

ÇOMAR

(Configuration Manager)

Nedir?, Yenir mi?

Dürtücü Teknolojiler Ar&Ge Labs.

16 Mart 2005

Özet

Bu belge, ÇOMAR'ın ne olduğunu ve ÇOMAR gibi bir altyapıya neden ihtiyaç duyulduğunu mimari ve yöntem ile ilgili teknik ayrıntılara girmeksizin, basit bir dille anlatmayı hedeflemiştir. Çok genel bir giriş teşkil etmesi amacıyla en basit anlamda ÇOMAR, bir yapılandırma altyapısı olarak tanımlanabilir.

İçindekiler

1 Giriş	2
2 Linux Dağıtımları, Mevcut Durum	2
3 Nasıl Olmalıydı?	3
4 ÇOMAR	4
5 Daha Fazlası	5
6 Sıkça Sorulan Sorular	5
6.1 ÇOMAR'ın açılımı nedir?	5
6.2 ÇOMAR ve ÇOMAR için hazırlanmış uygulamalar bana ne sunacak?	6
6.3 Sistemde ÇOMAR ile çalışacak şekilde düzenlenmemiş programları da kullanabilecek miyim?	6
6.4 ÇOMAR'ı devreden çıkarabilecek miyim, bu hali ile sistemi kullanmaya devam edebilecek miyim?	6
6.5 ÇOMAR sistemde kendi kendine bir takım işler yapacak mı?	6
6.6 CSL yeni bir programlama dili mi? Bu dili kim kullanacak?	7
6.7 CSL gibi yeni bir dile ihtiyaç var mıydı? Diğer diller yetmediler mi?	7
6.8 ÇOMAR bu kadar çok sorunu çözüyorsa neden şu ana kadar kimse düşünemedi?	7

1 Giriş

Herhangi bir bilgisayar kullanıcısının bilgisayardan beklentisi belirli ve bir tanedir “bilgisayar ile iş yapmak, bilgisayara bir iş yaptırmak”. Bu bahsi geçen iş,

- Film izlemek,
- Yazı yazmak, yazıcıdan çıktı almak,
- Internet’te oyun oynamak,
- Yeni monitörü denemek,
- E-posta okumak,
- CD yazmak,
- Yeni bir uygulama kurmak,
- Bilgisayarın Internet’e yalnız belirli saatlerde bağlanabilmesini sağlamak,

ve benzeri onlarca şey olabilir. Bu işlerin herhangi birisi için, bu işler için kullanılacak uygulamaların sistemin diğer bileşenleri hakkında fikir sahibi olması ve sisteme uygun şekilde yapılandırılmış olması gerekmektedir. Eğer daha önceden yapılandırılmamışsa, bir “iş”i yapabilmek için gereken uygulamaların yeteneklerinin ve nasıl yapılandırılacaklarının bilgisi kullanıcı tarafından biliniyor olmalıdır.

Herhangi bir nedenle bilgisayarın çalışma koşulları değiştiğinde de daha önce yapılan işleri aynı şekilde yapabilmek için kullanılan uygulamaların değişikliklerden haberdar edilmesi de gerekmekte ve hangi iş için hangi uygulamaların nasıl değişikliklere ihtiyaç duyduğu da yine kullanıcı tarafından biliniyor olmalıdır.

2 Linux Dağıtımları, Mevcut Durum

Linux dağıtımı, Linux çekirdeği kullanan bir işletim sistemi ortamını kullanıcıya sunmayı hedefleyen bir bütündür. Dağıtımlar bu işletim sistemi ortamını, kullanıcının ihtiyaç duyabileceği çeşitli uygulamaların beraberce ve uyum içerisinde çalışabilecekleri şekilde bir araya getiririp, tüm bunları kolaylıkla kurulabilir şekilde kullanıcıya ulaştırabilmeyi hedeflerler.

Linux dağıtımlarının büyük bir kısmı aynı zamanda kullanıcının çalışma zamanındaki ihtiyaçlarına cevap vermek üzere bir takım yapılandırma araçlarını da kullanıcıların hizmetine sunar. SuSe, Mandrake, Fedora gibi son kullanıcıyı da hedef alan Linux dağıtımlarının her birinin kendilerine ait, oturmuş yapılandırma araçları mevcuttur.

Kullanıcı bu araçlar sayesinde üzerinde çalıştığı sistemi ya da yapılandırma aracında yer verilmiş kimi özel uygulamaları nispeten kolayca yapılandırabilir. Fakat kullanıcı için aşağıdakiler hem kullanıcı hem de sisteme gelen uygulamalar için hala birer sorundur:

Kullanıcı Tarafından (Uygulamaya Özel Bilgi Gerekliliği):

Kullanıcılar uygulamaları birbirleri ile uyumlu şekilde çalıştırabilmek için bu uygulamalara özel bilgiye sahip olmak zorunda kalmaktadır. Bu da sistemi verimli bir şekilde kullanabilmeleri için zaman harcayarak araştırma yapmalarına, çoğunlukla yapmak istedikleri işi çok zorlanarak yapmalarına zaman zaman da yapamamalarına neden olmaktadır. Örneğin bir kullanıcı yapılandırma aracı ile yapılandırabileceği web sunucusunun yeteneklerini tanımıyorsa, karşısına çıkan

```
``değişken(label) = değer(input box ya da combo box)``
```

şeklindeki yapılandırma seçenekleri kullanıcıya bir şey ifade edememektedir. Bu da kullanıcının yapması gereken işten ziyade bunu yaparken hangi uygulamaların nasıl yapılandırılmış olması gerektiğini araştırmaya ve bu konu hakkında belki daha sonra hiç kullanmayacağı bilgileri bir şekilde edinmeye mecbur kalmaktadır. Bu durum, geniş bir perspektifle ele alındığında Linux'un gücünden yaygın bir şekilde faydalanamamak anlamına gelmektedir.

Uygulama Tarafından (Yeteneklerini İfade Edememeleri):

Uygulamaların kendi yeteneklerini tam olarak ortaya koyabilecekleri bir platform olmadığından dolayı, uygulamaların bir arada çalışmaları ile çözülebilecek sorunların aşılması, ya da uygulamaların bir arada çalışmaları ile elde edilebilecek yararlardan istifade edilebilmesi için çok fazla efor gerekmektedir. Çoğu zaman da normal olarak kullanıcı tarafından böyle çözümlerin varlığı bile düşünülememektedir. Linux ile herşeyin çözümü makul kolaylıktaki yollardan mümkün iken bu çözüm yöntemlerini sadece uygulamaların yeteneklerini daha önce tecrübe etmiş uzman kişiler kullanabilmektedirler.

3 Nasıl Olmalıydı?

İşletim sistemi içerisinde, herhangi bir uygulamayı sisteme maksimum verim ile entegre edebilmek ve uygulamayı maksimum verim ile kullanabilmek için sahip olunması gereken bilgi yükünü kullanıcının sırtından alacak işleyiciler olmalıdır. Uygulama, yapılandırması ve entegrasyonu ile ilgili kendi sorumluluğunu kendisi taşımalıdır, bu bağlamda sisteme gelen bir uygulama, yapabileceği işler ile ilgili bilgi ve yordamları kendi içerisinde barındırmalıdır. Bu bir nevi, HOWTO belgelerinin bir sistem modeli çerçevesine uydurularak bir bilgisayarın anlayabileceği şekle getirilmesi ve uygulama ile beraber hareket etmesinin sağlanması anlamına gelmektedir. Bu tip bir yaklaşımın sağlıklı bir şekilde hayata geçirilebilmesi için,

- Bir işletim sisteminde ihtiyaç duyulabilecek yapılandırma ihtiyaçlarını kavramsal olarak ortaya koymuş bir yapılandırma altyapısı, bir model olmalıdır,

- Uygulamaların sistem ile ve kendi aralarında konuşabilmeleri için bu modeli kullanan yapılandırma altyapısının sunduğu standart bir arayüz olmalıdır,
- Uygulamaların, yetenekleri ve kendilerinden beklenen çeşitli işleri nasıl yapabilecekleri ile ilgili deneyim, uygulamaların içerisine gömülebilmelidir,
- Uygulamaların içerisine gömülen bilgiler hem uygulamanın hem de içine gömülen bilgilerin taşınabilirliğini koruyabilmesi ve uygulamanın yazıldığı dilden bağımsız şekilde kolayca geliştirilebilmesi için uygulamanın orjinal kaynak kodunda herhangi bir değişiklik yapılmasını gerektirmemelidir,
- Uygulamalar sistem içerisinde kendi yeteneklerinden ve karşılayabilecekleri fonksiyonliteden yapılandırma altyapısını haberdar edebilmelidir,
- Uygulamalar sistemin geri kalanındaki yapılandırma bilgilerine yapılandırma altyapısının sunduğu standart bir arayüzden ulaşabilmelidir,
- Uygulamalar sistemin geri kalanındaki uygulamaların sunduğu fonksiyonlara yapılandırma altyapısı üzerinden erişebilmeli ve yeri geldiğinde onlara yetkileri dahilinde hareket ettiklerini garanti eden bir yolla iş yaptırabilmelidir.

Bu sayede,

- **Kullanıcı** sisteme yeni eklenen bir uygulamanın beklendiği gibi çalışabilmesi için, onu yapılandırmaya çalışmak zorunda kalmayacak, uygulama paketlerine takılan küçük bileşenler, uygulamanın sisteme geldiği zaman ortalığa göz atıp sistemi tanıyabilmesi ve kendisinin nasıl bir yerde olduğunu anlayabilmesi için gözü kulağı olacaktır.
- **Uygulama geliştiricisi** için yapılandırma altyapısının sunduğu olanaklardan faydalanan, mevcut durumdaki gibi kullanıcıdan uygulamaya özel bilgi beklemeyen, uygulamadan bağımsız ve görev temelli yapılandırma ön yüzleri inşa etmek özellikle mevcut durum ile kıyaslandığında son derece kolay olacaktır.
- **Sistem yöneticisi** için sistemin tüm yapılandırmasının standart bir altyapı üzerine oturtulması, bununla beraber beraber yapılandırmanın kolayca taşınması, kullanıcı seçimlerine bağlı profillerin oluşturulması ve yönetilmesi de kolay olacaktır.

4 ÇOMAR

ÇOMAR, tüm bu bahsedilen gerekliliklerin hayata geçirilebilmesi, uygulama isteklerinin belirli bir sıra düzende ve erişim kontrolü ile işletilebilmesi için gereken zemin ve bileşenlerini sunacak olan yapılandırma altyapısı olarak tasarlanmıştır.

ÇOMAR yapılandırma altyapısının tamamı temel olarak dört bileşenden meydana gelmektedir, bu bileşenler basit tanımları ile şu şekilde listelenebilir:

Sistem Modeli: Bir işletim sistemindeki yapılandırma ihtiyaçlarının kavramsal olarak nesnelerle tasvir edildiği modeldir. Bu model sayesinde, uygulamalar sistem modeli içerisindeki beklentilerden hangilerine cevap verebileceklerine karar verebilecek ve kendilerini Sistem Modeli'nin uçları ile ilişkilendirebileceklerdir. Sistem Modeli ÇOMAR'ın en önemli parçasıdır ve getirdiği en büyük yeniliktir.

CSL Betikleri: ÇOMAR'ın Uygulama paketi içerisinde çıkacak olan parçasıdır. Bu betikler içerisinde Sistem Modeli'nde adı geçen fonksiyonların uygulama tarafından nasıl gerçekleştirileceğine dair anlatım olacaktır.

CSL Yorumlayıcı: ÇOMAR'ın uygulama paketi içerisinde çıkan ve sistem modeline yerleşen betikleri işletebilmesi için onları yorumlayacak ve ÇOMAR'a sunacak bileşendir.

ÇOMARd: İşletim merkezidir. Erişim yetkilerini kontrol etmek, görevleri belirli bir sıra düzende yerine getirtmesini sağlamak, yapılandırma için kullanılabilecek görev temelli kullanıcı grafik ön yüzlerine standart sorgu arayüzleri sunmak, sistemdeki değişikliklerden gerekli uygulamaların betiklerini haberdar etmek, uzakdaki ÇOMARd'lere isteklerde bulunmak, onlardan gelen istekleri karşılamak gibi daha alt seviyeli işlevleri yerine getirecektir.

5 Daha Fazlası

Bu belgenin amacı ve şu ana kadar izlediği yaklaşım ÇOMAR'ın ne olduğunu ve ne olmadığını basit bir dil ile anlatmak idi. ÇOMAR'ın mimarisi, Sistem Modeli ve CSL ile ilgili teknik belgelerin hazırlanması da ÇOMAR'ın geliştirilmesi ile paralel bir şekilde ilerlemektedir. Diğer belgelerin ve bu belgenin son sürümlerini <http://www.uludag.org.tr/belgeler.html> adresinden takip edebilirsiniz. Ayrıca ÇOMAR ve ULUDAĞ'ın diğer bileşenleri ile ilgili yanıt bulamadığınız sorularınızı, düşüncelerinizi ya da fikirlerinizi <http://www.uludag.org.tr/listeler.html> adresindeki e-posta listelerimiz aracılığı ile bize iletebilirsiniz.

6 Sıkça Sorulan Sorular

6.1 ÇOMAR'ın açılımı nedir?

“Configuration Manager”. ÇOMAR'ın açılımı “Configuration by Objects Modify and Restart” idi. Fakat ÇOMAR'ın tasarım sürecinde “Modify and Restart” kısmının ÇOMAR'ın işlevselliğini tam olarak ifade etmez hale geldiği görüldü ve açılımının “Configuration Manager” olmasının daha doyurucu ve anlamlı olacağına karar kılındı.

6.2 ÇOMAR ve ÇOMAR için hazırlanmış uygulamalar bana ne sunacak?

- ÇOMAR sistemin mevcut yapılandırmasına göre kendi kendini yapılandırabilen uygulamalar hazırlanabilmesini, sistemdeki yapılandırmanın değiştirilmesi ile haberdar edilmesi gereken uygulamaları haberdar etmek gibi işleri kullanıcının üzerinden alarak yapılandırma bağlamında kaybedilen zamanı ve enerjiyi kullanıcıya geri kazandıracaktır.
- Yeterince deneyimli kişiler tarafından hazırlanmış betikler sayesinde kurulum ve çalışma zamanında başlangıç kullanıcılarının dahi uygulamaların yeteneklerini tam anlamıyla kullanabilmelerini sağlayacaktır.
- Sunduğu fonksiyonalite ile tek tek uygulama yapılandırmaktan ziyade görev temelli düşünülmüş grafik yapılandırma ön yüzleri yazılmasını kolaylaştıracak, bu ön yüzler sayesinde kullanıcının daha rahat “iş” yapabilmesini sağlayacaktır.

6.3 Sistemde ÇOMAR ile çalışacak şekilde düzenlenmemiş programları da kullanabilecek miyim?

Elbette. Bunun sebebi de, uygulamaların ÇOMAR ile çalışabilecek şekilde düzenlenmesinin, uygulamanın kaynak koduna dokunmaksızın, uygulama ile beraber gelecek bir ya da bir kaç betik dosyasının uygulama paketi içerisine eklenmesinden ibaret olmasından kaynaklanmaktadır.

6.4 ÇOMAR’ı devreden çıkarabilecek miyim, bu hali ile sistemi kullanmaya devam edebilecek miyim?

Evet ÇOMAR’ı devreden çıkarabileceksiniz, ve sisteminiz çalışmaya devam edecek. Fakat bunu yaptığınızda herhangi bir konfigürasyon değişikliği artık kendi kendine olmayacak, uygulamalar değişikliklerden haberdar olup kendilerini yeniden yapılandıramayacaklardır. Fakat bir Linux sisteme aşına olanlar yapılandırma dosyalarına müdahale ederek sistemin yapılandırmasının sürekliliğini elle sağlayabileceklerdir. ÇOMAR’ın aktif olmaması sistemi çalışmaz hale getirmeyecek fakat olması sistemi iyi çalışır hale getirecektir.

6.5 ÇOMAR sistemde kendi kendine bir takım işler yapacak mı?

ÇOMAR sistemde, konfigürasyon değişikliğini, bir uygulamanın tetiklenmesini ya da bir servisin yeniden başlatılmasını gerektirecek bir “olay” meydana geldiğinde ya da kullanıcı grafik arayüzlerinden gelecek bir çağrı neticesinde çalışacak ve sisteme müdahale edecek, kendi kendine bir takım işler yapmayacaktır.

6.6 CSL yeni bir programlama dili mi? Bu dili kim kullanacak?

CSL, ÇOMAR'ın Betik Dili olarak düşünöldü. Bu dili sadece ÇOMAR'dan faydalanmak isteyen uygulamaların paketleyicileri gerekli betikleri paketlere eklemek için öđrenecekler. CSL ile ilgili dökümanları ilerleyen tarihlerde dokumanlar sayfamızdan edinebileceksiniz. Kullanıcıların bu dili bilmesi gerekmeyecek.

6.7 CSL gibi yeni bir dile ihtiyaç var mıydı? Diğer diller yetmediler mi?

CSL tam olarak diğer programlama dillerinin yetersizliğinden dolayı ortaya çıkmış bir programlama dili deđil. CSL, ÇOMAR'ın beklediđi yapıya uygun çağrılarının minimum hata ve efor ile, kolaylıkla üretilebilmesi ve ÇOMAR'a iletilebilmesi için geliştirilmiş, son derece basit, Python temelli bir dil ve yorumlayısından ibarettir.

6.8 ÇOMAR bu kadar çok sorunu çözüyorsa neden řu ana kadar kimse düşünemedi?

ÇOMAR'ın gücü ve en büyük getirisi Sistem Modeli'nden gelir. Daha önce de dağıtım üreticileri yapılandırma ile ilgili sorunları çözmek için çeşitli yöntemler düşündüler. Onların çözümleri çođunlukla yapılandırma dosyalarına standart erişim için çeşitli yalıtım katmanları ve uygulamalara özel yapılandırma araçlarının ötesine geçmedi. řu ana dek geliştirilen yaklaşımların sorunun hep bir kısmını çözmesine rağmen, gerçek yapılandırma sorununun tüm sistemi kucaklayacak bir sistem modelinin sunduđu platform üzerinde çözümediđi taktirde asla beklenen noktaya ulaşamayacağını düşünöyoruz. Daha önce bunu benzer şekilde dile getiren ve uygulamaya çalışanlar da oldu, bunlardan en meşhur olanı da dünyanın çok büyük bir kısmını kullanıcı kitlesi olarak kendi etrafına toplamayı başarmış olan bir firma. ÇOMAR'inkine benzer yaklaşımların daha önce de düşünölmendiđini söylemek haksızlık olur, fakat neden řu ana kadar uygulanmadıđı sorusunun da tek ve net bir yanıtı yok.