

Toplu Taşıma Sistemlerinde Kullanılan Nesnelerin İnterneti Yazılımlarının Geliştirmesinde Model-güdümlü Mühendislik İhtiyacı

The Need for Model-driven Engineering in the Development of IoT Software for Public Transportation Systems

Sadık Arslan
Ar-Ge Departmanı
Kent Kart Ege Elektronik A.Ş.
İzmir, Türkiye
sadik.arslan@kentkart.com.tr

Geylani Kardeş
Uluslararası Bilgisayar Enstitüsü
Ege Üniversitesi
İzmir, Türkiye
geylani.kardas@ege.edu.tr

Özetçe—Nesnelerin İnterneti (IoT) sistemlerinin geliştirilmesinde her zaman gözlenen problemlerin yanı sıra toplu taşıma sistemlerinde kullanılan IoT yazılımlarının geliştirilmesi sırasında bu alana özel ihtiyaçların da göz önüne alınması gerekmektedir. Bu bildiride, IoT-tabanlı toplu taşıma sistemlerinin geliştirilmesinde karşılaşılan problemler tanımlandıktan sonra IoT geliştirme için mevcutta önerilen model-güdümlü mühendislik (MDE) yaklaşımları incelenerek bunların toplu taşıma sistemleri için uygulanabilirliği sorgulanmıştır. Belirlenen eksiklikleri gidermek için bu alana özel yeni MDE yaklaşımlarının neleri içermesi gerektiği ile ilgili görüşler aktarılmıştır. Ayrıca, IoT tabanlı toplu taşıma sistemlerinde yüksek mobilite, özel standartlar ve araç içi haberleşme hattı kullanımı ve farklı çip desteğinin sağlanması için yeni MDE yaklaşımlarının gidiş-dönüş mühendisliğini de içerecek şekilde kurgulanmasının gerekliliği ortaya konulmuştur.

Anahtar Kelimeler—IoT, toplu taşıma, model-güdümlü mühendislik

Abstract—The specific requirements of developing Internet of Things (IoT) software for public transportation systems should be considered in addition to the problems always observed in IoT system development. After defining the problems encountered in the development of IoT-based public transport systems, the current model-driven engineering (MDE) approaches for IoT development are examined and their applicability for public transport systems is questioned in this paper. Several points of view are given about what new MDE approaches specific to this field should include in order to eliminate the identified deficiencies. Moreover, it is indicated that new MDE approaches should be designed including round-trip engineering to provide high mobility, use of special standards, in-vehicle communication networks and support for different system-on-chips in these public transportation systems.

Keywords—IoT, public transportation, model-driven engineering

I. GİRİŞ

Günümüzde toplu taşımada farklı sistemler sıklıkla kullanılmaktadır. Bunlar, ücret toplama, kirlilik ölçümü, yolcu sayma, taşıt sistem yönetimi, yolcu bilgilendirme, reklam

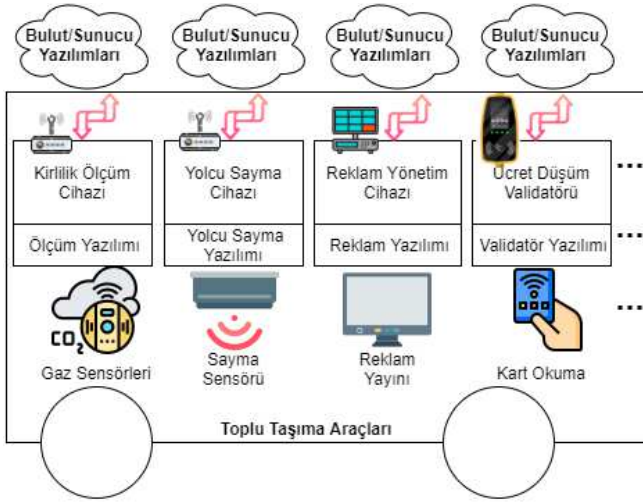
yönetimi, araç takibi gibi geliştirilen sistemlerdir. Bu sistemler birlikte çalışabileceği gibi tek başına da çalışabilmektedir. Toplu taşıma sistemleri otobüs, metro, tren, tramvay gibi çeşitli ortamlarda kullanılabilir. Bu ortamlarda sistemler yüksek sıcaklık farkları, sürekli titreşim, yağmur, güneş ve sürekli hareketin getirdiği zorlu koşullara maruz kalmaktadırlar.

Nesnelerin İnterneti (IoT) araştırma alanı, günümüzün en umut verici alanlarından biridir. IoT, akıllı şehirler, toplu taşıma, sanayi, üretim, tarım, sağlık gibi birçok alanda kullanılmaktadır. IoT, nesnelerin birbirine ve İnternet'in geri kalanına bağlı olduğu bir yapıdır. Buradaki “Nesneler”, İnternet'e bağlanabilen ve bilgi alışverişi yapabilen herhangi bir birimdir. IoT'de nesneler birbirine bağlıdır ve ayrıca veri alışverişi yapabilmektedirler [1].

Toplu taşıma IoT sistemlerinde genel IoT sistemlerine benzer şekilde heterojenlik oldukça önemli bir problemdir. Donanım ve yazılımın birbirine bağımlılığı yüksektir. Bu sistemlerde platforma bağımlı yazılımların geliştirilmesi oldukça yaygındır. Geliştirilen yazılımın, aynı işi yapan farklı bir platformda doğrudan çalıştırılması aynı programlama dili veya yazılım geliştirme araçları kullanılsa bile oldukça zordur.

Toplu taşımada kullanılan IoT sistemlerinde ayrıca bu alana özel problemler bulunmaktadır. Bunlar, IoT sistemlerinin farklılıkları, sürekli hareket halindeki sistemler olması, zorlu fiziksel koşullarda çalışma, alana özel “System-on-Chip” (SoC) kullanımı, alana özel donanımların kullanımı ve alana özel standartların (örneğin ITxPT [2]) kullanımı olarak özetlenebilir.

Toplu taşımada bulunabilecek bazı IoT sistemlerinin genel gösterimi Şekil 1'de verilmiştir. Bu basit çizimden de görülebileceği gibi toplu taşıma IoT sistemlerinde heterojenlik ve karmaşıklık oldukça yüksek olabilmektedir. Toplu taşımadaki araçların içerisinde ve bulut/sunucu ortamında donanım ve yazılımların çeşitliliği fazladır. Sistemdeki her birimin kendine has geliştirme ortamı ve ayarlamaları mevcuttur. Özellikle uç birimlerdeki bellek ve işlemci güçleri çok kısıtlıdır.



Şekil 1. Toplu taşımada kullanılabilir IoT sistemi örnekleri

Birçok farklı alanda başarılı uygulamaları bulunan model-güdümlü mühendislik (MDE) toplu taşıma IoT sistemlerinin geliştirilmesi için de kullanılabilir. Ancak yukarıda özetlenen farklardan dolayı genel amaçlı MDE yaklaşımlarının doğrudan toplu taşıma sistemlerindeki IoT yazılımlarının geliştirilmesinde kullanılması zor görünmektedir.

Bu çalışmada IoT-tabanlı toplu taşıma sistemlerinin geliştirilmesinde karşılaşılan ve bu alana özel konumda bulunan problemler tanımlanmaktadır. Ayrıca, IoT geliştirme için mevcutta önerilen MDE yaklaşımları incelenmiş ve bunların toplu taşıma sistemleri için uygulanabilirliği bu bildiride sorgulanmıştır. Ortaya çıkan eksiklikleri gidermek için bu alana özel yeni MDE yaklaşımlarının neleri içermesi gerektiği ile ilgili görüşler aktarılmıştır.

Bildirinin 2. bölümünde IoT sistemlerinin ortak problemleri özetlenmiştir. Bölüm 3'te toplu taşıma alanına özel IoT problemleri tanıtılmıştır. Bölüm 4'te alanla ilgili mevcut çalışmalar yer almaktadır. Bölüm 5 ve 6 sırasıyla IoT tabanlı toplu taşıma sistemlerinin geliştirilmesinde karşılaşılan problemler için çözüm önerilerini ve sonuçları içermektedir.

II. IOT SİSTEMLERİNDEKİ ORTAK PROBLEMLER

IoT sistemleri tıp, tarım, savunma, toplu taşıma gibi birçok farklı alanda yaygın bir şekilde kullanılmaktadır. Birbirine bağlı olan ve cihazların uzaktan kontrol edilmesine olanak tanıyan sistemlerin tasarımı giderek daha da karmaşık hale gelmektedir [1]. IoT sistemlerinin ortak bazı genel problemleri vardır. Önceki çeşitli çalışmalarda da (örneğin [3 - 5]) yaygın bir şekilde adreslenen bu problemler kısaca aşağıda listelenmiştir:

- Heterojenlik: Birlikte çalışan farklı türde birçok donanım ve yazılımın bulunması
- Cihazların kullanılabilirliği: Sürekli değişen bir ortamda cihazların mevcut olmaması veya kullanılma müsaitliklerinin (ing. availability of devices) az olması
- Büyük miktarda veri: Yüksek sayıda birimden sürekli veri üretilmesi ve bunun yönetimi
- Güvenlik ve mahremiyet: Sistem güvenliğinin ve kullanıcı mahremiyetinin sağlanması

Yukarıdaki gibi problemlerin çözülebilmesi için bazı IoT kavramları [5 - 8] ortaya çıkmıştır. Bu kavramlar ve kısa tanımları aşağıda verilmiştir.

- Cihaz tanımlama: IoT sistemlerindeki cihazlar sensörden okuma, aktüatör tetikleme gibi farklı işlemleri yapmaktadırlar. Cihazların tanımlarının doğru yapılması heterojenliğin üstesinden gelmek ve tasarımın doğru yapılması için önemlidir.
- Keşif: Sürekli değişen bir ortamda cihazların kullanılabilirliği sorunu ele almaktadır. Örneğin, yeni eklenen katılımcılar, cihazların ve hizmet süreçlerinin keşfedilmesiyle kaydedilebilir.
- Yerleştirme: Çalışabilir yazılım bileşenlerinin uygun ara yazılım (ing. middleware) ve donanıma yerleştirilmesidir. Yerleştirme, nesnelerin donanım yazılımını değiştirme olasılığıyla ilgilidir (kod yamaları, donanım eklentileri gibi).
- Kendi kendine adaptasyon: Bir sistemin, ortamda bir değişiklik olduğunda kendini adapte etme gücüdür. IoT sisteminde herhangi bir birimin bozulması, pil bitmesi gibi nedenlerle sistemden çıkması durumunda, sistemin otomatik olarak tepki vermesi gerekmektedir. Eksik birimin yerine diğer birimlerin özelliklerinin değiştirilmesi ile problemin üstesinden gelmek de bu kapsamda incelenebilir.
- Servis yapıları: IoT cihazlarında donanım soyutlamasının sağlanabilmesi için veri transferi ve IoT işlevlerinin servisler aracılığıyla yapılmasıdır.
- Bulut/Sis hizmetleri: Bulut, farklı kaynaklardan gelen verileri birleştirmek ve ağın ucunda çalışan gömülü cihazların sınırlı hesaplama gücünün üstesinden gelmek için oldukça önemlidir. Uç cihazlar ile merkezi bulut hizmetleri arasında gönderilen veri miktarını azaltmak veya belirli bir uzak aramada yanıt süresini azaltmak için kullanılan yerel bulut hizmetlerine sis denir. Zaman açısından kritik veriler, ağın ucunda ve/veya yerel sis kullanılarak işlenebilir.
- Ara yazılım: IoT cihazlarının ve iletişim protokollerinin heterojenliğini gizlemeyi ve etkileşim amaçları için homojen bir arayüz oluşturmayı amaçlamaktadır.

III. TOPLU TAŞIMAYA ÖZEL IOT PROBLEMLERİ

Genel IoT sistemlerindeki problemler toplu taşıma IoT sistemlerinde de bulunmaktadır. Özellikle heterojenliğin karmaşıklığı toplu taşıma IoT sistemlerinde çok daha yüksektir. Toplu taşıma alanında heterojenlik, cihazların kullanılabilirliği, büyük miktarda veri, güvenlik ve mahremiyet gibi konularda da özelleşmiş alt problemler bulunabilmektedir. Aşağıda bu gibi problemler ve tanımları kısaca maddelendirilmiştir.

- Yüksek mobilite: Şehir içinde hareketli sistemlerdir. Bu hareketli yapıdan kaynaklı olarak kablosuz ağ sistemlerinin seçimi ve yönetimi oldukça önemli olmaktadır. Sistem tasarımı esnasında bu yapıların bağlantısı, sistemden çıkıp tekrar sisteme dahil olmalarının yönetimi gibi birçok işleme dair farklı problemler önem kazanmaktadır.

- Özel standartlar: ITxPT [2] gibi toplu taşıma alanındaki sistemlerin tasarımlarının nasıl yapılması gerektiğini anlatan özel standartlar mevcuttur. Bu özel standartların kullanılması sırasında IoT uygulamalarının geliştirilmesinde sorunlar yaşanabilmektedir. Bununla birlikte toplu taşıma IoT alt sistemlerinden olan taşıt yönetim sistemlerinde ise “Controller Area Network” (CAN) hat yapısı kullanılmaktadır. Genel bir IoT sisteminde sensörlerden veri analog veya sayısal olarak okunabilmekteyken, toplu taşıma sistemlerindeki yapıda sensörlerden veri CAN hattı aracılığı ile alınabilmektedir. CAN hattında motor devri, yakıt seviyesi, hız gibi birçok sensör verisi bulunmaktadır. Alınan veri, analiz ve görüntüleme amacıyla sistem yöneticileri tarafından kullanılmaktadır.
- SoC kullanımı: Piyasada sık karşılaşılan Arduino, Raspberry Pi, Intel Edison gibi platformların toplu taşımada kullanım ihtimali oldukça düşüktür. Toplu taşımada kullanılabilir olan donanım platformları genellikle iMX, Sitara gibi SoC mikroişlemcileri ile tasarlanmış olan özel sistemlerdir. Genel tasarım amaçlı geliştirilen MDE tekniklerinin bu tip özel tasarlanmış sistemler ile uyumluluğu oldukça düşüktür.
- Özel donanım kullanımı: Sadece toplu taşıma alanına özel donanım kullanım örnekleri bulunmaktadır. Örneğin ücret toplama sistemlerinde, yolculardan ücret tahsil edilmektedir. Ücret tahsil işlemi farklı yollarla yapılabilmektedir. Örneğin, “Radio Frequency” (RF) kart, kredi kartı, “Quick Response” (QR) kod, akıllı cep telefonu gibi tekniklerle ödemeler yapılabilmektedir. Bu ödemelerin güvenliği için IoT sistemlerinde “Secure Access Module” (SAM) kartı gibi sistem ile entegre güvenlik donanımları bulunmaktadır.
- Özel protokol kullanımı: Toplu taşımadaki IoT sistemlerinde tasarım yapılırken alana özel Modbus, RS485 gibi protokoller kullanılmaktadır. Örneğin, toplu taşımada kullanılan yolcu bilgilendirme ve reklam yönetim gibi IoT sistemlerinde “Light Emitting Diode” (LED) paneller bulunmaktadır. Bu paneller yolcular için yeni durak bilgilendirmesi ve reklam yayını sağlamaktadır. LED panellerin çalıştırılabilmesi için toplu taşımaya özgü protokoller mevcuttur [9].
- İşletim sistemi seçimi: İşletim sistemleri IoT sistemlerinin geliştirilmesinde yüksek seviyede soyutlamayı sağlamada oldukça yardımcı olmaktadır. Ubuntu Core, BalenaOS, RIOT, Contiki, TinyOS, Zephyr, Raspian, Android Things, Apache Mynewt, Fuchsia, Windows 10 IoT, TizenRT, Amazon FreeRTOS, Embedded Linux, Mbed OS gibi çok sayıda işletim sistemi bulunmaktadır. Ancak bu işletim sistemlerinin toplu taşıma özelinde kullanılabilmesi için özel ihtiyaçların yönetilmesi gerekmektedir. Hem toplu taşımadaki kaynak kısıtlı ortamın desteklenebilmesi hem de toplu taşıma standartlarının ve donanımlarının desteği için özel işletim sistemi dağıtımlarına ihtiyaç bulunmaktadır. Özellikle Linux türevi işletim sistemlerinin oluşturulabilmesi için kullanılan Yocto gibi araçlar toplu taşıma alanına özel işletim sistemi dağıtımının elde edilmesini sağlamaktadır.

IV. İLGİLİ ÇALIŞMALAR

Bu bölümde ilk olarak, genel IoT sistemlerinin geliştirilmesi için MDE öneren çalışmalar incelenmiştir. İkinci başlıkta, literatürdeki IoT tabanlı toplu taşıma sistemleri için MDE çalışmaları anlatılmıştır.

A. Genel IoT için MDE Çalışmaları

IoT geliştirme zorluklarının üstesinden gelinebilmesi için MDE yaklaşımları umut vaat etmektedir. Yerleştirme ve heterojenlik gibi temel IoT problemlerine odaklanan bir yaklaşım “Internet of Things Modelling Language” (ThingML) kullanımını önermektedir [10]. ThingML, durum çizelgeleri ve bileşen diyagramları gibi görsel modelleme yapılarını ve mimari seviyeden bireysel cihazların davranışına kadar platforma bağımlı bir modelleme dili aracılığıyla farklı bakış açılarından IoT uygulamalarının modellenmesini destekler. Arduino, Raspberry Pi, Intel Edison gibi platform bağımlı kod üretimi ve popüler programlama dilleri (C/C++, Java, Javascript) desteklenir. ThingML gibi, Vorto projesi de benzer bir yaklaşıma sahiptir [11]. Vorto “Domain Specific Language” (DSL), farklı ayrıntı düzeylerinde araçların işlevlerini ve özelliklerini tanımlayan üreticiden bağımsız soyutlama katmanlarını belirtmek için kullanılmaktadır.

Sistem çalışma zamanında kendi kendine adaptasyonu sağlamak için bazı çalışmalar bulunmaktadır. Örneğin [6]’da otomasyon veya geliştirme yazılımlarının yeniden kullanılabilirliği gibi konular üzerinde MDE teknikleri ile çalışılmıştır. Benzer şekilde, görev-kritik (ing. mission-critical) IoT sistemlerinin mühendisliğine özel olarak odaklanılan farklı bir çalışma [5]’te sunulmuştur. Bu sistemler ile doğrulama amacıyla modellerden yararlanılarak çözülebilecek güvenilirlik, emniyet ve güvenlik gibi gereksinimler başka zorlukları ortaya çıkarmaktadır [12].

Endüstriyel IoT üretim sistemleri için geliştirilmiş UML4IoT isimli MDE geliştirme çerçevesi [13]’te tanımlanmıştır. “Unified Modelling Language” (UML) tabanlı bu çalışmada bir üretim sistemindeki IoT bileşenleri kullanılmıştır. Benzer şekilde [14,15]’te de endüstri alanında farklı IoT için MDE uygulamaları gerçekleştirilmiştir.

COMFIT [16], otomatik IoT uygulamalarını geliştirmek için bir uygulama yönetimi ve yürütme modülünün bulut arayüzünü kullandığı ve MDE altyapısına dayanan bir uygulama geliştirme paradigmasını sunmaktadır.

MDE ve servis-odaklı mimari (SOA) kullanmayı içeren [17]’deki metodoloji yazılım bileşenleri oluşturmada farklı düzeylerde soyutlamalar sağlamaktadır. Böylece IoT’de ağ kaynakları uygulama programlama arabirimleri (API’ler) kullanılarak dinamik olarak tahsis edilir. Benzer şekilde [18]’de SOA tabanlı IoT geliştirme yaklaşımı göz önüne alınmıştır.

Telsiz Duyarga Ağlarının (WSN) uygulanmasında TinyOS ve Contiki gibi işletim sistemleri oldukça yaygın kullanılmaktadır. WSN alanındaki donanım düğümleri ve yazılım platformları standart hale gelmiştir. Bu alanda MDE tekniklerinin kullanımı (örneğin [19, 20]) IoT’nin diğer alanlarına göre daha başarılıdır.

Yukarıdakilere ek olarak farklı alanlara özel IoT için MDE çalışmaları bulunmaktadır. Bunlar için örnekler hızlı prototip oluşturma [21], akıllı ev [22], sağlık [23], akıllı şehir [24], kapı geçiş kontrolü [25] ve otomotiv [26] sistemlerinden verilebilmektedir.

B. Akıllı Ulaşım ve Toplu Taşıma ile İlgili MDE IoT Çalışmaları

Yapılan araştırmalar göstermiştir ki, doğrudan toplu taşıma IoT problemlerine odaklanıp bunların çözümü için geliştirmiş bir MDE çalışması henüz bulunmamaktadır. Toplu taşıma ile ortak çalışma alanları olabilecek olan akıllı ulaşım alanında ise kısıtlı da olsa bazı öneri ve küçük vaka çalışmaları bulunmaktadır. Bu bölümde bu çalışmalar anlatılmıştır. Her çalışma eğer özel isimlendirilmiş bir araç/yaklaşım/metodoloji öneriyorsa anlatımda bu ismin kullanılması tercih edilmiştir. Diğer durumlarda doğrudan çalışmayı gerçekleştiren yazarların isimleri kullanılmıştır.

IADev [18]: Öznitelik odaklı tasarımı ve MDE ve SOA yapılarını birleştirmektedir. Platforma özgü kod üretebilen bir sistem sunulmuştur. Genel amaçlı geliştirilmiş bu çerçevenin vaka çalışması bir akıllı ulaşım sistemi üzerinden yapılmıştır. Burada daha çok trafik yönetimi ve yol güvenliği üzerinde durulmuştur.

Hause et al. [24]: Akıllı şehirlerdeki IoT sistemlerinde MDE sistem mühendisliği yaklaşımı konu edilmektedir. Trafik ışıkları, sensörler, aydınlatma ışıkları gibi bileşenler için tüm sistemin nasıl modellenebileceği, karmaşıklığın yönetimi ve sistemler arası senkronizasyon konuları ele alınmıştır. Toplu taşımada IoT'nin çok önemli olduğu belirtilmiştir. MDE yaklaşımları ile IoT sistemi geliştirmenin kolaylaştırılabileceği belirtilmiştir.

Iovino et al. [25]: Entegre kart tabanlı erişim kontrol sisteminin, "Near Field Communication" (NFC) etiketlerine dayalı olarak insanları yetkilendirmesi hakkında bir vaka araştırmasıdır. Bir IoT altyapısında yarı otomatik kod oluşturmayı destekleyen bir MDE yaklaşımı önerilmiştir. NFC ile geçiş yetkilendirme alanı için bir çözüm verilmiştir. Burada kullanılan NFC sistemi sadece kapılarda erişim için kullanılacak sistemler olarak düşünülmüştür. NFC sistemlerinin benzerleri, toplu taşıma sistemlerinden birisi olan ücret toplama kullanılabilmektedir. Ancak bu çalışmadaki yapı toplu taşıma sistemlerine özel olarak düşünülmemiştir. Toplu taşımada NFC kullanımı ile ücret düşümü SAM modülleri denilen özel donanımsal yapılarla birlikte kullanılmaktadır. Böylece ücret toplama işlemi güvenli hale gelmektedir. Ayrıca NFC ile ödeme tüm ücret toplama sisteminin sadece küçük bir kısmını oluşturmaktadır.

ENACT [27]: ENACT DevOps (EDO) çerçevesi, DevOps servis kalitesinin sürekliliğini ve IoT sistemlerinin geliştirme yaşam döngüsünün en iyi şekilde uygulanmasını amaç edinmiştir ve bu yönde gerekli bileşenleri tanımlamıştır. Ayrıca ENACT aracının bir test yatağı (ing. testbed) ortamında kullanılırken yaşanan zorluklarından bahsedilmiştir. Bu zorluklar temelde IoT platformlarının yaşamış olduğu heterojenlik gibi sorunlardır. DevOps açısından bir yaklaşım sunan ENACT, yine benzer IoT problemlerini sadece geliştirme açısından değil, sahada operasyonları ile birlikte inceler. Bu çalışmadaki test yatağı akıllı ulaşım sistemlerinde kullanılabilmektedir. Trenlerde kullanılan bu sistem ile tren kontrolü IoT sayesinde yapılmıştır.

UPECSI [28]: Bu çalışmada bulut tabanlı gizlilik gereksinimlerini yerine getirmesi hedeflenen bir araç sunulmaktadır. Genel IoT problemlerine odaklanılmıştır. Bu çalışmada, geliştirilen sistemin kullanım alanlarından birisinin akıllı ulaşım sistemleri olabileceği söylenmiştir.

Patel and Cassou [29]: Genel IoT problemlerine odaklanarak bir sistem tanıtmaktadır. Bu sistemin taşımacılık alanında WSN uygulaması için uyarlanabileceğinden bahsedilmiştir.

Vitruvius [26]: Programlama bilgisi olmayan kullanıcıların birbirine bağlı araçlardan ve sensörlerden gerçek zamanlı veri tüketimine dayalı zengin Web uygulamaları tasarlayıp hızlı bir şekilde uygulayabileceği bir platform olan Vitruvius sunulmaktadır. Araçlar üzerine takılabilecek olan bu platform için MDE geliştirme ortamı sağlayan bir sistem gerçekleştirilmiştir. Genel amaç, ne tip araç olursa olsun (ör. otomobil, kamyon), CO₂ miktarı, hız, lokasyon bilgisi gibi sensör bilgilerini merkezde toplayıp takip edilmesini sağlayacak sistem tasarımının MDE ile yapılmasıdır. Vitruvius genel olarak mobil araçlardan, bir platform kullanılarak sensör verisini toplama çalışmasıdır.

SecKit [12]: IoT cihazları için bir yönetim çerçevesine entegre olan, kullanıcı verilerinin korunmasını sağlamak amacıyla güvenlik politikalarının spesifikasyonunu ve verimli değerlendirmesini destekleyen ve MDE içeren bir araç seti önerilmiştir. Akıllı şehirler için kullanıcı verisinin korunmasını sağlayan bir IoT çerçevesi sunulmuştur. Temel hedef yine genel bir IoT geliştirme çerçevesi ortaya koymaktır. Ayrıca, genel kavramları içeren akıllı şehir vaka çalışması yapılmıştır.

Conzon et al. [30]: Yenilikçi bir genel IoT için MDE geliştirme araç seti tanıtılmıştır. Bu araç seti, anlamsal keşif hizmetinden yararlanarak, mevcut kaynakları veya cihazları dinamik olarak seçip konumlandırmaya izin verir ve geliştiricilerin karma uygulamalar oluşturmaya olanak tanıyan bir grafik arabirim de dahil olmak üzere esnek bir araç sağlar. Burada genel olarak endüstriyel uygulamalar için geliştirilen bir araç seti tanıtılmıştır. Çalışmada araç setinin toplu taşımada da kullanılabileceği öneri olarak belirtilmiştir.

Mazzini et al. [31]: IoT sistem bileşenlerinde kullanılacak gerçek zamanlı, güvenlik ve performans gibi işlevsel olmayan özellikleri içeren etkili ve verimli koruma mekanizmaları sağlamak için bir MDE metodolojisi sunulmuştur. Genel amaçlı bir IoT için MDE geliştirme yöntemi sunulmaktadır. Bu aracın akıllı ulaşım sistemlerinde bulunan veri yerleştirme sistemlerinde kullanıldığından bahsedilmiştir.

V. TARTIŞMA VE ÖNERİLER

Tüm yapılan literatür taramalarından da görülebileceği gibi, genel IoT ve bazı alana özgü alanlarda MDE teknikleri bulunmaktadır. Akıllı ulaşım alanı için kapsamı dar vaka çalışmaları bulunmakta iken toplu taşıma alanında anlamlı olabilecek herhangi bir MDE çalışmasına henüz rastlanılmamaktadır. Toplu taşıma IoT sistemlerinin geliştirme problemleri üzerinde durulan ve MDE teknikleri içeren herhangi bir detaylı çalışma da yapılmamıştır. Ancak, akıllı şehir ve toplu taşıma ile ilişkilendirilebilecek olan 10 adet çalışma için bildirinin 2. bölümünde verilen IoT problemleri ve çözüm kavramlarını karşılaştırmak gelecekte yapılabilecek çalışmalar için faydalı ve yol gösterici olabilecektir. Bu karşılaştırma Tablo 1'de verilmiştir. Tablodan görülebileceği gibi IoT problemleri arasında üzerinde en fazla MDE içeren çalışma yapılan konular heterojenlik ve güvenlidir. Bu konular sadece toplu taşımaya özel problemler değildir. IoT'nin doğası gereği heterojenlik ve güvenlik problemleri en çok karşılaşılan problemlerdir. Toplu taşıma çalışmaları arasında en az çalışılan problemin büyük veri olduğu görülmüştür. Tablo 1'de ayrıca toplu taşıma ile

ilişkilendirebilecek çalışmalarda kullanılan IoT çözüm kavramları kısmı bulunmaktadır. Bu kısımda görülebileceği gibi en çok kullanılan çözüm kavramları yerleştirme ve servislerdir. Yerleştirmenin önemi heterojen sistemlerde oldukça yüksektir. Çünkü zaten donanım-yazılım bağımlılığı yüksek olan heterojen sistemlerde yerleştirme çok dikkatli ve platforma uygun olacak şekilde yapılmalıdır. Yamaların yapılması ve yeni özelliklerin sistemlere eklenmesi ile MDE çalışmalarının kullanılması sorunların azaltılması konusunda çok yardımcı olmaktadır. Servis yapılarının kullanımının yüksek olması da yine heterojenlikle ilişkilendirilebilir. Servisler sayesinde soyutlamada daha başarılı yapılar kurulabilmektedir.

TABLO I. MEVCUT YAKLAŞIMLARIN KARŞILAŞTIRILMASI

| Çalışma | Karşılaştırılan özellikler | | | | | | | | | | |
|-----------------------|----------------------------|-----------------|------------|--------------|---------------------------------|-------|-------------|-------------------|---------------|-----------|-------------|
| | Adreslenen IoT Problemleri | | | | Kullanılan IoT Çözüm Kavramları | | | | | | |
| | Heterojenlik | Cihaz kullanımı | Büyük veri | Güv. ve mah. | Cihaz tanımı | Keşif | Yerleştirme | Kendi kendine ad. | Servis yapısı | Bulut/Sis | Ara yazılım |
| IADev [18] | X | X | | X | | X | X | X | X | | X |
| Hause et al. [24] | | | X | X | X | | X | X | | X | |
| Iovino et al. [25] | X | X | | X | | | X | | X | | |
| ENACT [27] | X | | | X | X | | X | X | X | | |
| UPECSI [28] | X | | | X | | | | X | X | X | |
| Patel and Cassou [29] | X | X | | | X | | X | | X | | X |
| Vitruvius [26] | X | | | X | | X | X | X | | | |
| SecKit [12] | X | | | X | | | X | | X | | |
| Conzon et al. [30] | X | X | | | X | X | X | X | X | | X |
| Mazzini et al. [31] | X | | | X | | | X | | | | |

Burada örnekleri verilen toplu taşıma IoT sistemlerinin, genel IoT sistemlerine kıyasla problemleri çok daha fazla sayıdadır. Farklılaşan problemler Bölüm 3'te tanıttığımız üzere yüksek mobilite, özel standartlar ve haberleşme hatları, SoC kullanımı, özel donanım kullanımı, özel protokol kullanımı ve işletim sistemi seçimidir. Bu farklı problemlerden dolayı genel kullanım amaçlı olarak tasarlanmış, IoT için MDE geliştirme araçlarının toplu taşımada kullanımları oldukça verimsiz ve eksik olacaktır. Toplu taşımaya özelleşmiş MDE araç ihtiyacını karşılamaya yönelik çalışmalara ihtiyaç duyulmaktadır. Bu çalışmada tartışılan toplu taşımaya özgü problemlerin çözümü için fikir aşamasında olan ve vaka çalışmaları ile sınırlı bazı arayışların başladığı [12, 18, 24 - 31] gözlenmiştir. Ancak yukarıda değinilen eksiklikleri giderme yönünde IoT tabanlı toplu taşıma sistemi uygulamalarının MDE olarak geliştirilmesi için yeni yöntemlerin ve bu yöntemleri uygulayan araçların geliştirilmesi gerekmektedir.

Tablo 1'deki genel IoT problemlerine ek olarak tanımlanan özel toplu taşıma problemlerine çözüm getiren ve IoT çözüm kavramlarını da birleştiren ve toplu taşıma alanına

özel bir MDE yöntemi ve bunu destekleyen araçların kullanımı, özellikle sistem modelleme ve kod üretim başarısını oldukça arttırabilecektir. MDE teknikleri kullanılarak örneğin IoT tabanlı araç içi ağı modellemesi ve üretilmesi kolaylık sağlayacaktır. Öte yandan toplu taşıma sistemleri yapısı gereği sürekli değişen ve/veya güncelenen çeşitli yazılım ve donanım bileşenlerini içermektedir ve yine Bölüm 3'te adreslenen yüksek mobilite, SoC kullanımı, vb. bu alana özel ihtiyaçlar da düşünülerek IoT tabanlı toplu taşıma sistemlerinin MDE geliştirilmesinde gidiş-dönüş (ing. round-trip) mühendislik tekniklerinin göz önüne alınması gerekmektedir. Çoğu zaman modellenmiş bir taşıt sisteminin uygulamasında değişikliğe gidilmektedir ve uygulamadaki değişikliğin bu sisteme ait modellere de yansıtılması ve bunlar arasında bir senkronizasyonun kurgulanması gerekmektedir. IoT tabanlı toplu taşıma sistemlerinin platform bağımsız bir düzeyden kod seviyesine kadar geliştirilmesini içeren bir MDE tekniği geliştirmenin yanında yeni yaklaşımların tersine mühendisliği de içermesi ve sistem uygulamalarındaki bir değişikliğin (örneğin taşıt içerisindeki bir ödeme yönteminin değiştirilmesi ya da yeni bir SoC mikroişlemcisinin kullanımının kesintisiz entegre edilmesi gibi) sistem modeline de yansıtılması uygulanan yöntemin verimini ve geniş çaplı kullanılabilirliğini arttıracaktır.

VI. SONUÇ

IoT sistemlerinin geliştirilmesinde her zaman gözlenen heterojenlik, farklı cihaz kullanımı ve büyük verinin getirdiği problemlerin yanı sıra toplu taşıma sistemlerinde kullanılan IoT yazılımlarının geliştirilmesi sırasında bu alana özel yüksek mobilite, özel standart ve araç içi haberleşme hattı kullanımı ve farklı SoC desteğinin sağlanması gibi çeşitli ihtiyaçların da göz önüne alınması gerekmektedir. Gerçekleştirilen bu çalışma ile ilk olarak bu alana özel problemler tanımlandıktan sonra IoT geliştirme için mevcutta bulunan MDE yaklaşımlarının bu problemleri gidermedeki eksiklikleri ortaya konmuştur. Henüz IoT tabanlı toplu taşıma sistemlerinin geliştirilmesini sağlayacak ve bu alanın ihtiyaçlarını karşılayabilecek bir MDE yaklaşımı bulunmamaktadır. Yeni MDE yöntemlerinin bu ihtiyaçları giderirken IoT tabanlı toplu taşıma sistemlerinin uygulamasında meydana gelebilecek değişiklikleri senkron bir yapıda sistem modellerine ve kodlarına da yansıtılabilecek şekilde; bir başka deyişle gidiş-dönüş mühendisliğini de içererek kurgulanması bu yöntemlerin kullanılabilirliğini güçlendirecektir.

KAYNAKLAR

- [1] A. Rayes, and S. Salam, Internet of Things From Hype to Reality: The Road to Digitization, Springer, 2019.
- [2] ITXPT Standarts, Information Technology for Public Transport Organization, <https://itxpt.org/>, son erişim: Ekim 2021.
- [3] S. K. Datta, R. P. F. D. Costa, J. Hürri, and C. Bonnet, "Integrating connected vehicles in Internet of Things ecosystems: Challenges and solutions," 17th IEEE International Symposium on A World of Wireless, Mobile and Multimedia Networks, 2016.
- [4] M. Bauer, N. Bui, P. Giacomini, N. Gruschka, S. Haller, E. Ho, R. Kernchen, M. Lischka, J. D. Loof, C. Magerkurth, S. Meissner, S. Meyer, A. Nettsträter, F. O. Lacalle, A. S. Segura, A. Serbanati, M. Strohbach, V. Toubiana, and J. W. Walewski, Internet-of-Things Architecture IoT-A Project Deliverable D1.2 – Initial Architectural Reference Model for IoT. Technical Report, 2011.
- [5] F. Ciccocozzi, I. Crnkovic, D. D. Ruscio, I. Malavolta, P. Pelliccione, and R. Spalazzese, "Model-Driven Engineering for MissionCritical IoT Systems," IEEE Software, 34-1, pp. 46–53, 2017.
- [6] F. Ciccocozzi, and R. Spalazzese, "MDE4IoT: Supporting the Internet of Things with Model-Driven Engineering," Intelligent Distributed

- Computing X - the 10th International Symposium on Intelligent Distributed Computing, 2016.
- [7] W. Wang, S. De, R. Tönjes, E. S. Reetz, and K. Moessner, "A Comprehensive Ontology for Knowledge Representation in the Internet of Things," 11th IEEE International Conference on Trust, Security and Privacy in Computing and Communications, 2012.
 - [8] C. M. Sosa-Reyna, E. Tello-Leal, and D. Lara-Alabazares, "Methodology for the model-driven development of service oriented IoT applications," *Journal of Systems Architecture* 90, pp. 15-22, 2018.
 - [9] M. B. Aydın, C. Öz, D. Ç. Tulazoğlu, ve G. Kardaş, "Toplu Ulaşım Araçları için ITxPT Standardı ile uyumlu bir Bilgi Sisteminin Geliştirilmesi," *Akıllı Ulaşım Sistemleri ve Uygulamaları Dergisi*, 2:2, pp. 1-13, 2019.
 - [10] H. Harrand, F. Fleurey, B. Morin, and K. E. Husa, "Thingml: a language and code generation framework for heterogeneous targets," *The ACM/IEEE 19th International Conference on Model Driven Engineering, Languages and Systems*, 2016.
 - [11] S. Pizonka, T. Kehrer, and M. Weidlich, "Domain Model-Based Data Stream Validation for Internet of Things Applications," 3rd International Workshop on Model-Driven Engineering for the Internet-of-Things, 2018.
 - [12] R. Neisse, I. N. Fovino, G. Baldini, V. Stavroulaki, P. Vlacheas, and R. Giaffreda, "A Model-based Security Toolkit for the Internet of Things," 9th International Conference on Availability, Reliability and Security, 2014.
 - [13] K. Thramboulidis, and F. Christoulakis, "UML4IoT: A UML-based approach to exploit IoT in cyber-physical manufacturing systems," *Computers in Industry*, 82, pp. 259–272, 2016.
 - [14] N. Muthukumar, S. Srinivasan, K. Ramkumar, D. Pal, J. Vain, and S. Ramaswamy, "A model-based approach for design and verification of Industrial Internet of Things," *Future Generation Computer Systems*, 95, pp. 354–363, 2019.
 - [15] A. Ahmed, M. Kleiner, and L. Roucoules, "Model-Based Interoperability IoT Hub for the Supervision of Smart Gas Distribution Networks," *IEEE Systems Journal*, 13 - 2. 2019.
 - [16] C. M. Farias, I. C. Brito, L. Pirmez, F. C. Delicato, P. F. Pires, T. C. Rodrigues, I. L. Santos, L. F. R. C. Carmo, and T. Batista, "COMFIT: A development environment for the Internet of Things," *Future Generation Computer Systems*, 75, pp. 128–144, 2017.
 - [17] B. Costa, P. F. Pires, and F. C. Delicato, "Towards the adoption of OMG standards in the development of SOA-based IoT systems," *The Journal of Systems & Software*, 169, 2020.
 - [18] W. Rafique, X. Zhao, S. Yu, I. Yaqoob, M. Imran, and W. Dou, "An Application Development Framework for Internet-of-Things Service Orchestration," *IEEE Internet of Things Journal*, 7:5, 2020.
 - [19] H. M. Marah, G. Kardaş, and M. Challenger, "Model-driven round-trip engineering for TinyOS-based WSN applications," *Journal of Computer Languages*, 65, 2021.
 - [20] S. Vorapojpisut, "Model-based Design of IoT/WSN Nodes: Device Driver Implementation," *International Conference on Embedded Systems and Intelligent Technology*, 2018.
 - [21] R. Xiao, Z. Wu, and D. Wang, "A Finite-State-Machine model driven service composition architecture for internet of things rapid prototyping," *Future Generation Comp. Sys.*, 99, pp. 473–488, 2019.
 - [22] J. Kölsch, S. Post, C. Zivkovic, A. Ratzke, and C. Grimm, "Model-based development of smart home scenarios for IoT simulation," 8th Workshop on Modeling and Simulation of Cyber-Physical Energy Systems, 2020.
 - [23] C. Kotronis, M. Nikolaidou, G. Dimitrakopoulos, D. Anagnostopoulos, A. Amira, and F. Bensaali, "A Model-based Approach for Managing Criticality Requirements in e-Health IoT Systems," 13th Annual Conference on System of Systems Engineering, 2018.
 - [24] M. Hause, J. Hummell, and F. Grelier, "MBSE Driven IoT for Smarter Cities," 13th Annual Conference on System of Systems Eng., 2018.
 - [25] L. Iovino, M. D. Sanctis, and M. T. Rossi, "Automated Code Generation for NFC-based Access Control," 4th International Workshop on Model-Driven Engineering for the Internet-of-Things, 2019.
 - [26] G. Cueva-Fernandez, J. P. Espada, V. García-Díaz, C. G. García, and N. Garcia-Fernandez, "Vitruius An expert system for vehicle sensor tracking and managing," *Journal of Network and Computer Applications* 42, pp. 178-188, 2014.
 - [27] J. Judvaitis, K. Nesenbergs, R. Balass, and M. Greitans, "Challenges of DevOps ready IoT Testbed," 4th International Workshop on Model-Driven Engineering for the Internet-of-Things, 2019.
 - [28] M. Henze, L. Hermerschmidt, D. Kerpen, R. Häußling, B. Rumpe, and K. Wehrle, "A comprehensive approach to privacy in the cloud-based Internet of Things," *Future Generation Computer Systems* 56, pp. 701–718, 2016.
 - [29] P. Patel, and D. Cassou, "Enabling high-level application development for the Internet of Things," *The Journal of Systems and Software* 103, pp. 62–84, 2015.
 - [30] D. Conzon, P. Brizzi, P. Kasinathan, C. Pastrone, F. Pramudianto, and P. Cultrona, "Industrial application development exploiting IoT vision and model driven programming," 18th International Conference on Intelligence in Next Generation Networks, 2015.
 - [31] S. Mazzini, J. Favaro, and L. Baracchi, "A Model-Based Approach Across the IoT Lifecycle for Scalable and Distributed Smart Applications," *IEEE 18th International Conference on Intelligent Transportation Systems*, 2015.