için oluşturulan kapı durumu test ortamı.

```

Payload is:
<?xml version="1.0" encoding="utf-8"?>
<VEHICLEtoIPDelivery>
  <VEHICLEtoIP>
    <Odospeed>11</Odospeed>
    <Ododistancetotal>0</Ododistancetotal>
    <Ododistance>0</Ododistance>
    <Doorstatus>DoorOpen</Doorstatus>
    <Doorcontrol>
      <Door>
        <Doorid>3</Doorid>
        <Doorstatus>open</Doorstatus>
      </Door>
      <Door>
        <Doorid>4</Doorid>
        <Doorstatus>closed</Doorstatus>
      </Door>
      <Door>
        <Doorid>5</Doorid>
        <Doorstatus>open</Doorstatus>
      </Door>
    </Doorcontrol>
  </VEHICLEtoIP>
</VEHICLEtoIPDelivery>
Sending datagram message...OK

```

**Şekil 10.** Veriyükündeki kapı durumu.

Lighttpd uygulaması “/etc/lighttpd.conf” ayar dosyası ile ayarlanabilmektedir. Bu ayar dosyasında port numarası, kullanılan xml dosyası ve bu xml dosyasının yolu ayarlanmıştır. Bu ayarlar Şekil 11’de gösterilmiştir. HTTP 80 portunu kullandığı için 80 değeri girilmiştir. ITxPT standartlarında dosyanın *moduleinfo* adıyla bulunması gerekliliğinden dolayı *index-file.names* bu adla eşitlenmiştir.

```

root@imx-kentkart:/usr/valapp/lighttpd/etc/lighttpd# cat lighttpd.conf
server.document-root = "/usr/valapp/lighttpd/etc/index/"

server.port = 80

#server.username = "www"
#server.groupname = "www"

mime.type.assign = (
  ".html" => "text/html",
  ".txt" => "text/plain",
  ".jpg" => "image/jpeg",
  ".png" => "image/png"
)

static-file.exclude-extensions = ( ".fcgi", ".php", ".rb", "-.", ".inc" )
index-file.names = ( "index.html" )

$HTTP["host"] == "moduleinfo" {
  index-file.names = ( "moduleinfo.xml" )
}
root@imx-kentkart:/usr/valapp/lighttpd/etc/lighttpd#

```

**Şekil 11.** Oluşturulan *lighttpd.conf* dosyası.

lighttpd uygulaması Şekil 12’de gösterildiği gibi çalıştırılmıştır. “-f” parametresi ile ayar dosyasının uzantısı gösterilmektedir.

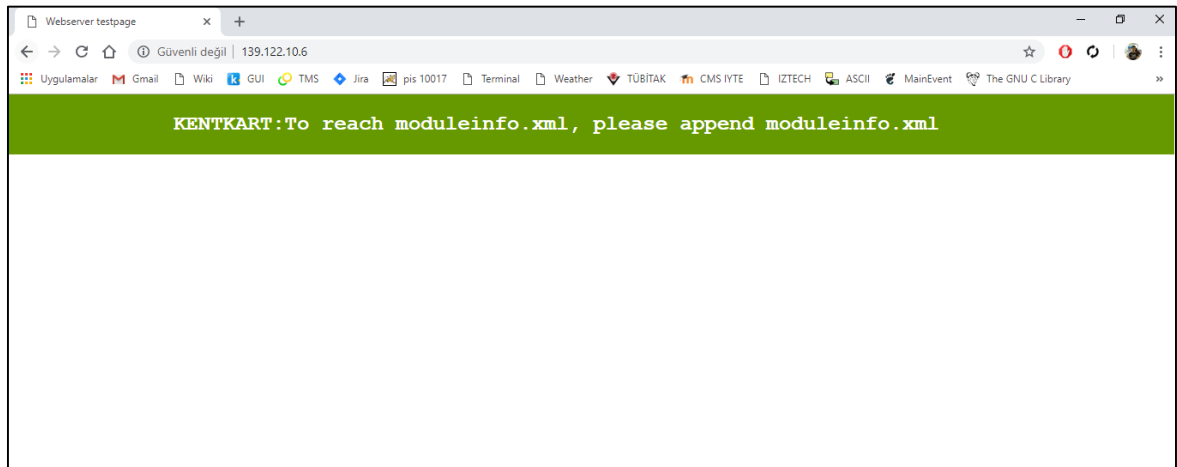
Daha sonra uygulamanın ana sayfası için gerekli *index.html* dosyası HTML kodları oluşturularak hazırlanmıştır. Web sunucusu bulunan cihaza bir web tarayıcısı üzerinden ulaşıldığında yazılan bu HTML kodları işletilerek Şekil 13’te görülen ana ekran ile karşılaşılacaktır.

```

root@imx-kentkart:/usr/valapp/lighttpd/sbin# ./lighttpd -f /usr/valapp/lighttpd/etc/lighttpd/lighttpd.conf

```

**Şekil 12.** *lighttpd* uygulamasının çalıştırılması.



**Şekil 13.** Web tarayıcısında görülen envanter servisi ana sayfası.

Şekil 13'te ekranda gösterilen mesaja uygun olarak 139.122.10.6/moduleinfo.xml web adresine erişildiğinde birim envanterinin yer aldığı "moduleinfo.xml" dosyası istenen bir cihaza indirilebilir ve Şekil 14'te gösterildiği gibi içeriği

görüntülenebilir. Araca kurulan her bir modülün otomatik olarak envanterinin oluşturulması sonrasında gerekli değişiklikler bu dosyaya yansıtılmaktadırlar ve ilgili ITxPT servisi ile diğer cihazlar söz konusu güncellemelerden haberdar olmaktadır.

```
root@imx-kentkart:/usr/valapp/lighttpd/etc/index# cat moduleinfo.xml
<?xml version="1.0" encoding="utf-8"?>
<modulesdelivery>
  <module>
    <type>Communication Gateway</type>
    <model>Model26</model>
    <manufacturer>CompanyABC</manufacturer>
    <serialNumber>0123456789</serialNumber>
    <softwareVersion>v1.2.3</softwareVersion>
    <hardwareVersion>Revision D</hardwareVersion>
    <macAddress>00:05:68:02:66:E5</macAddress>
    <status>0</status>
    <xstatus page="0" timestamp="01/17/19 - 14:55:18">C0FFFFFFFFFFFFFF</xstatus>
    <statuslist page="0">
      <item code="1">Generic module status</item>
      <item code="2">Consolidated Submodule status</item>
      <item code="3">Low uptime</item>
      <item code="4">High uptime</item>
      <item code="5">Free RAM memory</item>
      <item code="6">Free storage memory</item>
      <item code="7">root filesystem</item>
      <item code="8">application filesystem</item>
      <item code="9">filesystem load</item>
      <item code="10">Missing file</item>
      <item code="11">CPU load</item>
      <item code="12">Network load</item>
      <item code="13">Tampering</item>
      <item code="14">Backoffice connection</item>
      <item code="15">WAN connection</item>
      <item code="16">Configuration</item>
      <item code="17">Running applications</item>
      <item code="18">System temperature</item>
      <item code="19">Time sync</item>
      <item code="20">Power supply</item>
    </statuslist>
    <submodule>
      <type>FMS-interface</type>
      <model>ModelA79</model>
      <manufacturer>Company76G</manufacturer>
      <serialNumber>2642967405</serialNumber>
      <softwareVersion>sw:118, dcf: 111125</softwareVersion>
      <hardwareVersion>6605-21</hardwareVersion>
      <status>0</status>
      <xstatus page="0" timestamp="01/17/19 - 14:55:18">C0FFFFFFFFFFFFFF</xstatus>
    </submodule>
  </module>
</modulesdelivery>root@imx-kentkart:/usr/valapp/lighttpd/etc/index#
```

Şekil 14. moduleinfo.xml dosyasında kaydedilen cihaz bilgileri.

## 5. İlgili Çalışmalar

EBSF mimarisinin [2] farklı cihazları destekleyecek şekilde toplu ulaşım sistemlerinde uygulanması için ITxPT'nin önemi [11]'da vurgulanmıştır. Aynı çalışmada yeni kurulmakta olan ITxPT konsorsiyumunun özellikle yeni cihaz üreticilerinin EBSF mimarisini kullanmasını ve entegrasyonunu sağlayacağı belirtilmiştir. Benzer şekilde Heino ve ark. [12] toplu ulaşım

araçlarındaki BT altyapısının inşa edilmesinde karşılaşılan birlikte çalışabilirlik sorunlarının giderilmesinde ITxPT'nin önemli bir alternatif oluşturacağını belirtmişlerdir. Ancak bu her iki çalışmada da ITxPT standartlarının nasıl uygulanacağı ile ilgili bir yaklaşım sunulmamıştır.

Öte yandan ITxPT araç içi servislerinin geliştirilmesi ve uygulanması yönündeki çalışmalar ITxPT tanımlamalarının yakın

tarıhli olması nedeniyle ilgili literatürde de yeni ve az sayıdadır. Örneğın Puerta ve ark. [1] ITxPT mimarisinin mevcut araç içi BT mimarilerine göre farklarını ve avantajlarını anlattıktan sonra gerekli araç içi ağların fiziksel ve mantıksal uygulamalarını açıklamış ve NONVIS adı verilen ve sürücülere ekonomik sürüşte yardımcı olacak bir araç içi BT modülünü tanıtmışlardır. Bizim çalışmamızda olduğu gibi bu çalışmada da NONVIS içerisindeki servisin SRV tanımları örneklenmiştir ancak çalışmada ITxPT mimarisinin sadece IP ağ iletişimini içeren bir alt kümesi kullanılmıştır.

Otobüs filolarında tahminlemeye dayalı bir yönetim sisteminin oluşturulması [6] ve [13]'te anlatılmıştır. Geliştirilen sistem otobüsteki yağ seviyesi ve kalitesini kontrol ederek araçta bir hasar oluşmadan parça değişimlerinin ne zaman yapılabileceği hakkında bir kestirim yapmayı amaçlamaktadır. Söz konusu bakım sistemi içerisinde ITxPT DNS-SD iletişim protokolünün kullanıldığı belirtilmekle beraber bu protokolün uygulamasına yönelik detaylar [6] ve [13]'te yer almamaktadır.

Norveç'teki en büyük toplu ulaşım otoritesi olan Ruter bünyesindeki BT'lerde ITxPT'nin uygulanması ile ilgili yakın zamanda gerçekleştirilen bir çalışma [7]'de tanıtılmıştır. Bu çalışma ile otobüs yolculuklarının planlanmasında kullanılan uygulamaların gerçek zamanlı verileri kullanabilmesi ve böylece araç kalkış zamanlarının anında bildirilmesi amaçlanmıştır. Mevcut durumda Ruter'a bağlı olan işletmeler araç içerisindeki bu verilere sahiptir ancak Ruter bu bilgilere doğrudan erişememektedir. Çözüm için ITxPT mimarisi üzerine kurgulanmış ve bir Servis olarak Ulaşım (ing. Transportation-as-a-Service) (TaaS) adı verilen bir sistem geliştirilmiştir. Çalışma özellikle ITxPT'nin MQTT köprülemesinin iyi bir örneğini temsil etmektedir ancak mevcut dokümanlarda ITxPT altyapısı ile gerekli mesajlaşmanın

nasıl sağlanacağına dair uygulama örnekleri verilmemiştir.

Bu makalede tanıttığımız ITxPT servisleri mimarisinin yukarıda özetlenen bu alandaki az sayıdaki çalışmaya ITxPT araç içi servislerinin toplu taşıma araçlarında uygulanması ile ilgili adımları tanımlaması ve örnekleme açısından bir katkı verdiği inaniılmaktadır. Ayrıca bildiğimiz kadarıyla bu çalışma Türkiye'de ITxPT standartlarının toplu ulaşım sistemlerinde kullanılmasına yönelik gerçekleştirilen ilk çalışmadır.

## 6. Sonuç

Bu çalışmada ITxPT servislerinin uygulaması için bir sistem mimarisi tanıtılmıştır ve Envanter, GNSS konum ve VEHICLEtoIP servislerinin bir toplu ulaşım aracı için tasarım ve uygulaması gerçekleştirilmiştir. Önerilen mimariye göre tasarlanan bu ITxPT servislerinin bir toplu taşıma bilgi sistemleri üreticisi olan Kentkart firmasının araç içi birimleri (OBU) üzerinde başarıyla çalıştığı gözlenmiştir.

Tasarlanan mimarinin Avahi Sistemi, UDP Multicast ve HTTP REST servisleri temelli kurgulanmış olması ITxPT servislerinin, üzerinde Linux çalışan OBU'larda daha kolay ve hızlı bir şekilde yüklenmesini, çalıştırılmasını ve izlenmesini sağlamıştır. GNSS konum ve envanter servislerinin uygulanması haricinde otobüs sürücülerinin kapı durumlarındaki değişimler hakkında anlık bilgilendirilmesini sağlayan bir ITxPT VEHICLEtoIP uygulaması da yine bu çalışma kapsamında hayata geçirilmiştir.

Sistemin Avahi temelli kurgusu sayesinde mevcut mimariye başka bir araç içi servis, bu servisin DNS-SD keşfini ve yayınlanmasını destekleyecek "SRV" ve "TXT" kayıtlarının oluşturulmasından sonra yine ITxPT standardına uygun olarak kolaylıkla dahil edilebilir.

İleriye yönelik planlanan ilk çalışmada tasarımı bu makalede tanıtılan sistem mimarisi kullanılarak araç içerisinde

güzergah ve durak bilgilerinin ITxPT iletişim kanalına yollanmasını sağlayan AVMS servisinin uygulaması gerçekleştirilecektir. Daha sonra MADT ve Ücret Toplama servislerinin mevcut sistemimize ITxPT tanımlarına uygun olarak dahil edilmesi planlanmaktadır.

### Teşekkür

Bu çalışma Avrupa Birliği EUREKA EURIPIDES etiketli ve Türkiye Bilimsel ve Teknolojik Araştırma Kurumu (TÜBİTAK) Teknoloji ve Yenilik Destek Programları Başkanlığı (TEYDEB) tarafından desteklenen ... no'lu ... başlıklı uluslararası sanayi Ar-Ge projesi kapsamında gerçekleştirilmiştir. *Proje no ve adı derginin double-blind review kuralları gereği boş bırakılmıştır.*

### Kaynaklar

- [1] Puerta, J., Brazalez, A., Suescun, A., Iparraguirre, O., Atutxa, U. Standardizing IT Systems on Public Transport: An Eco-Driving Assistance System Case Study. *13th International Workshop on Communication Technologies for Vehicles (Nets4Cars 2018/Nets4Trains 2018/Nets4Aircraft 2018)*, 2018. Madrid, Spain, 17-18 May 2018. Lecture Notes in Computer Science 10796: 149-158.
- [2] Tozzi, M.; Bousse, Y.; Karlsson, M.A.; Guida, M. A European Initiative for More Efficient and Attractive Bus Systems: The EBSF\_2 Project. *Transportation Research Procedia*. 2016, 14, 2640-2648.
- [3] Tektaş, M.; Tektaş, N. Akıllı ulaşım sistemleri(AUS) uygulamalarının sektörlere göre dağılımı. *Akıllı Ulaşım Sistemleri ve Uygulamaları Dergisi*, 2019, 2, 1, 32-41.
- [4] ITxPT. *Open IT architecture for Public Transport*. Accessed on: September 22, 2019. [Online]. Available: <https://itxpt.org/>
- [5] Corazza, M. V., Guida, U., Musso, A., Tozzi, M. From EBSF to EBSF\_2: A compelling agenda for the bus of the future: A decade of research for more attractive and sustainable buses. *2016*

*IEEE 16th International Conference on Environment and Electrical Engineering (EEEIC 2016)*, 2016. Florence, Italy, 7-10 June 2016.

- [6] Corazza, M. V.; Magnalardo, S.; Musso, A.; Petracci, E.; Tozzi, M.; Vasari, D.; de Verdalle, E. Testing an innovative predictive management system for bus fleets: outcomes from the Ravenna case study. *IET Intelligent Transport Systems*, 2018, 12, 4, 286-293.
- [7] Vig, C., *Traffic-as-a-Service: Plug-and-play IT systems for public transport*, Ruter, Norway, 2018.
- [8] Information Technology for Public Transport, *S02 - Onboard Architecture specification, Release S02v2.0.1*, ITxPT, Belgium, 2018.
- [9] Cheshire, S.; Steinberg, D. H. *Zero Configuration Networking: The Definitive Guide*. O'Reilly, USA, 2005.
- [10] Easttom, C.; Palladino, S. *Essential Linux Administration: A Comprehensive Guide for Beginners*. Course Technology, USA, 2012.
- [11] Sciandra, V., *Toward unifying on-board Intelligent Transportation Systems architecture in public transports*. Ph.D. Dissertation, Université Paris-Est, France, 2013.
- [12] Heino, I., Kostianen, J., Lahti, J., Linna, J., Pihlajamaa, O. Living Lab Bus platform for IoT service development in public transport context. *25th ITS World Congress*, 2018 Copenhagen, Denmark, 17-21 September 2018.
- [13] Corazza, M. V., Mangialardo, S., Petracci, E., Tozzi, M., Vasari, D.; de Verdalle, E. Research and Innovation in Predictive Management for Bus Fleets: The Ravenna Case Study. *7th Transport Research Arena (TRA 2018)*, 2018. Vienna, Austria, 16-19 April 2018.