

DİZİN YAPISI (Directory) UYGULAMALARI

Mürsel Ö. KARABIYIK
Teknik Danışman
Gantek Teknoloji

mursel.karabiyik@gantek.com

Ajanda:

- Şimdiki Durum
- Dizin Yapısı Nedir ?
- Dizin Yapısının Fonksiyonel Özellikleri Nelerdir?
- Dizin Yapısı Kimler için Önemlidir ?
- Dizin Yapısının Ayrıntıları Nelerdir ?
 - Dizin yapısı dışarıdan nasıl görünür ?
 - Dizin yapısı uygulamaları hangi Protokolleri kullanır ?

...Ajanda:

- Alternatifler Var mı ?
 - Alternatifler ile Dizin Yapısı Arasındaki Farklar Nelerdir ?
 - Meta Directory Sunucusu
 - RDBMS / Dizin Sunucusu, Hangisi?
- Uygulamaları Dizin Yapısı ile Uyumlu Geliştirmek için Nelere Dikkat Etmek Gerekir ?
- Dizin Yapısı Bir Kuruma Uygulanırken Nelere Dikkat Edilmelidir ?
- Örnek Çalışma:
 - Bir Üniversitenin İhtiyaçları Nelerdir ?
 - Dizin Yapısı Bu İhtiyaçları Nasıl Karşılar ?

Şimdiki durum:



...Organize Olmamış Bir Yapı:



...Arzulanan Yapı

Kullanıcılar

**Sistemler /
Bilgi Kaynakları**

Dizin Yapısı tabanlı Arayüz



...Ortak Bilgi Saklama



Dizin Yapısı Nedir ?

Çeşitli biçimlerdeki verilerin, organize olmuş (sorgulanabilir, yönetilebilir) bir yapı altında toplu şekilde tutulmasıdır.

Saklanabilecek bilgi çeşitleri:

- İsim, Soyisim
- Fotoğraf
- Telefon numarası, adres, e-ileti adresi
- Kaynak ulaşım bilgileri (yetkilendirme)
- Uygulama konfigürasyonları
- Server, yazıcı, kullanıcı bilgisi/şifresi, ağ cihazları konfigürasyonları
- Kullanıcıya özel çeşitli uygulamalara ait bilgiler
- PKI anahtarları
- Yukarıdakilerin dışında **sık** ve **hızlı erişim** gerektiren aklınıza gelen HERŞEY!!

Dizin Yapısının Fonksiyonel Özellikleri Nelerdir ?

- Bütün kullanıcı ismi/şifre ya da public key tabanlı doğrulama işlemleri tek noktadan takip edilir ve yönetilir.
- Bilgiler, organizasyon yapısına benzer bir hiyerarşik mimari üstünde tutulur.
- Çok kolay yönetilebilir, her birim kendi dizin yapısından sorumlu olabilir.
- Bir çok uygulama kolaylıkla dizin altyapısında saklanan bilgilere ulaşabilir – bilgilerin ortak kullanımı.
- Okuma ağırlıklı işlemler için optimize edilmiştir. Erişim hızı yüksektir.
- Nesne tabanlı bir mimariye sahip olduğu için uygulama geliştiriciler çok rahat adapte olabilirler.
- TCP/IP tabanlı olduğu için bilgiye her konumdan ulaşabiliriz.

Kimler için Önemlidir ?

- Kullanıcılar
- Uygulama Geliştiriciler ve Sistem Yöneticileri
- Karar verme mekanizmalarının başında duranlar

...Kullanıcılar İçin Ne Önemlidir ?

- Başka kullanıcıları bulmak
- İletişim Kurmak
- Bilgi paylaşımı
- Projelerde ortak çalışma yapmak
- İş ile ilgili uygulamalarını çalıştırmak
- Kişisel tercihe uygun İşletim Sistemini çalıştırmak
- Güvenilirlik
- Kolay kullanım

...Uygulama Geliştiriciler ve Sistem Yöneticileri için Ne Önemlidir ?

- Plug-n-Play yükleme araçları
- Esnek tanı araçları - diagnostik kolaylığı
- Bilgi dönüştürme arayüzleri ve araçları
- Güçlü fakat basit SDK
- Uygulama geliştirme ve yönetimsel anlamda eğitim imkanları
- Sunucu İşletim Sistemi taşınabilirliği
- Dökümantasyon

...Karar Verme Mekanizmaları için Ne Önemlidir ?

- Düşük başlangıç ve bakım maliyetleri
- Artan verimlilik
- Azalan IT operasyon maliyetleri
- Açık standartlar desteği
- Yatırımın geri dönüşü
- Geleceği hedefleyen IT yatırımları

...Bunu Nasıl Sağlarız ?

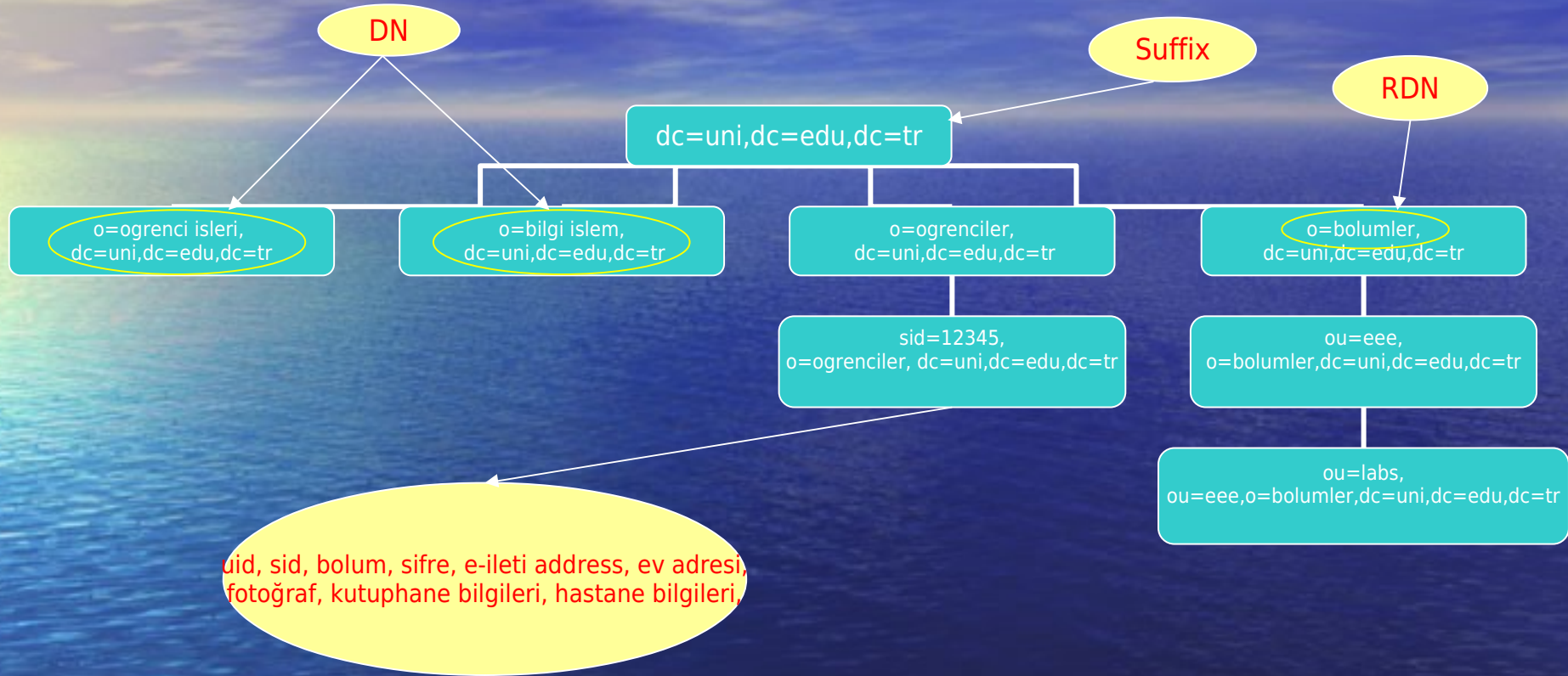
- Basit, iyi tanımlanmış dizin ulaşım protokolü (Directory Access Protocol)
- Basit API
- Güvenlik anahtarlarının saklanabilme desteği
- İşletim Sistemi taşınabilirliği
- Kopyalama/Eş zamanlı kılabilme (Replication/Sync.)
- Yüksek performanslı arama motoru
- Genişletilebilir dizin şeması
- Varolan Intranet ile kolay entegrasyon

destekleyen bir Protokol lazım..

Dizin Yapısının Ayrıntıları Nelerdir ?

- **Çok hızlı okuma işlemleri:** Yazma ve değiştirmeden çok okuma için optimize edilmiştir.
- **Statik sık sık değişmeyen bilgi:** Saklanan bilgilerin büyük çoğunluğu değişmez.
- **Dağıtık:** Kullanılan bilginin doğasından geliyor.
- **Hiyerarşik:** Bilgiler organizasyonun yapısına benzer bir yapı altında saklanır.
- **Nesne tabanlı:** Dizinde saklanan entryler objectler halinde dururlar. Her bir object ise inherit edilmiş olan object classlarda bulunan attribute' ları kullanır.
- **Standard Şema:** Bütün uygulamalar standard şemalar ile çalışırlar. Bu sayede uygulamalar arasında uyumsuzluk yaşanmaz.
- **Çok-değerli Attributelar:** Dizin attributeları bir yada birden çok değere sahip olabilirler.
- **Multi-Master Replication:** Dizin sunucuları aynı anda bir yada birden çok dizin sunucusuna kendisinde olan değişiklikleri anlık yada daha sonra kopyalayabilir.

...Dizin yapısı dışarıdan nasıl görünür ?



...Dizin Yapısı Uygulamaları Hangi Protokolleri Kullanır ?

- X500 ve LDAP
- X500 > LDAP

X500	LDAP
OSI Protocol Stack' i kullanır	TCP/IP kullanır
Karmaşık ve Ağır Protocol	Kolay ve hafif - Lightweight
Chaining yapısına dayanır	Referral yapısını kullanır
Tek bir server görüntüsü sergiler	Dağıtık görünüm sergiler
Binary OID kullanılır	Attribute tipleri stringlerden oluşur

Alternatifler Var mı ?

- RDBMS
 - MySQL
 - PostgreSQL
 - Oracle
 - M\$SQL
- Metin Dosyaları

...Alternatiflerin Dizin Yapısından Farkları Nelerdir ?

RDBMS	Metin Dosyaları
<ul style="list-style-type: none">•Yazma ağırlıklı işlemler•Geçmişe dönük bilgi saklama•Her uygulama için ayrı şema yapısı•Karmaşık bilgi modelleri (view, join, foreign key)•Bilgi bütünlüğü (rollback, referential integrity, transaction-oriented)•ACID transactions<ul style="list-style-type: none">- Atomic,- Consistent,- Isolation ve- Durable	<ul style="list-style-type: none">•Yönetim yok!!!•Sadece bilgi•Aynı anda ulaşım problemleri•Bilgi bütünlüğünü sağlamak imkansız

Altyapımızda Böyle bir Bilgi sunucusu varsa ne yapacağız?



Meta Directory Sunucusu !!

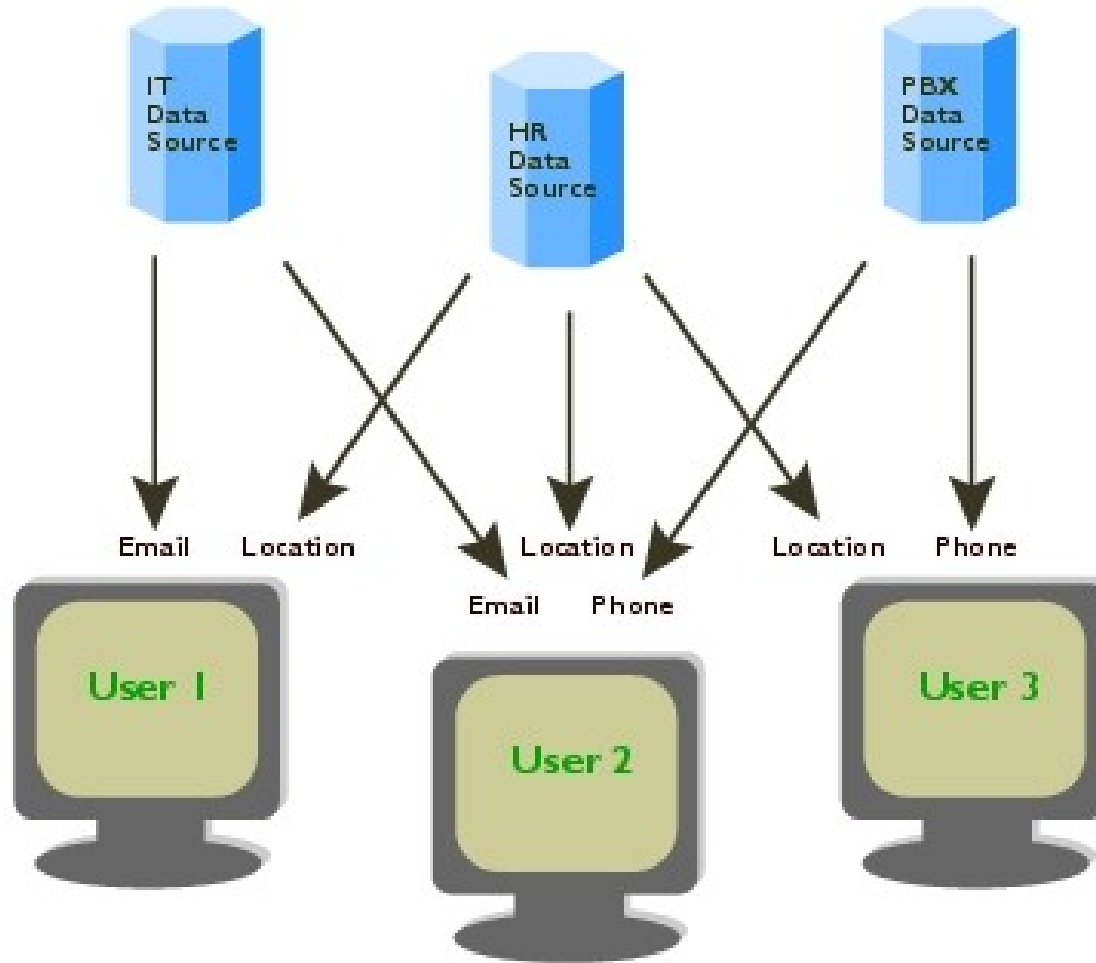
Meta Directory Sunucusu

Birbirine benzemeyen dağıtık veri kaynaklarına tek bir Dizin verisi görünüşü sağlayarak dizin servislerinin avantajlarını ortaya koyar.

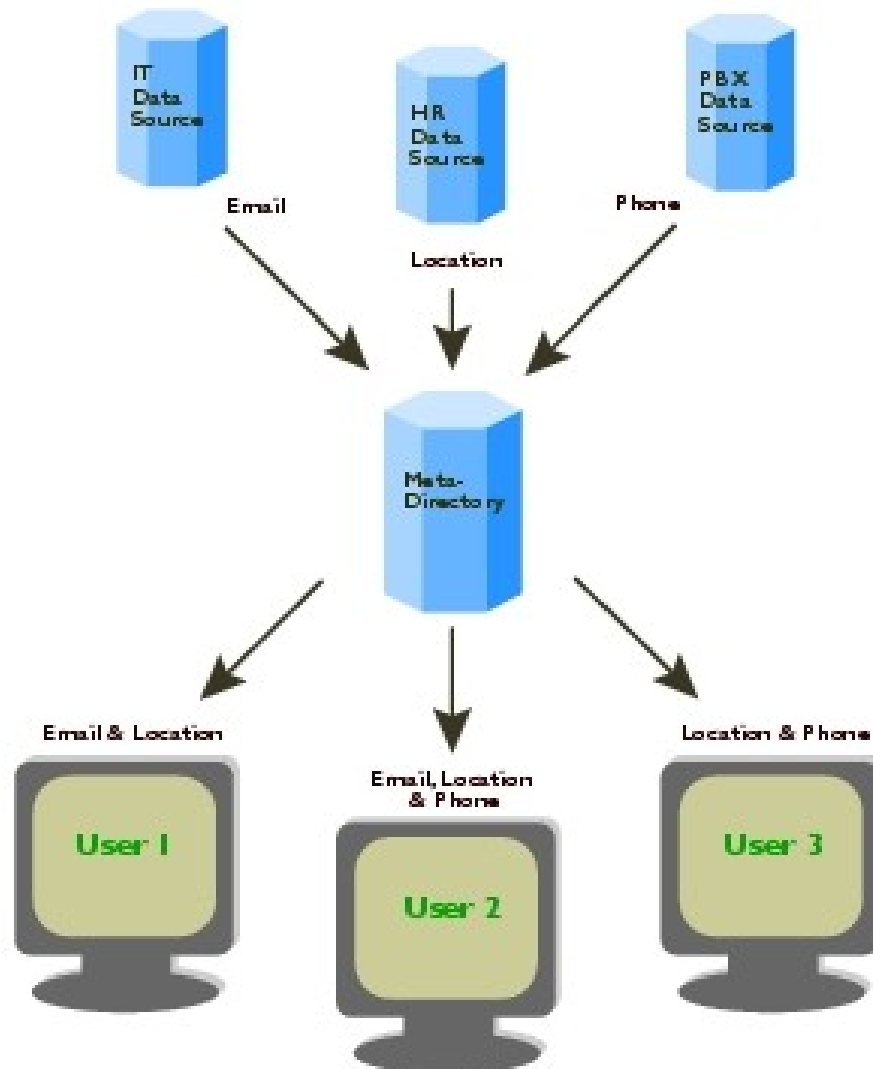
Bilgi kaynakları:

- Oracle RDBMS Server
- M\$ NT domain yapısı
- M\$ Active directory yapısı, ya da
- başka bir Dizin Sunucusu

...Bilgilerin Değişik Kaynakları Olabilir.



...Meta Directory Sunucusu Sonrası,



...Meta Directory Sunucusu Ne Zaman İşimize Yarar ?

- Eski yöntemleri – hemen – terketmek istemiyorsak
- Kontrol edilebilir bilgi kaynağı merkezi yaratmak
- Tek noktadan farklı sunucularda saklanan bilgilere erişmek
- Hızlı bir şekilde uygulama geliştirmek istiyorsak
- Yapılan yatırımın korunması

RDMS mi yoksa Dizin Sunucusu mu?

- Saklanacak bilgi statik mi, dinamik mi?
- Bilginin dağıtık olması gerekiyor mu?
- Bir uygulamadan fazla uygulama aynı bilgiye ihtiyaç duyacak mı?
- Tek attribute da birden çok değer saklanacak mı?
- Uygulama hiyerarşik yapıdan faydalanacak mı?
- Esnek güvenlik güvenlik ayarlarına ihtiyacımız var mı?
- IT altyapınızda tek noktadan belgeleme gerekiyor mu?
- Atanmış bilgi deposu yönetimi isteniyor mu?

Uygulamaları Dizin Yapısı ile Uyumlu Geliştirmek için Nelere Dikkat Etmek Gerekir ?

Dizin sunucuları uygulama seviyesinde protokoller kullanır.

- Kullanıcı belgelemesi yaparken dikkatli olmak:
 - Clear text şifreler kullanılmamalı
 - SASL tabanlı kimlik belgeleme tercih edilmeli
- Dizin sunucusundan bilgileri SSL yada TLS desteği ile almak
- Kullanıcı yetkilendirmelerine dikkat edilmeli.

Dizin Yapısı Bir Kuruma Uygulanırken Nelere Dikkat Edilmelidir ?

- Mutlaka “Yatırımın Geri Dönüşü” (ROI) incelemesi yapılmalı
- Uygulamanın sonuna kadar olan aşamaları içeren bir proje planı hazırlanmalı
- Ölçülebilir problemler adreslenmeli
- Organizasyondaki bütün bölümler projeye dahil edilmeli
- Karmaşık yapılardan kaçınılmalı
- Kurulum sonrası Dizin Yapısının yönetimi dikkatli bir şekilde yapılmalı
- “Dizin Yapısı iyi bir teknolojidir” mantığı altında bir tercih yapılmamalıdır.

Örnek Çalışma:

Üniversitenin Öğrenci Bilgisi Kullanan Birimleri:

- Öğrenci İşleri
- Bilgi İşlem Merkezi
- Kütüphane
- Mediko (Üniversite Hastanesi)
- Öğretim Üyeleri/Öğrenci Adres Defteri
- Telefon Altyapısı Kullanıcı Bilgileri

...Örnek Çalışma:





Sorular ?



Teşekkürler...

Daha fazla bilgi için:
mursel.karabiyik@gantek.com