

Bolum 2

URL'ler ve Kaynaklar

Interneti bir dev gibi,genisleyen sehirler gibi,gorulecek ve yapılacak bir suru mekanin oldugu bir yer gibi dusunun.Siz ve diger sehir sakinleri ve turistler yukselen cevrenizde sehirin muazzam servislerini ve etkilesimli yerlerini adlandiriyorsunuz.Muzelere,restorant ve insanların evleri icin sokak adresleri.Itfaiye,patronun sekreteri ve anneniz yada yeterince bizi aramiyorsun diyen biri icin telefon numaralarini kullaniriz.

Her sey isim ile standartlastirilir.Sehirin kaynaklarini isimler yardimiyla siralayabiliriz.Kitaplar ISBN numarasina sahip,otobusler rota sayilari ve durak sayilarina sahip,banka hesaplari hesap numarasina sahip ve insanlar sosyal guvenlik numarasina sahip.Yarin is arkadasinizla havaalanindaki 31. Kapida bulusacaksiniz.Her sabah trene binip Kendall Square'de iniyorsunuz gibi.

Ve herkes farkli isimlerin standartlari konusunda ayni goruste birlesirler.Biz kolayca sehrin iyi yanlarini baskasiyla paylasabiliriz.Ornegin kardesim beni anayoldaki batti ciktida tukur dediginizde size anlar.

URL'ler(Uniform resource locators)Internet kaynaklarinin isimlerinin standartlastirilmasinda bizi yardimci olur.URL elektronik bilgilerin bir parcasidir.Size bir seyin nerde oldugunu ve onla nasil etkilesime gecicegimizi soylar.

Bu bolumde su bolumler uzerinde durucuz:

URL syntax ve URL cesitli bilezenlerinin anlami nedir ve ne yaparlar.

URL kisayollarini cogu web istemcisi destekler.

URL encode etme ve karakter kurallari.

Mevcut URL semalari Internet bilgi system varyasyonlarinin cogunu destekler.

URL'nin gelecegi (URN'yide kapsayacak sekilde)

Internet Kaynaklarinda Gezinmek

URL'ler tarayicinizin bilgiyi bulmasi icin gerekli olan kaynak konumlaridir.Insanlarin uygulamalari bulmasini,kullanmasini ve milyarlarca data kaynagini internette paylasmayi kolaylastirir.URL'ler insanlar tarafından artik siradanlasmis bir seydir ve erisim icin HTTP'de ve diger protokollerde kullanilir.Kisi tarayiciya URL'yi girer ve,sahnenin arka tarafinda,tarayici uygun protocol mesajini yollayarak kullanicin istedigii kaynagi getirir.

URL'ler aslında kaynak belirleyicilerin alt kumesidir. Kaynak belirleyici dedigimiz sey URI(Uniform resource identifier)daha onceki bolumlere bahsetmistik. URI genel konsept olarak iki alana ayriliyordu. URL ve URN. URL'ler kaynagin nerde oldugunu tarif ederek kaynagini belirler. URN'ler ise nerde bulunduagina bagli olmaksizin isimden kaynagi belirler. Merak etmeyin URN'ye ilerki bolumde deginecegiz ama unutmayin sadece kitap cercevesinde her seyin ogretilmesi icin verecegiz normalde bizim isimiz kesinlikle URL ile.

HTTP'in belirli kisimlari cogunlukla genel konsept olarak URI'yi kullanir. Pratikte tabiki :D

Ancak HTTP uygulamalari URL ile kankadir sadece onun dilinden anlar. URN, URI, URL espri yapsa URL'nin esprisini anlar diyebiliriz. Bu kitap boyunca URI diyebiliriz URL diyebiliriz ama bilinki URL'den bahsediyoruz.

<http://www.joes-hardware.com/seasonal/index-fall.html> URL sini bir cekelim serverdan neler oluyor bakalim.

URL'nin ilk kısmi olan (http)kısmi URL semasıdır. Sema kaynaga nasıl ulaşılacağını söyler. Burda http protokolu ile ulaşıyoruz.

URL'nin ikinci kısmi (www.joes-hardware.com) server konumudur. İstemciye kaynagin nerede bulunduğunu söyler.

URL'nin üçüncü kısmi (/seasonal/index-fall.html) kaynak yoludur. Bu kaynak yolu bize serverdaki hangi belirli kaynakların istendiği söyler.

Resimde görüldüğü gibi.

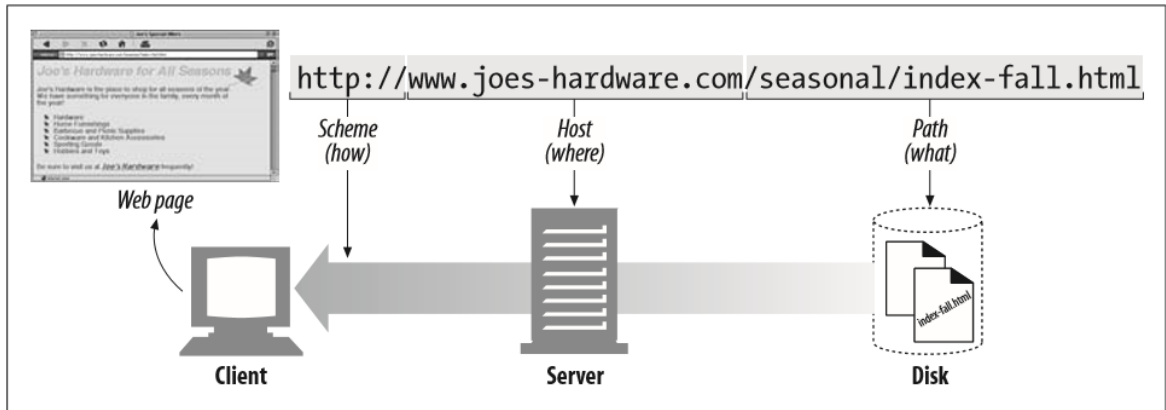


Figure 2-1. How URLs relate to browser, machine, server, and location on the server's filesystem

URL'ler protokoller aracıyla size müsait kaynaklara yoneltir. İnternetteki herhangi bir kaynagi size isaret edebilirler, misal bir kisiden email hesabi:

<mailto:bengenelkurmajcumhurbaskanibasbakaninizim@tbmm5001.gov>

müsait olan dosyalarda olabilir. Dosya Transfer Protokolu kullanılarak uygulanır (FTP)

<ftp://ftp.lots-o-books.com/pub/complete-price-list.xls>

Filmlere ev sahipliği yapan stream video serverlarında olabilir.

rtsp://www.joes-hardware.com:554/interview/cto_video

URL'ler bize bir sürü yol sağlar.Cogu URL'ler aynı "sema://server adresi/yol" sahiptir.Kaynaklar tasınırken her bir kaynagi isimlendirmek için sadece bir yolunuz var demektir çünkü herkes bu ismi kullanarak kaynaga ulaşacaktır.Ancak,her zaman aynı durum söz konusu değildir.

URL'lerden Önceki Kara Günler

Webden ve URL'lerden önce Uygulamaların etiketleme cesitine dayanarak Nette datalara erismeye çalışıyorlardı.Cogu insan doğru uygulamayı bulacak kadar şanslı değildi veya kullanmak için sabırlı ve basit kapabilen insanlar değildi.

URL'ler gelmeden önce,eger *complete-catalog.xls* dosyasının arkadaşınızla paylaşmak isterseniz,soyle bir şey demeniz lazım:"FTP bağlantısı ftp.joes-hardware.com bağlanmak için kullanalım.Anonim olarak login olalım.Kullanıcı ismi ve password girelim.Dizini degistirelim.Binary mode'a gecerim.Simdi dosyayı indirelim ve görüntüleyelim.

Bugün,Mozilla gibi,Internet Explorer gibi tarayıcılar çokca fonksiyonu bir basit paketle edindiler.URL'leri kullanarak,bu uygulamalar cogu kaynagin uniformuna erisebilme yetisi kazandılar.Sadece bir arayuzle hemde.Yani karmakarışık talimatlar yerine yukarıda gördüğünüz gibi,browsera girisün kardesim tik yazisün gelio. ftp://ftp.lots-o-books.co/pub/complete-catalog.xls.

URL'ler uygulamaların kaynaga nasıl erisileceğinin farkında olmasını sağlar.Hatta,birçok kullanıcı tarayıcı kaynaklara erisirken hangi protokolü kullanıyor,hangi method kullanıyor,ne istiyor habersizdir.

Browserlarla beraber,İnternette haberleri okumak için artık baskasının bir FTP serverına bağlanmanıza gerek yok.Email mesajlarını alıp veya yollamak için elektronik mail programına ihtiyacınız yok.URL'ler internet dünyasına basitlik anlamında,tarayıcılara kaynaklara erisimi ve yönetimi konusunda akıllılastırarak internet dünyasına baya bir şey kattı.Uygulamalar URL'leri bilgilere basitce erismek için kullanabilir.

URL'ler size ve tarayiciniza gerekli olan bir parça bilgiyi sağlar.Hangi kaynagin nerde olduğunu,kaynaga nasıl ulaşılacağını ve istediğiniz özel kaynagi tanımlayarak bize çok yardımcı olur.

URL SYNTAX'I

URL'ler İnternette kaynagin nerede olduğunu acikca belirtir,ama tabiki farklı semalarla erisilebilir bu kaynaklara(HTTP,FTP,SMTP)ve URL syntaxları semadan semaya degisir.

Bu demek değildir ki her URL semasi farklı bir syntaxe sahiptir,cogu URL genel URL syntaxine sahiptir,baska URL'lerde acikca büyük bir degisiklikte gozlenebilir tabiki.

Cogu URL syntax 9 genel formattan olusur.

<sema>://<kullanici>:<sifre>@<host>:<port>/<yol>;<parametre>?<sorgu>#<frag>

Nerdeyse hicbir URL tum bilesenleri kapsamaz.Uc tane onemli bolumumuz var sema,host ve yol.Asagidaki tabloda incellersiniz cesitli bilesenler kisaca anlatilmis.

Table 2-1. General URL components

Component	Description	Default value
scheme	Which protocol to use when accessing a server to get a resource.	None
user	The username some schemes require to access a resource.	anonymous
password	The password that may be included after the username, separated by a colon (:).	<Email address>
host	The hostname or dotted IP address of the server hosting the resource.	None
port	The port number on which the server hosting the resource is listening. Many schemes have default port numbers (the default port number for HTTP is 80).	Scheme-specific
path	The local name for the resource on the server, separated from the previous URL components by a slash (/). The syntax of the path component is server- and scheme-specific. (We will see later in this chapter that a URL's path can be divided into segments, and each segment can have its own components specific to that segment.)	None
params	Used by some schemes to specify input parameters. Params are name/value pairs. A URL can contain multiple params fields, separated from themselves and the rest of the path by semicolons (;).	None
query	Used by some schemes to pass parameters to active applications (such as databases, bulletin boards, search engines, and other Internet gateways). There is no common format for the contents of the query component. It is separated from the rest of the URL by the "?" character.	None
frag	A name for a piece or part of the resource. The frag field is not passed to the server when referencing the object; it is used internally by the client. It is separated from the rest of the URL by the "#" character.	None

Ornegin,dusunun http://www.joes-hardware.com:80/index.html " semamiz "http" hostumuz "www.joes-hardware.com" portumuz "80" ve yolumuz "/index.html".

Semalar:Hangi Protokol Kullaniliyor

Sema gercekten verilen kaynaga ulasilmasi konusunda gercek bir tanimlayici;uygulumaya konusmak icin hangi protokole ihtiyac oldugunu soylar.Basitce HTTP URL'si "http"dir.

Sema bileseni alfabetik karakterle baslamali,ve geri kalan URL'den ":" karakteri ile ayrilmalidir.Semalar yok harf kucukmus buyukmus varmis yokmus dinlemez zbam zbam diye giriler.Yani http://www.joes-hardware.com ve HTTP://www.joes-hardware.com bizim icin ayni.

HOSTLAR VE PORTLAR

Internette bir kaynagi bulmak icin,uygulamanin bilmesi gereken hangi makina kaynagi tasiyor ve arzu edilen kaynagin makinada nerde oldugu.Bu iki seyi iste host ve portlar sagliyor.

Host bileşeni Internetteki kaynağa sahip olan host makinasını belirler. Hostname olarakta çıkabilir karsımıza (www.joes-hardware.com) veya bir IP adresi olarakta. Örneğin, aşağıdaki iki URL'de aynı kaynaktır. İlki hostname ile ikincisi IP ile gösterilir.

Http://www.joes-hardware.com:80/index.html

http://161.58.228.45:80/index.html

Port bileşeni hangi serverin dinleneceğini belirler. HTTP için, TCP protokolünün parçası olan HTTP için, default port 80'dir.

KULLANICI VE PASSWORDLAR

Çok daha ilginç bileşenler kullanıcı ve passwordlardır. Çoğu server dataya ulaşmadan önce kullanıcı adı ve password ister. FTP serverları buna güzel örnektir. Aşağıda örnek olacak.

ftp://ftp.prep.ai.mit.edu/pub/gnu

ftp://anonymous@ftp.prep.ai.mit.edu/pub/gnu

ftp://anonymous:my_password@ftp.prep.ai.mit.edu/pub/gnu

http://joe:joespassword@www.joes-hardware.com/sales_info.txt

İlk örnek kullanıcı adı ve password yok, standart semamız, host ve yol var. Eğer uygulama kullanıcı adı ve password gerektiren bir URL kullanıyorsa mesela FTP gibi, eğer eklenmediyse default kullanıcı adı ve password denencektir. Mesela, eğer kullanıcı adı ve password belirtmeden FTP li bir URL ye browserinizi yönettirirsiniz, kullanıcı adını "anonymous" ve default password yollayacaktır (Internet Explorer için "IEUser", Mozilla için "Mozilla").

İkinci örnek kullanıcı adını "anonymous" olarak belirlemiş. Bu kullanıcı adı host bileşeniyle kombin edilmiş. Email adresi gibi. @ karakteri kullanıcı ve password bileşenlerini URL'den ayırır en yukarıda resimde gösteriliyor zaten.

Üçüncü örnekte hem kullanıcı adı hemde password girilmiş ve ":" karakteri ile ayrılmıştır. Dördüncü örnekte benziyor zaten.

YOLLAR

Yol bileşeni Kaynağın makinada nerde yaşadığını gösterir. Yol aynı bizim bilgisayarımızdaki dosya sistemi yolu gibi, Örneğin:

Http://www.joes-hardware.com:80/seasonal/index-fall.html

Yol burda "/seasonal/index-fall.html" dir, UNIX dosya sistemini biliyorsanız aynısı. Yol serverin kaynağı nerde tuttuğunun bilgisidir. HTTP URL'lerinde yol "/" ile ayrılır (Gene UNIX dosya sistemi). Her yol segmenti kendine ait parametre bileşenine sahip. Yukarıdaki resimde iyice inceleyebilirsiniz zaten her şey rahatlıkla anlaşılır.

PARAMETRELER

Bir çok sema için, host ve yol bir obje için yeterli değil. Hangi port server tarafından dinleniyor hatta kaynaklara erişim için kullanıcı adı ve password bunların hepsi bir kenara yeterli değildir, Cogu protocol daha fazla bilgiye ihtiyaç duyar.

Uygulamalar kaynaklara erişmek için URL'deki parametreleri yorumlaması lazım. Aksi takdirde, Server obur taraftaki servise istek yapamaz, belki daha kötü, servis yanlış olabilir. Örneğin, FTP protokolünden gidelim gene, İki transfer modu var, binary ve text dosyası. Binary modundaki resmin text modunda transferini istemeyiniz. CD çizilmesi gibi binary formatındaki resim bitleri çizilebilir.

Uygulamaya parametreleri vermek serverlar düzgünce konuşmasını sağlar. URL'ler parametre bileşenine sahiptir bu yüzden. Bu bileşen URL'deki isim/değer çiftidir, ve geri kalan URL'den “;” ile ayrılır. Kaynağa ulaşılması için uygulamaya fazladan bilgi sağlanır. Örneğin:

`ftp://prep.ai.mit.edu/pub/gnu?type=d`

Bu örnekte bir tane parametre var, type=d, yukarıda dediğim gibi isim değer çiftidir. İsim “type” ve değer “d”.

Önceden bahsettiğimiz gibi HTTP URL'leri yol bileşeniyle ayrılıp kendi parametresine sahip olabilir.

`http://www.joes-hardware.com/hammers;sale=false/index.html;graphics=true`

Bu örnekte iki tane yol segmentimiz var. hammers ve index.html. hammers segmenti “sale” parametresine sahip ve değer false. Index.html ise “graphics” parametresine sahip değeri true.

Sorgu Katarları

Bazı kaynaklar, veritabanı servisleri gibi, soru sorabilir ya da istekte kaynakları daraltarak sorgu yapabilir. Diyelimki Joe'nin Hardware dükkanında satılmamış envanterler listeli bir şekilde duruyor ve envanter sorgulanmasına izin veriyor. Çünkü hangi ürünün stokta olduğunu bilmesi lazım. item numarası 12731 diyelim. URL sorgusu şöyle olur.

`http://www.joes-hardware.com/inventory-check.cgi?item=12731`

Cogu bölüm için diğer URL'ler benzer yeni olan ise ? işareti. Buna sorgu bileşeni diyoruz. Bunu görünce sorgu gelio bilin yani. Zaten bu kitabı bitirince tarayicinizi açtığınız andan itibaren ne nasıl nerede nolu her şey kafanızdan tık tık geçecek anlıcağınız. Sorgu bileşeniyle URL gateway kaynağına geçer, Yol bileşeniyle yani “/” ile gateway kaynağını tanımlarız gençler sakın olun. Gateway Bölüm 8 de tartışılır.

Aşağıdaki resimde örnek bir sorgu bileşeniyle gateway arasındaki geçişi gösteriyor. Sorgu itemin yani objenin var olup olmadığının denetlenmesidir, 12731 ile görüyoruz item vardır. Envanter boyutu large rengi blue dur.

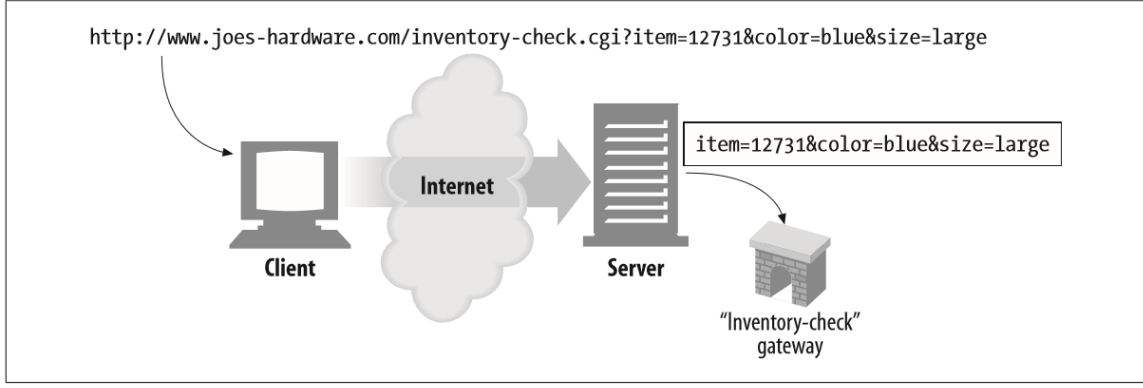


Figure 2-2. The URL query component is sent along to the gateway application

Sorgu bileşeni için hiç bir formata gerek yoktur. Ancak güvenlik için illegal karakterlerin engellenmesi gerek. Her iki bölümlerde bakıcaz bunada. Birçok kurul tarafından çoğu gatewayde “name=value” bölümü & karakteri ile ayrılmıştır.

<http://www.joes-hardware.com/inventory-check.cgi?item=12731&color=blue>

Bu örnekte iki isim/değer çifti mevcut.

FRAGMENTLER

Bazı kaynak tipleri HTML gibi olanlar mesela kaynak leveli olarak bölünebilir. Mesela, tek ve geniş text dokümantasyonlarının bölümlere ayrıldığını düşünün ayrı ayrı ve URL tüm text dokümantasyonunu gösterirken siz kaynağın bölümünü seçebilirsiniz.

Kaynağı fragment bölümlerini ayrılmasına izin verilmesi için URL'nin frag bileşenini desteklemesi gerek. Örneğin, URL HTML dokümantasyonundaki bir bölümü veya bir resmi işaret edebilir.

Fragment URL'nin sağ tarafında kendini belli eder. Öncesinde # karakteri ile ayrılır zaten görürsünüz anlarsınız yani. Örneğin:

<http://www.joes-hardware.com/tools.html#drills>

Bu örnekte “drills” fragmenti Joe'nun Hardware web serverindeki bulunan /tools.html web sayfasının referans olarak bir kısım olmsutur. O kısmın adı “drills”dir.

Çünkü HTTP serverleri genelde tüm obje ile ilgilenirler, objelerin fragmentleri ile değil. Kullanıcılar fragmenti server tarafına geçiremez. Tarayıcı serverdaki tüm kaynağı getirdikten sonra sizin istediğiniz kaynağın fragmenti ile ilgilenir. Aşağıdaki resimde görülecek az sonra anlatacağınız.

URL KISALTMALARI

Web istemcileri anlar ve URL kısaltmaları kullanır. Relative URL diyeceğimiz kısım kaynagi daha kısaca belirleyerek uygun duruma getirir. Cogu tarayici ayni zamanda “otomatik genisleticiyi” barindirir. Hani g yazariz google.com cıkar ya onlar iste. Hatırlanır tarafi yazınca direk cıkan yerler iste. Bunlar “Expandomatic URL’ler” kısmi altında acıklanacak.

RELATIVE URL’Ler

Bu URL’lerin iki tadi var: absolute ve relative kısmi. Bu zamana kadar biz hep absolut URL’lere değindik ve inceledik. İste yukardaki cafacların hepsi absolute URL’de. Absolute URL ile kaynaga erismek için gereken tüm bilgilere sahibiz.

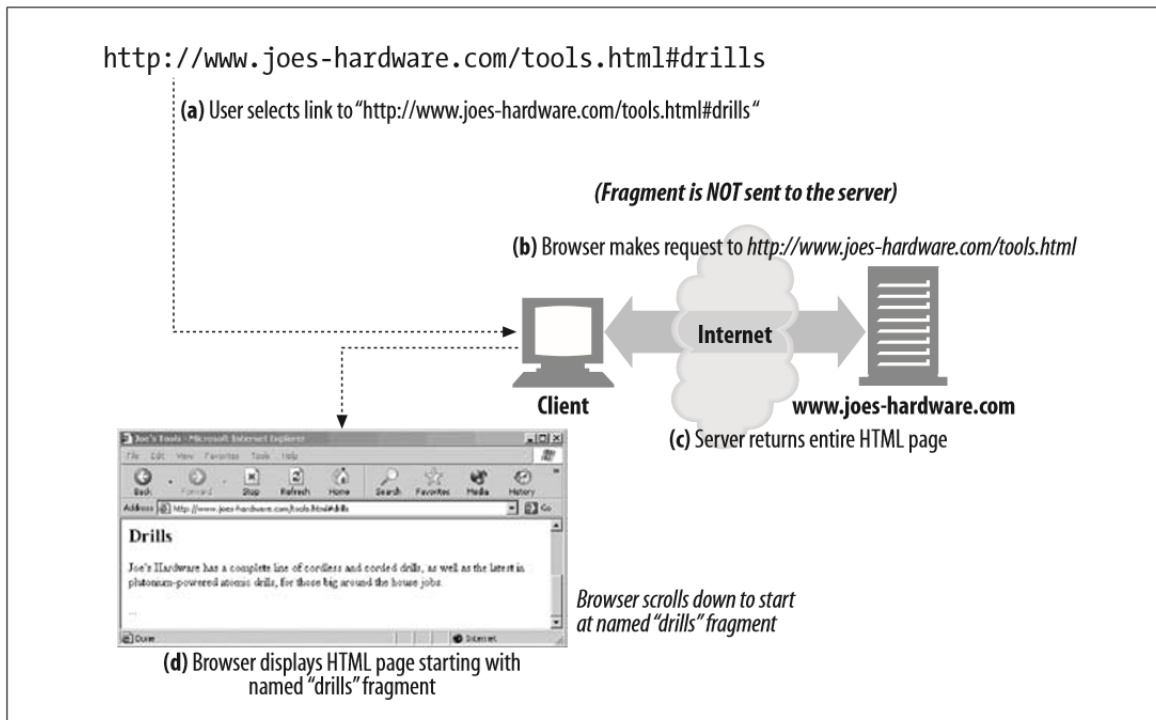


Figure 2-3. The URL fragment is used only by the client, because the server deals with entire objects

Bir diger taraftan, relative URL’ler tamamlanmamis URL’lerdir. Relative URL’lerden Kaynaga ulasmak için gereken tüm bilgileri getirmek için bizim base dedigimiz ilerde anlatacagımız URL’yi yorumlamamız gerek.

Relative URL'ler URL kısaltmaları için uygundur. Eğer hiç HTML yazdıysanız, muhtemelen güzel bir kısayol bulmalısınız. Aşağıdaki örnekte HTML relative URL barındırıyor.

Example 2-1. HTML snippet with relative URLs

```
<HTML>
<HEAD><TITLE>Joe's Tools</TITLE></HEAD>
<BODY>
<H1> Tools Page </H1>
<H2> Hammers <H2>
<P> Joe's Hardware Online has the largest selection of <A HREF="/hammers.html">hammers
</A> on earth.
</BODY>
</HTML>
```

HTML documentimiz burada <http://www.joes-hardware.com/tools.html>

HTML documentinde, bir hyperlink barındıran URL var [./hammers.html](#). Bu URL tamamlanmamış görünüyor. Ama bu Kabul edilen bir relative URL'dir. Burada URL ile ilişkili değerlendirilir ve document bulunur. Bu durumda kaynağa göre [tools.html](#) Joe's Hardware serverindedir.

Kısaltılmış relative URL syntax HTML'i yazan kişiye URL'nin sema, host, ve diğer bileşenlerini atlatmasına izin verir. Base URL tarafından bileşenlerin kaynakların içinde olduğu anlaşılır. URL'ler diğer kaynaklar tarafından kısaca belirlenebilir.

Örnekte bizim base URL'miz.

<http://www.joes-hardware.com/tools.html>

URL base olarak kullanılıyor. Bilgi eksikliği olduğu sonucunu çıkarabiliriz. Kaynağın [./hammers.html](#) olduğunu biliyoruz. Ama sema veya hostu bilmiyoruz. Bu base URL'sini kullanarak, HTTP'nin sema olduğunu hostun www.joes-hardware.com olduğunu çıkarabiliriz. Aşağıdaki resimde anlayacaksınız şimdi.



Figure 2-4. Using a base URL

Yukarıda Relative URL üzerine tüm anlattıklarımızı bu resme bakarak anlayabilirsiniz. Relative URL'ler URL'lerin fragmentleri yada bir parçasıdır. (Applications yada Uygulama diye bahsettiklerim Browser veya tarayıcı olarak anlayın şimdi bahsetme gereği duydum uygulama diyince neyin uygulaması diyebilirsiniz fakar kitapla paralel gidip çevirisini yaptığım için onlar hangi dili kullanıyorsa bende oyle bahsediyorum.) Browserler relative ve absolute url'yi birbirine çevirmesi gerekir. Bunu da beceriyorlar zaten.

Aynı zamanda sunu belirtmekte fayda var URL'lere kaynakların tasınabilir olmasının uygun yollarını sağlar. Eğer relative URL'leri kullanıyorsanız, Documentlerinizi oradan oraya taşıyabilirsiniz ve linkler hala çalışır durumda olur, çünkü base URL tarafından birbirlerine bağlı olarak yorumlanacaktır. Bu aynı zamanda içeriklerin diğer serverlara yansıtılmasına izin verir. Yani bir şey indirirken yansıtıcıları görürsünüz ya işte o bu.

Base URL'ler

İşlem dönüştürme sürecindeki ilk adım base URL'i bulmaktır. Base URL'ler relative URL'lere referans olarak sunulur. Bu bir kaç yerden gelebilir.

Kaynağın içinde açıkça sağlananlar:

Bazı kaynaklar açıkça base URL'i belirtir. Bir HTML documenti, örnek olarak, <BASE> tagini içerebilir. Bu tagle HTML documentindeki tüm relative URL'ler base URL'isini bilir.

Base URL'i kaynağı kısa ve öz biçimde açıklayabilir:

Relative URL base URL'si olmayan bir kaynak bulursa. Aşağıdaki örnekte olduğu gibi, kaynağın URL'ini kullanarak base URL'i bulmaya çalışır. Örnekte yapmıştık bunu.

Hiçbir Base URL olmasa:

Bazı durumlarda, base URL olmaz. Bu sıklıkla sizin absolute URL'ye sahip olduğunuzu gösterir. Ancak, bazen kırık ya da tamamlanmamış URL'ler ile karşılaşabilirsiniz.

Genciler kısacası absolute URL bizim bildiğimiz tüm adrestir. Örnek www.facebook.com/profile gibi mesela her şey açık. Relative URL ise parentleri belli değildir. Yani örnekte yapmıştık mesela [./profile](#) gibi hangi site protocol ne yani sema(http) bilinmiyor. Bir şeye bağlı yani zaten relative kelime anlamı olarak bağlı demek. Şimdi devam edelim.

Relative referanslarını çözme

Öncesinde, URL'nin basit bileşenlerini ve syntaxini gösterdik. Şimdiki adım relative URL'i absolute gibi düşünüp parçalayarak relative ve base URL'olarak bileşenleri ayıracağız.

Gerçekten, URL'i ayırtırsınız, Biz buna sıklıkla URL'i dağıtmak diyoruz. Çünkü bileşenleri ayırıyorsunuz sonuçta. Sema, host, yol(path), parametre, sorgu(?) hepsi ayrılıyor. Base ve relative URL'leri bir kere bileşenlerine ayrıldı mı aşağıdaki resimdeki gibi bir algoritmayla dönüştürülür. Genciler bu resim çok önemli iyice bakın. Büyük Büyük diğer sayfaya vericem.

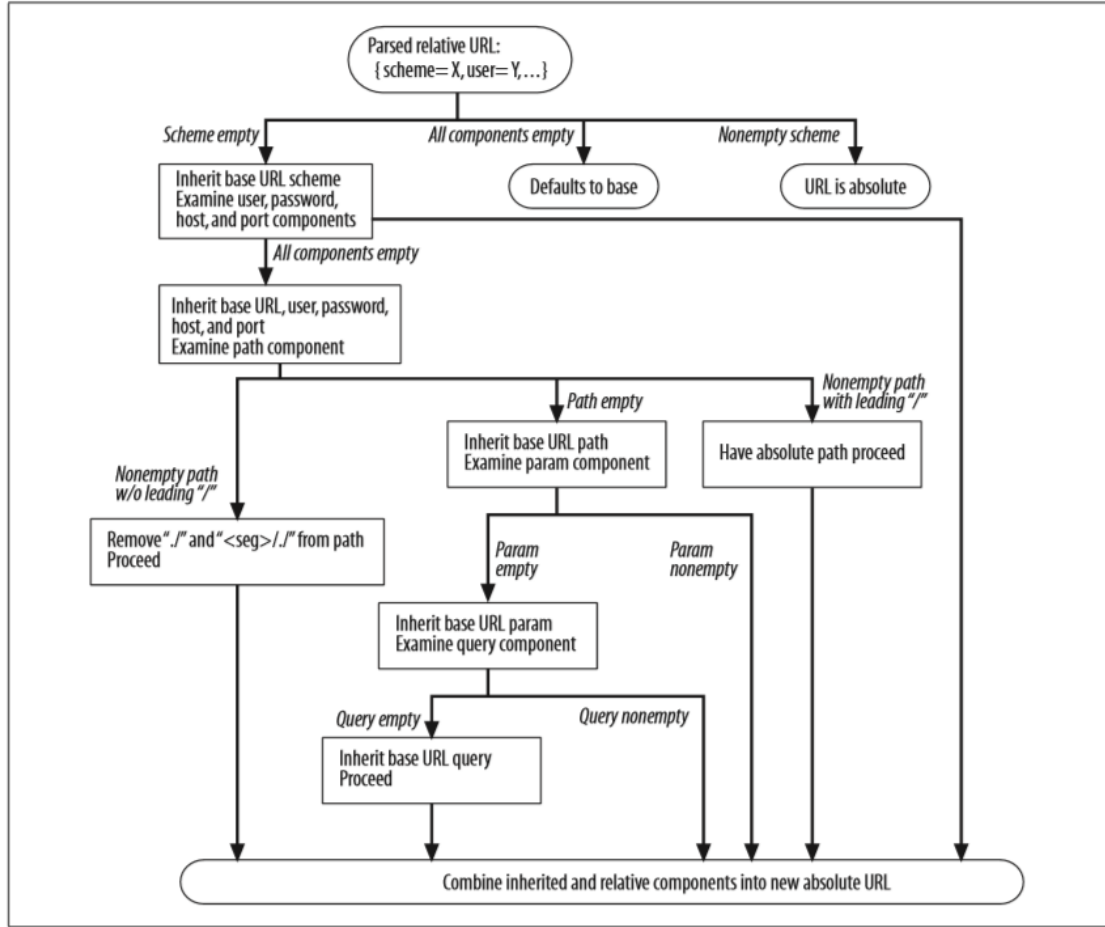


Figure 2-5. Converting relative to absolute URLs

Bu algoritma ile relative URL'yi absolute forma ceviriyoruzki sonradan alisik olarak kaynakla ilgilenelim. Bu algoritma RFC 1808 ve sonradan ortaklasa RFC 2396 orjinal olarak belirlendi.

Bizim ./hammers.html ornegimizi bu algoritmaya uygulayalim gelin.

1. Yol ./hammers.html. Base URL ise http://www.joes-hardware.com/tools.html.

2. Sema bos. Sol taraftan ilerlererek base URL'den sema miras aliniyor. Oradaki inherit base URL sema bu anlama geliyor.

3. En azindan bir bilesen bos degil; Ortaya iniliyor. Host ve port bilesenleri miras aliniyor.

4. Miras aldiklarimizla relative URL'miz (yol: ./hammers.html) combine ediliyor.

Miras aldiklarimiz (sema: http, host: www.joes-hardware.com, port: 80) yeni absolute URL'miz

http://www.joes-hardware.com/hammers.html.

Genisletilmis URL'ler

Bazi tarayicilar(Mozilla,chrome)gibi URL'leri otomatikman genisletir.Yazdiktan sonra veya daha yazarken.Bu kullanicilara kisayol saglar.Tum URL'yi yazmalarina gerek yoktur,cunku otomatikman genisleyerek tamamlar kendimi.

Bu genisleme iki cesit olarak karsimiza cikar:

Hostadi genislemesi(Hostname expansion)

Hostadi genislemesinde,Tarayici siklikla daha hostadini yazarken direk tum hostadini verir hemde sizin yardiminiz olmadan,sadece bir kac basit sezgiyle bunu basariyor tarayici.

Ornegin eger yahoo yazarsaniz tarayicinin address kutusuna,otomatik olarak “www” ve “.com” hostadinin etrafina eklenir ve “www.yahoo.com” yaratilir.Bazi tarayicilar eger siteyi bulmakta yetersiz ise vazgecmeden once birkac deneme yapacaklardir.Tarayicilar bu basit numaraya basvurarak zamaninizdan yirttiracak ve bos yapmayi onleyecektir :D

Ancak bu genisleme numarasi hostadlari uzerinde diger HTTP uygulamalarinda hata cikarabilir.Misal proxyler.Bolum 6 da detaylariyla tartiscaz daha 2.deyiz:D

Gecmis genislemesi(History expansion)

Tarayicilarin kullandigi diger teknik ise siz yazarken gecmiste ziyaret edilenlerden bulup cikarmasidir.Siz yazarken bir suru opsiyon size sunar.Siz zaten bunla cok karsilastiginiz icin bilirsiniz cok detaya girmeden uzatmicam.Misal yo yazin youtube.com cikicak bu gibi iste.

Sunun farkinda olun URL otomatik-genisletme proxylerde farkli sekilde davranir.Bolum 6 da (URI otomatik tamamlama ve Hostname Cozunurlugunde) tartiscaz.Web application security ile ugrasmak istedigim icin ozellike proxy bolumune hizlica gelmek istiyorum.Bakalim ilerde neler ogrenicez.

Shady Karakterleri

Ciki ciki slim shady.May have attention please.real slim shady please stand up.Shady karakterlerinden bahsedicez.

URL'ler tasinabilir olarak tasarlanmistir.Ayni Internetteki tum kaynaklari adlandirmak icin tasarlanmistir.Bu demektri ki cesitli protokollerle tasinma saglanicaktir.Cunku tum protokoller datalari tasimak icin farkli mekanizmalara sahptir.Bu URL'nin tasarlanmisinin onemli amacalarindandirki biz Internette protokolleriyle guvenli tasimalarda bulunalım.

Guvenli tasima demek URL'lerin datadaki bilgiyi kaybetmeden guvenli tasinmasi demek.Bazi protokoller,SMTP gibi(Simple Mail Transfer Protocol)elektronik mailler,tasimada bazi kapsimli karakterleri cikarirlar.Bu kuralin etrafından dolasmak icin,URL'ler sadece herkes tarafından kabul edilmiş guvenli karakterlere izin verir.

Ek olarak URL'nin tüm internet protokolleri için taşınabilir olması istendiği için, dizayn edenler tarafından insanların için okunabilir olarak tasarlanmıştır. Bu yüzden yazılamayacak karakterler görünmez, zaten URL bunlar yasak koymuştur. Buna rağmen bu karakterler maillerden geçebilir ve aksi takdirde taşınabilir olur.

Karışıklık olayı daha önemli olduğu için, URL aynı zamanda tamamlanabilir olması gerek. URL tasarımcıları fark ettikleri kullanıcıları istediği zaman URL'de binary data'sı veya evrensel olarak kabul edilmiş güvenli karakterlerin dışına çıkabiliyorlar. Bu yüzden, kaçış mekanizması eklendi. Güvenli karakterler taşıma için güvenli karakterlere encode edilir. Cross site script'te scriptlerin filtrelenmesi gibi.

Bu bölümün universal alfabe dediğimi şeylerden bahsedicek ve URL için encode kurallarından bahsedicez.

URL Karakter Takimi

Bilgisayar sistemlerinde default olarak karakter takimleri sıklıkla onaylıdır. Tarihe baktığımızda, bilgisayar uygulamaları US-ASCII karakter takimini kullanır. US-ASCII karakterli 7 bit ile temsil edilir. Donanım sinyali ve text formatları için yazılmayan karakter kontrolü vardır. US-ASCII baya taşınabilir. Efsanedir kendisi, ama uygun olarak gösterildiği sırada Avrupa ve Roman dillerini kapsamıyor. Zaten basında US var gördüğümüz gibi.

Ayrıca bazı URL'ler isteğe bağlı olarak binary data içermek zorundadır. Tamamlanmış URL için bu fark edildiğinde, dedemizde dediğimiz gibi URL tasarımcıları kaçış durumları çıkardı. Kaçış durumları isteğe bağlı karakterleri encode etmeye veya kısitli US-ASCII karakterlerle data'yı, taşınabilir ve tamamlanır yapmaya izin verir.

ENCODE MEKANİZMASI

Web app security ile ilgilenenler buraya iyi bakın kutulara script bırakırken bunların tekniklerini bilmeniz önemli.

Güvenli karakter takiminin limitlerinden sıyrılmak için, encode seması tasarlandı. URL'de karakterlerin yerini alıyorki güvenli değil bu. Encoding derki güvenli karakterler varsa kaçış gösterimiyle gösteririm yani (%) ile iki tane hexadecimal digit kullanırım.

Table 2-2. Some encoded character examples

Character	ASCII code	Example URL
~	126 (0x7E)	http://www.joes-hardware.com/%7Ejoe
SPACE	32 (0x20)	http://www.joes-hardware.com/more%20tools.html
%	37 (0x25)	http://www.joes-hardware.com/100%25satisfaction.html

Bu tarz örneklerle tarayıcınızda karşılaşırsınız. Sık kullanılan karakterlerin hexadecimal versiyonunu öğrenmek önemli özellikle web app sec ile uğraşacaklar baya bilinir.

Karakter Kisitlamalari

Az çok karakter URL'nin içinde özel bir anlama gelir. Diğerleri US-ASCII takımında tanımlanmamıştır zaten. ve hala diğerleri dediğimiz karakterler gateway ve protokollerin kafasını karıştırır. Yani kullanımları hayal kırıklığıdır.

Altteki resim URL'deki encode edilen karakterleri listeliyor.Bunlar bir amac kullanmadan once kullanim yerleri ayrilmistir zaten.

Table 2-3. Reserved and restricted characters

Character	Reservation/Restriction
%	Reserved as escape token for encoded characters
/	Reserved for delimiting splitting up path segments in the path component
.	Reserved in the path component
..	Reserved in the path component
#	Reserved as the fragment delimiter
?	Reserved as the query-string delimiter
;	Reserved as the params delimiter
:	Reserved to delimit the scheme, user/password, and host/port components
\$, +	Reserved
@ & =	Reserved because they have special meaning in the context of some schemes

Table 2-3. Reserved and restricted characters (continued)

Character	Reservation/Restriction
{ } \ ^ ~ [] '	Restricted because of unsafe handling by various transport agents, such as gateways
< > "	Unsafe; should be encoded because these characters often have meaning outside the scope of the URL, such as delimiting the URL itself in a document (e.g., "http://www.joes-hardware.com")
0x00–0x1F, 0x7F	Restricted; characters within these hex ranges fall within the nonprintable section of the US-ASCII character set
> 0x7F	Restricted; characters whose hex values fall within this range do not fall within the 7-bit range of the US-ASCII character set

İngilizce'niz biraz varsa okuyabilirsiniz, ilginç şeyler öğrenebilirsiniz.

BIR BIT DAHA

Herhalde merak etmissinizdir guvensiz karakterli kullandigimizda neler oldugunu.Ornegin Joe'nin sayfasini ziyaret edelim.

<http://www.joes-hardware.com/~joe>

“~” karakteri encode edilmez. Bazi transfer protokolleri icin bu sikinti degildir. Ama bazi uygulama developerleri akilsizca guvensiz karakterleri unsafe etmez.

Uygulamalar ince cizgide yurumelidir.Demek istedigimi anlamissinizdir umarim yani dikkatli develop edilmelidir.Uygulamalar icin en iyisi ayrilmis veya guvensiz karakterleri baska bir uygulamaya yollamadan once URL’de convert edilmesidir.Tum guvensiz karakterler encode edildimi,URL Kabul edilmiş forma girip uygulamalar arasında paylasilabilir;Diger uygulamanin kafasi karistmi karakterlerden diye kuskulanmaya gerek yok.

Gercek uygulama URL yi kullanicidan alir en iyi forma nasil sokulur karar verir ve gerekli olanlari encode eder.Cunku URL’deki her bir bileşen güvenli veya guvensizdir.Hangi karakterleri güvenli veya guvensiz olma durumu,Kullanicidan alinan URL’nin encode edilebilme durumuna baglidir.

Tabi ki,bazi ekstrem uygulamalar tum karakterleri encode edebilir.Ki bu önerilmez,Sert ve hızlı bir kural encode karakterleri için olmamalı ve yoktur.Zaten encode karakterleri güvenlidir;Ancak,pratikte bu kırık ve garip davranislara yol acabilir,cunku bazi uygulamalarda güvenli karakterler encode edilmemesi gerekir.

Bazen,zararlı arkadaşlarımız ekstra karakterleri encode etmeye başvurabilir ve uygulamanin etrafından dolanıp URL modeli ile eşleşip kötü şeyleri yapmaya çalışabilir.Orneğin,web filtrelemeli uygulamalar.Güvenli URL bileşenlerini encode etmek,Uygulamanin URL eşleşmelerinde aradiginizi bulmanız konusunda size yardımcı olamayabilir.Genelde,uygulamalar URL’leri işleme koymadan önce decode ederler.

Bazi URL bileşenleri,sema gibi(http,ftp,smtp gibi),daha önce dediğimiz gibi alfabetik karakterle başlamalı ve farkına varılıp okunmalı.Ilgilenenler için ”URL Syntaxi” bölümünde bir resimde ayrilmiş ve guvensiz karakterlerle farklı URL bileşeni resmi var.

SEMA DÜZENİ

Bu bölümde hemen,İnternetteki sık kullanılan sema formatlarına bakıcaz.En sonraki kaynaklar bölümünde baya resim vericez ama şimdilik en popüler semaları vericez zaten bunlar işinize yarayacaktır.Kitap sonundaki semalar bazı deneysel protokolleri bazılarını hiç kullanmıksınız bile su ankiler omrunuza yeter.Simdi çok fazla resim gelio bunlara iyice bakın.

Table 2-4. Common scheme formats

Scheme	Description
http	<p>The Hypertext Transfer Protocol scheme conforms to the general URL format, except that there is no username or password. The port defaults to 80 if omitted.</p> <p>Basic form:</p> <p><i>http://<host>:<port>/<path>?<query>#<frag></i></p> <p>Examples:</p> <p><i>http://www.joes-hardware.com/index.html</i></p> <p><i>http://www.joes-hardware.com:80/index.html</i></p>
https	<p>The https scheme is a twin to the http scheme. The only difference is that the https scheme uses Netscape's Secure Sockets Layer (SSL), which provides end-to-end encryption of HTTP connections. Its syntax is identical to that of HTTP, with a default port of 443.</p> <p>Basic form:</p> <p><i>https://<host>:<port>/<path>?<query>#<frag></i></p> <p>Example:</p> <p><i>https://www.joes-hardware.com/secure.html</i></p>
mailto	<p>Mailto URLs refer to email addresses. Because email behaves differently from other schemes (it does not refer to objects that can be accessed directly), the format of a mailto URL differs from that of the standard URL. The syntax for Internet email addresses is documented in Internet RFC 822.</p> <p>Basic form:</p> <p><i>mailto:<RFC-822-addr-spec></i></p> <p>Example:</p> <p><i>mailto:joe@joes-hardware.com</i></p>

Table 2-4. Common scheme formats (continued)

Scheme	Description
ftp	<p>File Transfer Protocol URLs can be used to download and upload files on an FTP server and to obtain listings of the contents of a directory structure on an FTP server.</p> <p>FTP has been around since before the advent of the Web and URLs. Web applications have assimilated FTP as a data-access scheme. The URL syntax follows the general form.</p> <p>Basic form:</p> <pre>ftp://<user>:<password>@<host>:<port>/<path>;<params></pre> <p>Example:</p> <pre>ftp://anonymous:joe%40joes-hardware.com@prep.ai.mit.edu:21/pub/gnu/</pre>
rtsp, rtspu	<p>RTSP URLs are identifiers for audio and video media resources that can be retrieved through the Real Time Streaming Protocol.</p> <p>The “u” in the rtspu scheme denotes that the UDP protocol is used to retrieve the resource.</p> <p>Basic forms:</p> <pre>rtsp://<user>:<password>@<host>:<port>/<path></pre> <pre>rtspu://<user>:<password>@<host>:<port>/<path></pre> <p>Example:</p> <pre>rtsp://www.joes-hardware.com:554/interview/cto_video</pre>
file	<p>The file scheme denotes files directly accessible on a given host machine (by local disk, a network filesystem, or some other file-sharing system). The fields follow the general form. If the host is omitted, it defaults to the local host from which the URL is being used.</p> <p>Basic form:</p> <pre>file://<host>/<path></pre> <p>Example:</p> <pre>file://OFFICE-FS/policies/casual-fridays.doc</pre>
news	<p>The news scheme is used to access specific articles or newsgroups, as defined by RFC 1036. It has the unusual property that a news URL in itself does not contain sufficient information to locate the resource.</p> <p>The news URL is missing information about where to acquire the resource—no hostname or machine name is supplied. It is the interpreting application’s job to acquire this information from the user. For example, in your Netscape browser, under the Options menu, you can specify your NNTP (news) server. This tells your browser what server to use when it has a news URL.</p> <p>News resources can be accessed from multiple servers. They are said to be location-independent, as they are not dependent on any one source for access.</p> <p>The “@” character is reserved within a news URL and is used to distinguish between news URLs that refer to newsgroups and news URLs that refer to specific news articles.</p> <p>Basic forms:</p> <pre>news:<newsgroup></pre> <pre>news:<news-article-id></pre> <p>Example:</p> <pre>news:rec.arts.startrek</pre>

Table 2-4. Common scheme formats (continued)

Scheme	Description
telnet	The telnet scheme is used to access interactive services. It does not represent an object per se, but an interactive application (resource) accessible via the telnet protocol. Basic form: <code>telnet://<user>:<password>@<host>:<port>/</code> Example: <code>telnet://slurp:webhound@joes-hardware.com:23/</code>

GELECEK

URL guclu bir arac.Dizayn olarak tum objelere ve cikarilmis formatlara izin veriyor.Uniform isimlendirme mekanizmasi ile internet protokolleri arasi paylasimi sagliyor.

Ancak mukkemmell degil.URL gercekten adressler,lakin isim degildir.Bu demektirki URL o anda kaynakin nerde oldugunu soyley.Isim ile belli bir server da belli bir portu saglar.Kaynagin buldugunuz yer iste.Eger kaynak tassinmissa sema yoktur.Yani URL artik mevcut degildir.Bu noktada,objenin yerini belirlenemez.

Eger objenin gercek ismine sahip olunsaydi ideal ne olur.Objenin yerine bakinmak icin ismi kullanabilirdiniz.Bir insan olarak,kaynagin ismini vermek ve az cok diger faktorler ile,kaynagin yerini takip edebilirdiniz nereye gittigine bakilmaksizin.

IETF(Internet Engineering Task Force) yeni standartlar uzerine calisiyor,URN'ler(Uniform Resource Names),bazi zamanlar icin,adreste sikinti cikarabiliyor.URN obje icin stabil isim saglar.Obje nereye tasinirsa tasinsin.

Kisaca uzatmayayim URN'ye gecisler icin calisilio kaynakin ismini bilerek nerde oldugunu hep bilebilirsiniz.Sonucta biz URL ile kaynakin konumunu yani adresini belirtiyoruz ama IPv6 gibi tabi geceler diyelim devam edelim.Simdiki baslik benim dedigimi gelio zaten

Simdi Degilse,Ne zaman?

URN hakkındaki bazi fikirler akla geliyor bezen,Dogrudur.Eger URN icin belirlenen zamana bakarsiniz,kendinize neden adapte olamadigini anlatabilirsiniz.Ya ben anlatiyorumda zaten turk milletiyiz kardesim al bilgiyi git demi :D.

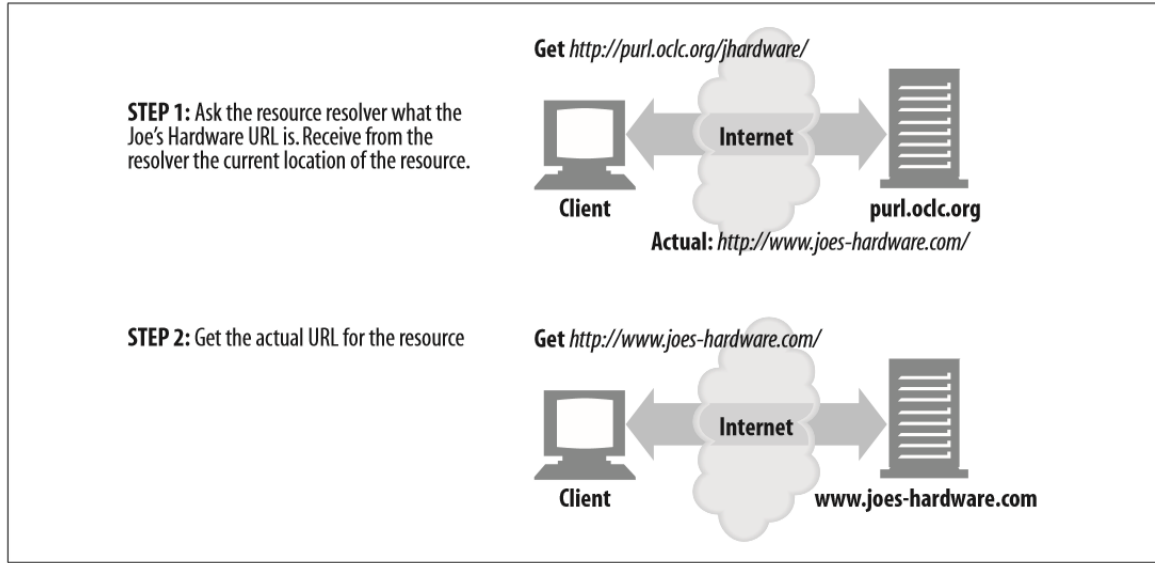


Figure 2-6. PURLs use a resource locator server to name the current location of a resource

URL'den URN'ye donmek sikintili bir is.Standartlastirma isleri yavas ilerler ki bu iyidir.URN desteklemek baya degisiklige gitmemize sebep olur.HTTP'nin Body kisimlariyla oynanir,HTTP uygulamalari cesitleri donusume ugrar gibi.Bu kritik degisim icin cok degisiklige ihtiyac var,ve maalesefki(belkide uzuntuyle beraber)URL arkasinda baya momentum var arkasi saglam yani:D.

Webin verimli bir sekilde buyumsiyle beraber,Internet kullanicilari-bilg muhlerinden normal kullanicilara kadar-URL kullanmayi biliyor.Salak bir syntaxla aci cekerlerken(acemilerden bahsediyorum) ve temellestirme problemleri varken,Insanlar URL ile anlasmayi ogrendiler.URL biraz limite sahip ve web development kominutesi bu sikintiyla basil basina ilgilenmiyor.

Su anlik,gorulebilir bir gelecekte,URL internette kardesim.Her yerdeler ya ve internetin cok onemli bir parcesi gibi.BESIKTAS gibiler yani burdalar.Ondan sikinti yok URN ye gecilemez daha sikintilari var onların.Haydi 3.bolumde gorusmek uzere.