Yeşil Bilgi İşlem Bağlamında Kampus Ağlarında Enerji Tüketiminin İzlenmesi

Abdulkadir Özdemir[1], Aydın Koçak[1*]

1 Atatürk University, * Corresponding author, aydin.kocak@infoturk.com.tr

Özet

Bu çalışmada enerji tüketen cihazların veya tüm sistemin tüketim verileri IoT ürünleri kullanılarak bir veri tabanına kaydedilerek işlenebilir hale getirilmektedir. Böylece veriler üzerinde istenilen analiz yapılabilecebilecektir. Ayrıca bilişim altyapılarında kullanılan akım sensörleri, röle ve buna benzer sensörlerin verileri de Arduino, WeMoS gibi mikro denetçiler yardımıyla veri tabanlarına kaydedilebilmektedir. Sunuculara kaydedilen bu veriler yardımıyla bütün bir kampüsün enerji tüketimi kayıt altına alınabildiğinden tüketim eğilimleri görüntülenip tasarrufa yönelik çalışmalar yapılabilecektir. Böylece bütün tüketim verileri sayısallaştırılıp depolandıktan ve görüntülenebilir olduktan sonra günlük, haftalık düzenli enerji kullanım politikaları veya anlık verilere dayalı dinamik politikalar oluşturularak ağ cihazlarının enerji verimliliğini artırılabilecektir. Proje daha sonra uzaktan okuma yapılabilecek elektrik tüketim ölçüm saatine bile kolaylıkla çevrilebilir.

Anahtar kelimeler: IoT, Enerji tüketim izleme, Yeşil bilgi işlem, Sensör, Arduino.

Title

Monitoring of Energy Consumption in Campus Networks in the Context of Green Computing

Abstract

In this work, energy consuming devices or consumption data of the whole system are recorded in a database using IoT products and made processable. Thus, desired analysis on the data will be possible. In addition, the current sensors used in the information infrastructure, relays and similar sensors can be recorded in the databases with the help of micro controllers like Arduino, WeMoS. With the help of this data recorded on the servers, energy consumption of an entire campus can be recorded, so that consumption trends can be displayed and studies can be carried out. Thus, after all consumption data has been digitized, stored and viewed, energy efficiency of network devices can be improved by creating dynamic policies based on daily, weekly regular energy usage policies or instantaneous data. The project can then be easily converted to the electricity consumption measurement hour, which can be read remotely.

Keywords: IoT, Monitoring energy consumption, Green IT, Sensor, Ardunio.

Citation: Özdemir, A., Koçak, A. (2018, October) *Yeşil Bilgi İşlem Bağlamında Kampus Ağlarında Enerji Tüketiminin İzlenmesi*. IMISC 2018 Conference Proceedings, 8-11.

Editor: H. Kemal İlter, Ankara Yıldırım Beyazıt University, Turkey

Received: August 19, 2018, **Accepted:** October 18, 2018, **Published:** November 10, 2018

Copyright: © 2018 IMISC Özdemir, Koçak. This is an open-access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Giriş

Yeşil Bilgi İşlem son yıllarda popüler bir kavram olsa da aslında ilk adım 1992 yılında Energy Star kurumunun Amerikan Çevre Koruma Derneği tarafından kurulmasıyla başlıyor.[1] Günlük hayatta kullanılan bilgisayarların uyku moduna geçirilmesi gibi projeler hazırlayan Engery Star, zamanla doğayı koruyan ve tasarruf yapan teknolojilerin standartlarını da belirleyen bir kurum halini almıştır.

Enerji verimliliği konusunda ilk adımlarını çevre birimlerinin ve özelikle işlemcilerin daha az enerji tüketmesi ve bununla birlikte soğutma için kullanılan enerjiden tasarruf etme düşüncesiyle adım atan sektör son yıllarda ciddi ilerleme kat etmiştir. Üst seviye teknoloji diyebileceğimiz işlemci dünyasında çok çekirdekli işlemciler konvansiyonel işlemcilere oranla ciddi enerji tasarrufu sağlayacak şekilde tasarlanırken, çevre birimleri dünyasında fotokopi makineleri ve yazıcılar birkaç sene öncesine kadar harcadıkları enerjiyi yarı yarıya düşüren ürünlerle karşımıza çıkmıştır. Bir milyardan fazla kişi PC'ye sahip, bunların çoğu ise daha fazla enerji sarf eden eski bilgisayarlardan oluşmaktadır. Bugün dünya nüfusunun neredeyse yarısı cep telefonu kullanıyor. [1] Veri merkezlerinin verimli çalışabilmesi için 26°C yeterli olmasına rağmen genellikle 18°C'ye kadar soğutuluyor. [1] Cep telefonu enerjilerinin yaklaşık üçte ikisi boşa harcanıyor, Dizüstü

bilgisayarların batarya dolum işlemi tamamlanmasına rağmen şarj adaptörleri saatlerce prizde takılı bırakılarak hem bataryanın kullanım süresinin kısalmasına hem de elektrik israfına neden oluyor.

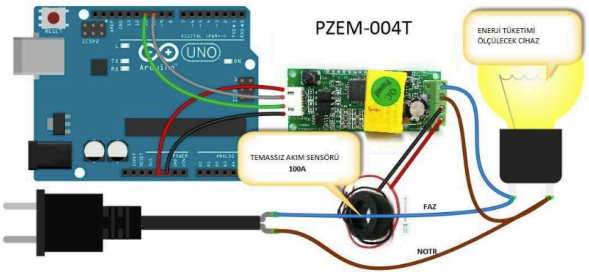
Yeşil Bilgi İşlemin en önemli destekleyici teknolojilerinden birisi de sanallaştırma çözümleridir. Sunucu / Ağ / Depolama çözümleri gün geçtikçe yaygınlaşmaktadır. Fiziksel sunucuların aynı ölçekteki sanal sunucularından sadece %12 daha performanslı olması bu alanın gelişmesine ve yaygınlaşmasına olanak tanımaktadır. Aslında %12'lik bir performans kaybı hala bilişim dünyası için çok az bir fark değildir ancak geçmiş 10 yılda bu oranın %35 civarında olduğunu düşünürsek gelişmeler ileriki bir zamanda benzer performans değerlerine ulaşılabileceğini göstermektedir.[1]

Bulut Bilişim ile sanallaştırma bir adım daha ileri götürülerek kaynak kullanım ve enerji tüketimi en alt seviyelere getirilmeye çalışılmaktadır. Bununla birlikte web tabanlı uygulamalar geliştirilip kullanıldığında kullanıcı tarafında da çok performans gerektirmeyen ince istemciler kullanılabilecek ve bu da kullanıcı tarafında yeşil bilgi işlem tezini destekleyecektir.

Bu çalışmada ilk önce bütün enerji tüketiminin görünür olmasını sağlayacak bir IoT (Internet of Things – Nesnelerin İnterneti) bazlı sistem geliştirilmiştir. Yapılan çalışma ile sadece bilişim cihazları değil enerji tüketen bütün cihazların tüketim verileri depolanabilir ve incelenebilir hale getirilebilecektir.

Prototip olarak oluşturduğumuz enerji verisi toplama seti şu parçalardan oluşmaktadır:

- Mikro Denetleyici: Arduino[2] temelli ve 802.11n kablosuz haberleşme özelliği içeren WeMoS D1 R2[3],
- Akım ölçer: Temassız akım ölçümü yapabilen ve seri port üzerinden bu verilerin alınabildiği PZEM – 004T[4]
- Veri Depolama: Bütün sistemin ürettiği veriyi depolayıp görüntülemek için Ubuntu[5] Linux tabanlı bir sunucu bilgisayar sistemi kullanılmıştır.



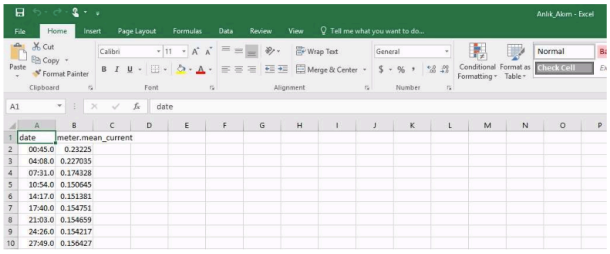
Şekil 1. IoT ölçüm sistemi.

Şekil 1'de prototipin basit şeması görülmektedir. Bu sistemden elde edilen enerji tüketim verileri sonraki bölümde ayrıntıları verilecek olan Ubuntu Linux sistemi üzerine aktarılmaktadır.

Şekil 2'de Linux tabanlı sunucuya gönderilen enerji tüketim verinin CSV formatında bir örneği görülmektedir. Bu ölçüm verilerinin nasıl elde edildiği ve nasıl merkezi sunucuya gönderildiğinin ayrıntıları sonraki bölümde detaylandırılmıştır.

Sistem Tasarımı

IoT temelli ve özgün olarak tasarlanan sistem; kablosuz ağ destekli mikro denetleyici ve temassız akım ölçümü ve bu verilerin gönderilip depolandığı, görselleştirildiği, üzerinde alarmlar üretilebildiği Linux tabanlı sistemden oluşmaktadır.



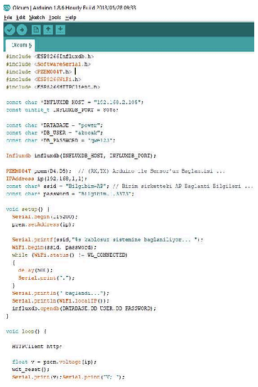
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1	date	meter.mean_current														
2	00:45.0	0.23225														
3	04:08.0	0.227035														
4	07:31.0	0.174328														
5	10:54.0	0.150645														
6	14:17.0	0.151381														
7	17:40.0	0.154751														
8	21:03.0	0.154659														
9	24:26.0	0.154217														
10	27:49.0	0.156427														

Şekil 2. Örnek ölçüm verisi.

Şekil 1'de ölçüm sistemi devre şeması verilmiştir. Bileşenleri kısaca açıklanacak olursa;

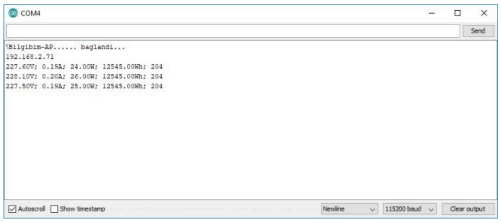
- Mikro Denetleyici: ESP8266[6] kablosuz erişim modülü ile 802.11n destekli ağlara kablosuz olarak bağlanabilen WeMoS D1 R2[3] Arduino[2] geliştirme ortamı destekleyen bir mikro kontrollerdır.
- Ölçüm Sensörü: Akım/Voltaj/Anlık ve Toplam Güç tüketimini hesaplayabilen Peacefair PZEM – 004T[4] sensörüdür.

Ölçüm verilerini Linux tabanlı sunucuya gönderen sistem donanım olarak bu iki bileşenden oluşmaktadır. Bu sistemin ölçüm yapabilmesi ve aldığı ölçüm bilgilerini kablosuz olarak Linux sisteme gönderebilmesi için Arduino[2] geliştirme ortamında programlandı. Program kodunun örneği aşağıda verilmiştir.



Şekil 3. IoT bağlantı ve ölçüm kodları C++ (1)

Kodlardan da anlaşılabileceği gibi önce mikro denetleyici kablosuz ağlara bağlanabilmesi ve ölçüm yapabilmesi başta olmak üzere birçok kütüphane kullanılmıştır. Bu kısımda mikro denetleyici ve akım ölçüm sensörünün bağlantısı için kullanılacak sayısal girişler belirlenmektedir. Ayrıca en önemli kısmı alınan ölçüm bilgilerinin hem seri ekrana hem de Linux sistem üzerinde kurulu olan "Time Series Database"e gönderilmesi ile ilgili kodlar gözükmektedir. Arduino[2] programlama sisteminin seri ekranının görüntüsü ise aşağıdaki şekildedir.



Şekil 4. IoT seri port bağlantı ve tüketim verileri.

Koddaki en önemli kısım bu tüketim verilerinin kurulan Time Series Database'e (InfluxDB)[7] payload olarak web servis üzerinden aktarılmasıdır. Eğer aktarmada sorun çıksaydı bununla ilgili hata mesajı da seri port ekranında görebilecektir. Mikro denetleyici yukarıdaki şekilde programlandıktan sonra değişiklik gerekmediği sürece 9 Voltluk DC bir besleme ile kararlı bir şekilde kodu çalıştıracak ve ölçüm bilgilerini istenilen yere gönderecektir.
Sistemimizin analiz işlemleri için kullanılan bileşeni ise Ubuntu 18.04 LTS [5] sunucusu üzerinde kurulu olan ve verilerin depolandığı Time Series Database (InfluxDB) dir. Bu verilerin görselleştirildiği Chronograf[8] yazılımı ve oluşturulan veriler üzerinde analizler ve alarmlar oluşturmayı sağlayan Kapacitor[9] yazılımları da sisteme entegre edilmiştir.

Time Series Database(InfluxDB)[7]: Zamana bağlı değişen veriler için saklama alanı geleneksel veri tabanları yerine zaman bazlı veri tabanları olarak önerilmektedir. Zaman bazlı veri tabanları özellikle zamana göre değişen kullanım verileri oluşturan CPU, RAM, vs. için kullanılmaktadır. Bu uygulamada InfluxDB[7]'nin seçilmesinin sebebi açık kaynak kodlu, kararlı ve geliştirilmiş bir sürüme sahip olmasıdır. Bu yazılımın güncel versiyonu ise aşağıdaki komutlar ile kurulmuştur.

```
wget https://dl.influxdata.com/influxdb/releases/
influxdb_1.6.0_amd64.deb sudo dpkg -i
influxdb_1.6.0_amd64.deb
```

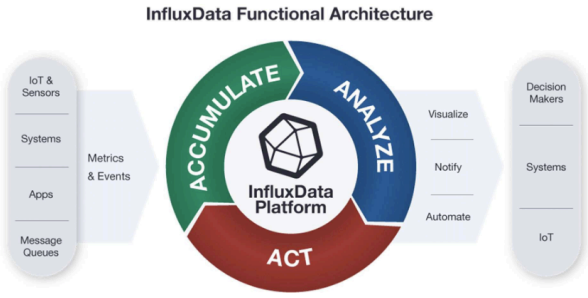
Veri Görselleştirme Platformu (Chronograf)[8] : InfluxDB veya diğer time series database verilerinin istenilen şekilde görselleştirilmesini sağlayan platformdur. Ayrıca bu programın web tabanlı yönetim ekranından InfluxDB ve veri analizi yapabilen Kapacitor[9] programının da yönetim ve ayarları yapılabilmektedir. Yazılımın kurulumu aşağıdaki komutlar ile yapılabilmektedir.

```
wget https://dl.influxdata.com/chronograf/releases/
chronograf_1.5.0.1_amd64.deb sudo dpkg -i
chronograf_1.5.0.1_amd64.deb
```

Veri Analizi ve Alarm Platformu(Kapacitor)[9] : Time series database verileri üzerinde analizler yapılmasını veya alarmlar oluşturularak tetiklenmesini sağlayan platformdur. Chronograf üzerinden bağlantı ayarları ve alarmlar oluşturulabilmektedir. Yazılımın kurulumu aşağıdaki komutlar ile yapılabilmektedir.

```
wget https://dl.influxdata.com/kapacitor/releases/
kapacitor_1.5.0_amd64.deb sudo dpkg -i
kapacitor_1.5.0_amd64.deb
```

Bu üç yazılımda influxdata yazılım şirketine (www.influxdata.com) ait açık kaynak kodlu yazılımlardır. Bu nedenle de prototipte InfluxData yazılım şirketinin önerdiği aşağıdaki mimari[10] biraz daha özelleştirilerek oluşturulmuştur.



Şekil 5. Sistem mimarisi.

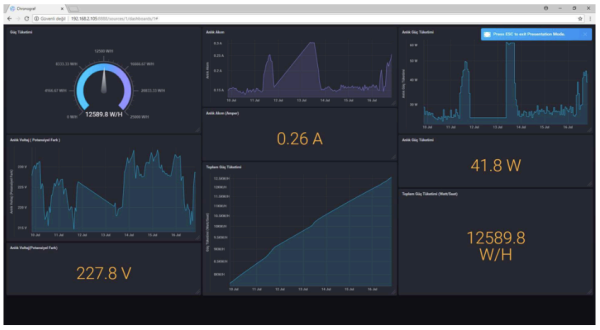
Şekil 5'deki mimaride de gözükebileceği gibi IoT sisteminden alınan veriler time series database (InfluxDB) gönderilmektedir. Sonra bu verileri Cronograf yazılımıyla görselleştirilmektedir. Kapacitor bileşeni ile de analizler oluşturup alarmlar üretebiliyor. Bu mimarinin son kısmında ise karar oluşturma kısmı yer almaktadır. Elde edilen verilere göre enerji tasarrufu sağlayacak kararlar belirlenip uygulanabilir. Bütün enerji verisi sistem üzerinde bulunduğu için elektrik voltaj belirli bir süre boyunca aşırı şekilde artarsa sistemlerinin zarar görmemesi için belirli bir süre için enerji kesilmesi kararı verilebilir veya sadece uyarı e-postası'i de atılabilir. Bu tür karar örnekler artırılabilir. Elektrik voltajı 225 Volt'u geçerse ilgili kişiye e-post atılmasını sağlayan bir Kapacitor kodu aşağıda görülmektedir.

Örnek Kapasitör Alarm Kodu (TICK Script):

```
var db = 'power'
var rp = 'autogen'
var measurement = 'meter'
var groupBy = []
var whereFilter = lambda: TRUE
var name = 'Voltaj_Uyari'
var idVar = name
var message = '{{.ID}} {{.Name}} {{.TaskName}}
{{.Group}} {{.Tags}} {{.Level}} {{ index .Fields "value"
}} {{.Time}}' var idTag = 'alertID'
var levelTag ='level'
var messageField = 'message'
var durationField = 'duration'
var outputDB = 'chronograf'
var outputRP = 'autogen'
var outputMeasurement = 'alerts'
var triggerType ='threshold'
var details = 'Voltaj da aşırı yükselme mevcut.'
var crit = 225
var data = stream
    |from()
    .database(db)
    .retentionPolicy(rp)
    .measurement(measurement)
    .groupBy(groupBy)
    .where(whereFilter)
    |eval(lambda: "voltage")
    .as('value')
var trigger = data
    |alert()
    .crit(lambda: "value" > crit)
    .message(message)
    .id(idVar)
    .idTag(idTag)
    .levelTag(levelTag)
    .messageField(messageField)
    .durationField(durationField)
    .details(details)
    .email()
    .to('aydin.kocak@bilgibim.com.tr')
trigger
    |eval(lambda: float("value"))
    .as('value')
    .keep()
    |influxDBOut()
    .create()
    .database(outputDB)
    .retentionPolicy(outputRP)
    .measurement(outputMeasurement)
    .tag('alertName', name)
    .tag('triggerType', triggerType)
trigger
    |httpOut('output')
```

Program işlemlerinin tamamı ilgili sunucunun TCP - 8888 portuna herhangi bir web tarayıcı ile bağlanıp gerekli kullanıcı adı ve şifreleri girerek yapıla bilinmektedir. Bu nedenle de toplanan tüm verilerin görselleştirilmiş haline ulaşılabileceği gibi yukarıdaki örnek kod gibi istenildiği kadar alarm ve analiz işlemi de yapılabilir.

Bu işlemlerin yanında karar oluşturmada etkili olan akım ve gerilimin, anlık ve toplam enerji tüketim verilerinin görselleştirilmiş ekran görüntüleri aşağıdaki gibi izlenebilmektedir.



Şekil 6. Görselleştirilmiş tüketim verileri.

Şekil 6'dan da görülebileceği gibi karar üretmede etkili olan anlık ve toplam enerji tüketimleri, voltaj ve akım bilgilerini zamana bağlı olarak depolanıp görsel olarak takip edilebilmektedir.

Sonuç ve İleriki Çalışmalar

Yapılan çalışma çok büyük enerji tasarrufu sağlayacak denetim sistemlerinin, 15 $ gibi çok küçük maliyetlerle yapılabileceği gösterilmiştir. Bilişim sistemlerinin kalbi olan sistem odalarının veya tüm binanın anlık ve zaman içerisindeki bütün enerji tüketim ve ilgili elektriksel parametreleri kayıt altına alınıp anlamlı bir şekilde görselleştirilebilmektedir. Bu çalışma ile daha önce yapılan kablolu ve kablosuz ağ ürünlerinin yeşil bilgi işlem kriterine uygun olarak çalıştırılmaları [11] ve Kampüs ağlarında PoE anahtarların güç tüketiminin izlenmesi [12] çalışmalarının geliştirilmesidir. Daha önce yapılan iki çalışmada da cihazların akıllı cihaz olmaları zorunluluğu vardı. Bu nedenle veri toplanması [12] için sadece veri hattı üzerinden elektrik iletimi sağlayabilen (PoE) anahtarlar olması şartı vardı. Bu çalışmada enerji tüketimi olan herhangi bir cihazın bütün tüketim verileri hassas bir şekilde dijital olarak en uygun platform olan time series database'e aktarılabilmektedir. Aktarılan veriler üzerinde yapılacak analizler ve alarmlar sınırsız sayıda tanımlanabilmektedir.

Sonraki çalışmada statik ve dinamik tüketim politikaları kullanılarak ölçümler yapılacak ve elde edilen verim bilgileri anlamlı hale getirilmeye çalışılacaktır. Ayrıca çalışmanın kapsamı dışında olmakla birlikte aslında yapılan bu çalışma ile elektrik dağıtım şirketleri bütün tüketim verilerine uzaktan erişip sayaç okuma işlemini de uzaktan yapabilir hale gelebilir. Yapılan çalışmanın nihai amacı geleceğin en önemli ihtiyacı olan elektrik enerjisi tüketimini azaltmaya yönelik kararlar almayı sağlayan sistemler tasarlayarak hem çevreye hem de ekonomiye katkı sağlamaktır.

Kaynakça

[1] Green IT Nedir?, https://www.bilgiustam.com/yesil-bilgi-teknolojileri-green-it-nedir/
[2] Arduino Projesi, http://www.arduino.cc
[3] WeMoS Projesi, https://www.wemos.cc/
[4] PZEM – 004T akım sensörü,ve kütüphanesi, https://github.com/olehs/PZEM004T
[5] Ubuntu Linux Dağıtımı, http://www.ubuntu.com
[6] ESP8266 Seri Kablosuz İletişim Modülü, https://www.espressif.com/en/products/hardware/esp8266ex/overview
[7] InfluxDB Time Series Database, https://www.influxdata.com/time-series-platform/influxdb/
[8] Chronograf Veri Görselleştirme Platformu, https://www.influxdata.com/time-series- platform/chronograf/
[9] Kapacitor Veri Analiz ve Alarm Platformu, https://www.influxdata.com/time-series- platform/kapacitor/
[10] Influx Data Mimarisi, https://www.influxdata.com/products/
[11] A. Özdemir, A.Koçak, Kablolu ve Kablosuz Yerel Alan Ağlarında Yeşil Bilgi İşlem Teknolojilerinin Kullanımı, 2.Ulusal YBS Kongresi Erzurum, 2015.
[12] A. Özdemir, A.Koçak, Kampüs ağlarında PoE Ağ Anahtarlarının Güç Tüketiminin İzlenmesi, 3.Uluslararası YBS Kongresi İzmir, 2016.