aselsan
Savunma Sistem Teknolojileri Grubu

YAZILIM ÜRÜN HATTI DEĞİŞKENLİĞİNİN DENETİM ÇEVRİMİ İLE ELE ALINMASI

Emra AŞKAROĞLU
ASELSAN A. Ş.

aselsan
Savunma Sistem Teknolojileri Grubu

İÇERİK

- Yazılım Ürün Hattı nedir?
- Yazılım Ürün Hattı Değişkenliği
  - Ürün Özellik Ağacı Oluşturma
  - Uygulama Örnekleri
- Değişkenliğin Denetim Çevrimi (Inversion of Control) ile Ele Alınması
  - Denetim Çevrimi Yöntemleri
  - SPRING .NET Çatısı ve Kullanımı
  - Uygulama Örnekleri
- Sonuç
- Sorular

YAZILIM ÜRÜN HATTI

- Yazılım ürün hattı, belirli bir pazar kesiminin ya da görevin ihtiyaçlarını karşılamak için, ortak yetenek kümelerini kullanan sistemlerden oluşan yapıya verilen isimdir.
- Yazılım ürün hattı yaklaşımının temelinde, geçmiş yeniden kullanım yaklaşımlarından alınan dersler önemli bir yer tutmaktadır.

aselsan
Savunma Sistem Teknolojileri Grubu

YAZILIM ÜRÜN HATTI DEĞİŞKENLİĞİ

- Yazılım ürün hattı değişkenliği belirlenirken;
  - Yazılım ürün hattından çıkarılması düşünülen ürünlerin neleri içereceği/içerebileceği belirlenmeli
  - Ortaklıklara ve Değişkenliklere göre bu özellikler kategorize edilmeli
  - Ürün özellik ağacı çıkarılmalıdır.

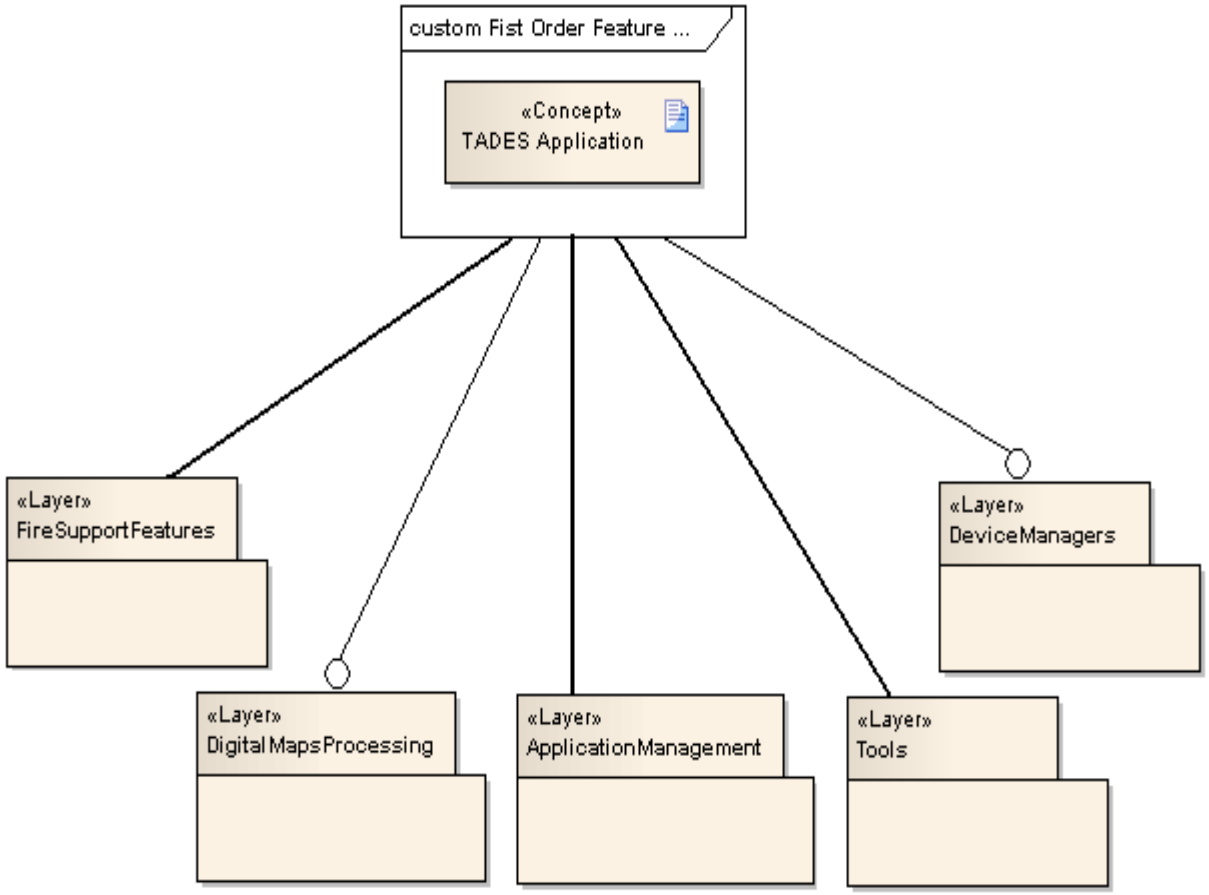
ÖZELLİK TİPLERİ

Özellik Tipi	Anlamı	Kullanılan Şekil
Zorunlu (Mandatory)	Özellik mutlaka olmalıdır.	
Seçenekli (Optional)	Özelliğin kullanımı tüm sistemden bağımsızdır. Kullanımı isteğe bağlıdır.	
Alternatif (Alternative)	Özellik eklendiğinde diğer özellik yerine kullanılabilir	
Birbirini Kapsayan (Mutually Inclusive)	Özelliğin kullanılabilmesi için başka özelliklerin de kullanılıyor olması gerekmektedir.	
Birbirini Dışlayan (Mutually Exclusive)	Özelliğin kullanılabilmesi için başka özelliklerin kullanılmaması gerekmektedir.	

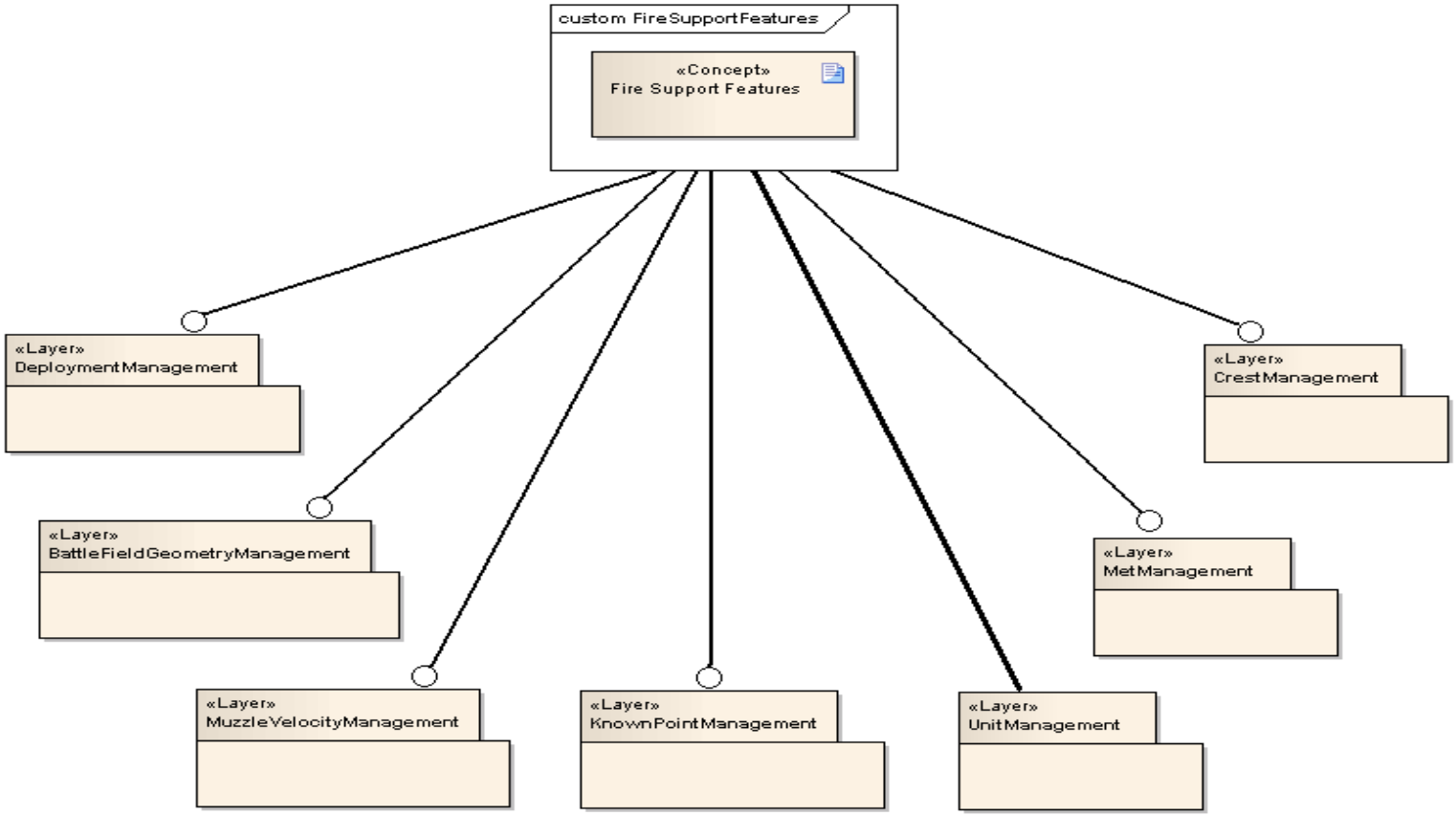
TADES HAKKINDA

- Teknik Ateş Destek Sistemleri(TADES)
  - Komuta kontrol sistemleri için geliştirilmektedir.
  - Yazılım ürün hattı yaklaşımı kullanılmaktadır.

ÜRÜN ÖZELLİK AĞACI

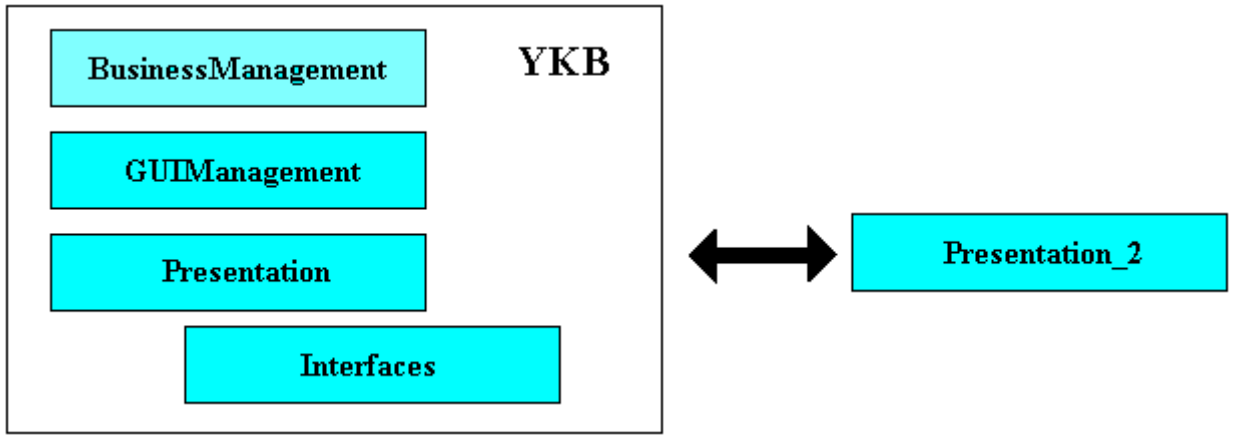


ÜRÜN ÖZELLİK AĞACI - 2



TADES HAKKINDA

- Bileşenler yerine Yazılım Konfigürasyon Birimleri
- Model-View-Controller(MVC) tasarım örüntüsü



Denetim Çevrimi(Inversion of Control)

- Denetim Çevrimi, proje parçaları ya da uygulamalar arasındaki iş akışının ve bağımlılığının yönetilmesini sağlar.
- Böylece;
  - İhtiyaç duyulan nesnenin, kullanılacak nesne tarafından yaratılmadan kullanılmasına olanak sağlar.
  - Yazılım geliştirme esnasında bir yazılımcının diğer YKB'lerin gerçeklenmesini beklemeden, bağımsız bir şekilde kodlama yapabilmesine olanak sağlar.

Denetim Çevrimi Yöntemleri

- Arayüz İletimi (Interface Injection):
  - Bu yöntemde bağımlılıkların iletimi için arayüzler tanımlanır.
- Kurucu İletimi (Setter Injection):
  - Bu yöntemde yapılan şey, yararlanılacak hizmet ve ya kullanılacak bileşen için setter metodlarının yazılması ve yaratımlar için yapılandırma dosyalarının kullanılmasıdır
- Yapıcı İletimi (Constructor Injection):
  - Bu yöntemde sınıf yapılandırıcıları tüm bağımlılıkları içerir. Bağımlı olunan sınıflara ait nesneler bu yapılandırıcının parametreleri olarak kullanılırlar.

SPRING .NET ÇATISI

- SPRING .NET çatısı, nesneye dayalı programlama dillerinde (C#, Java vb.) nesneleri XML tabanlı bir yapı kullanarak bağlamaya yarayan bir araçtır.
- SPRING .NET çatısı açık kaynak kodludur.
- SPRING .NET çatısının temelinde Denetim Çevrimi ve Bağımlılık İletimi(Dependency Injection) bulunmaktadır.

SPRING .NET ÇATISI

- TADES yazılım ürün hattında SPRING .NET çatısı yardımıyla Kurucu İletimi yöntemi kullanılmaktadır.
- Tüm bileşenler, bağımlı oldukları nesneler için Setter metodları tanımlayarak SPRING tarafından yaratılan nesnenin referansını elde etmektedirler.
- TADES yazılım ürün hattından üretilen tüm ürünlerde bileşen bağımlılıklarını ele alan bir yapılandırma dosyası bulunmaktadır.
  - Bu XML yapısındaki yapılandırma dosyası SPRING tarafından kullanılarak bağımlılıklar atanmaktadır.

YAPILANDIRMA DOSYASI KULLANIMI

```
<object id="objBMDeploymentManager"
    type="Tr.Com.Aselsan.TADES.FireSupport.DeploymentManager.BMDep
    loymentManager, DeploymentBusinessManager">
    ...
</object>
```

```
<object id="objBMCoordinateManager"
    type="Tr.Com.Aselsan.TADES.ApplicationFramework.CoordinateOperati
    ons.BMCoordinateManager,  AppFWBusinessManagers">
    ...
</object>
```

YAPILANDIRMA DOSYASI KULLANIMI

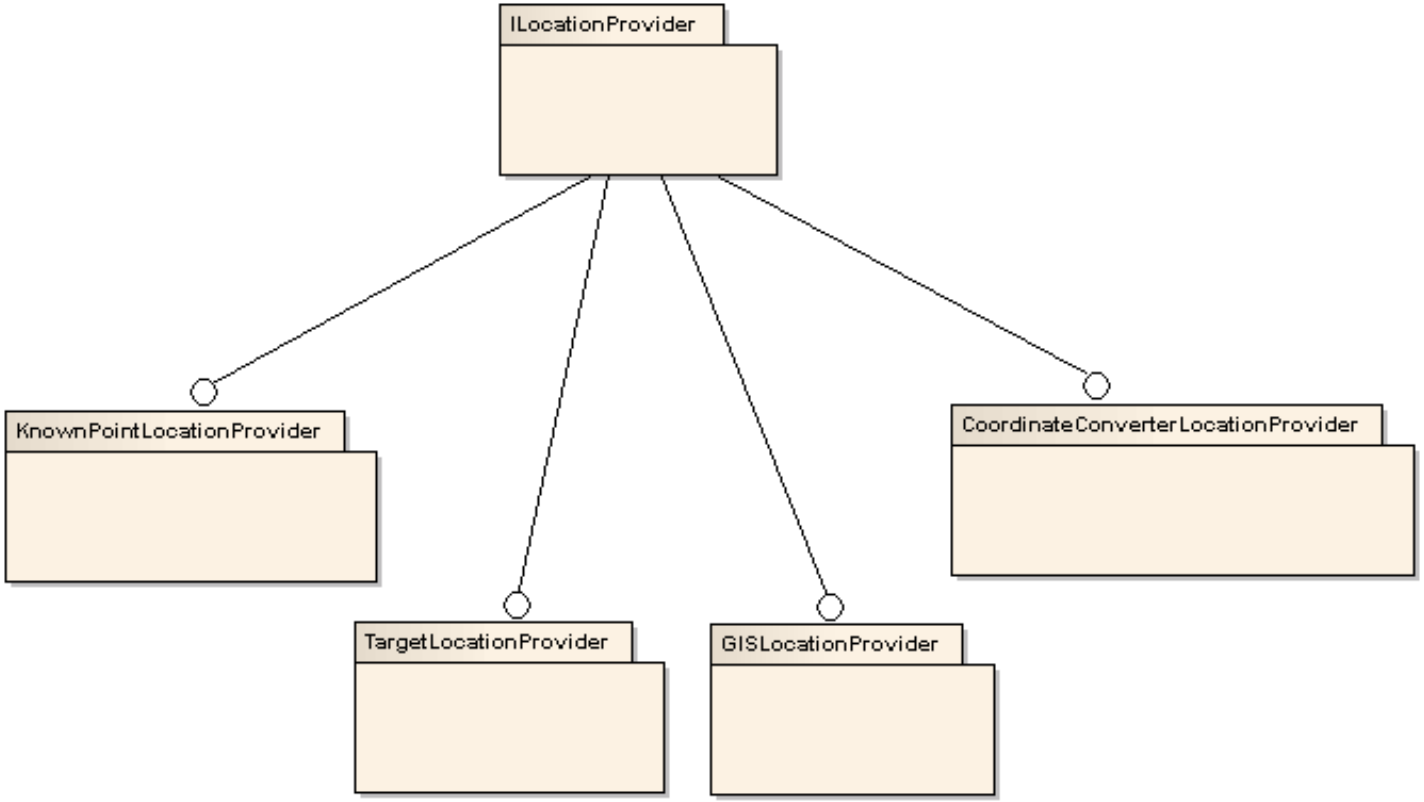
```
<object id="objBMNavigationManager"
    type="Tr.Com.Aselsan.TADES.FireSupport.NavigationManager.BMNavig
    ationManager, NavigationBusinessManager">

    <property name="BMCoordinateManager"
    ref="objBMCoordinateManager" />

    <property name="BMDeploymentManager"
    ref="objBMDeploymentManager" />

</object>
```

YAPILANDIRMA DOSYASI KULLANIMI



YAPILANDIRMA DOSYASI KULLANIMI

```
<property name ="LocationProviderList">
     <list element-type=
    "Tr.Com.Aselsan.TADES.Interfaces.ILocationProvider,
    AppFWBMInterfaces">
          <ref local="objGMTargetManager"/>
          <ref local="objGMKnownPointManager"/>
     </list>
</property>
```

aselsan
Savunma Sistem Teknolojileri Grubu

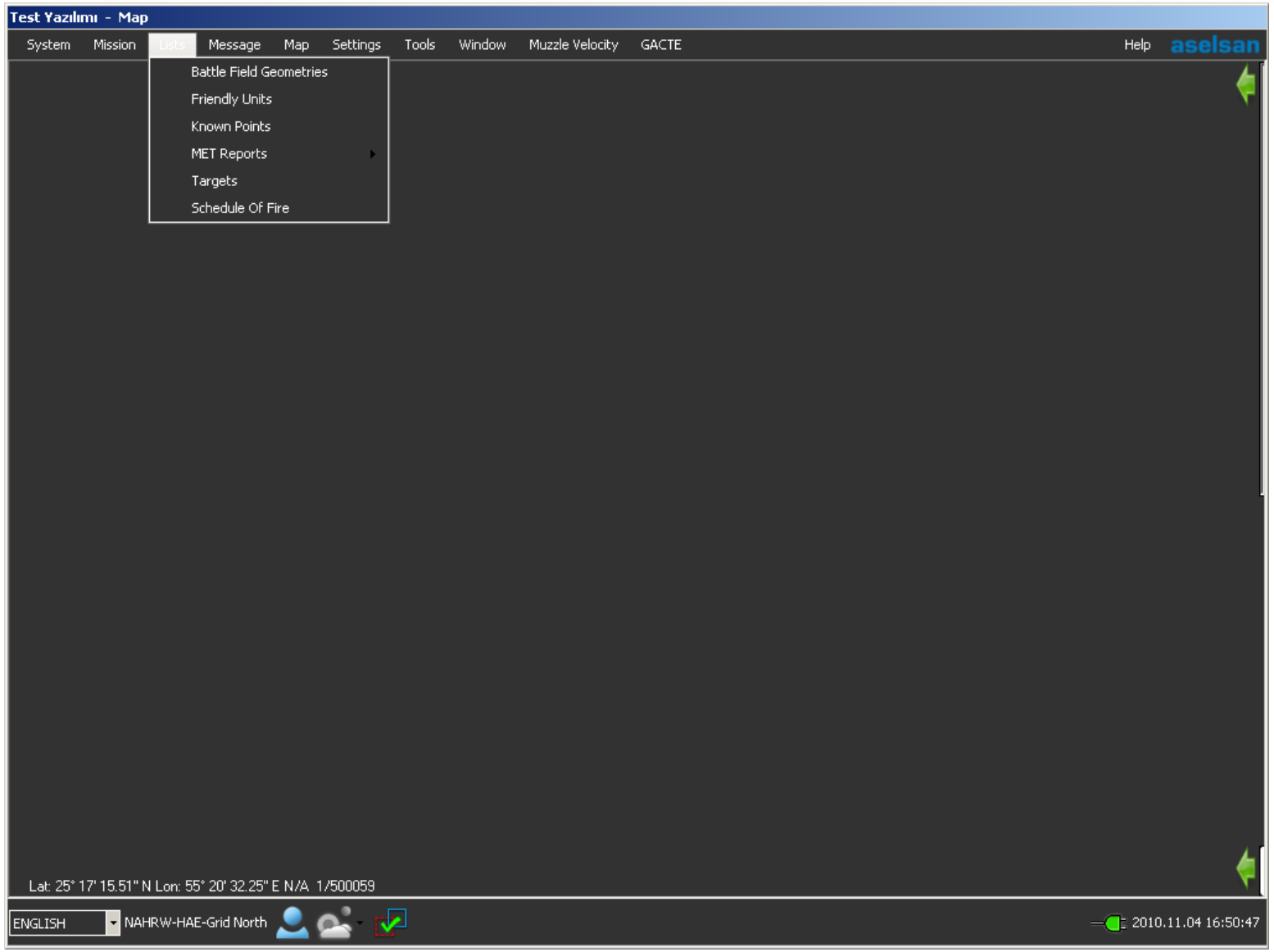
YAPILANDIRMA DOSYASI KULLANIMI

- SPRING .NET çatısı yapılandırma dosyasına göre çalışma zamanında(run-time) nesne ataması yapar.
  - Yeniden derleme ihtiyacı yoktur.
- Yapılandırma dosyası kullanımı ile referans değişiklikleri yapılarak ürünlerde değişkenlikler ele alınabilir.

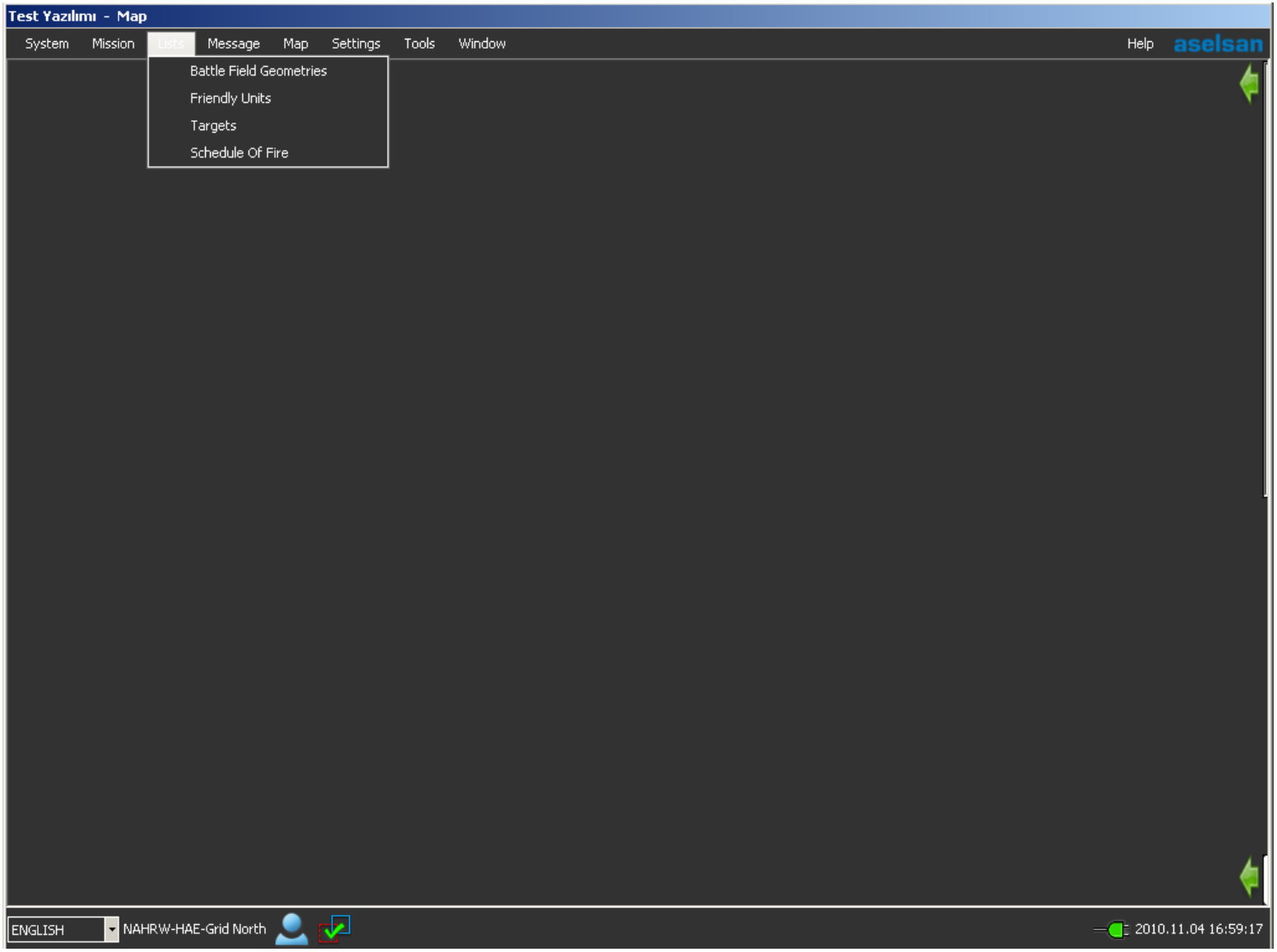
YAPILANDIRMA DOSYASI KULLANIMI

- TADES projesinde, ApplicationFramework YKB'si SPRING .NET çatısının yarattığı nesnelerin referanslarını kullanarak ana uygulamada gösterilecek menü elemanlarını alır.
- Çıkarılacak ürünler için, hangi bileşenlerin içerilmesi isteniyorsa yapılandırma dosyası ona göre ayarlanır.

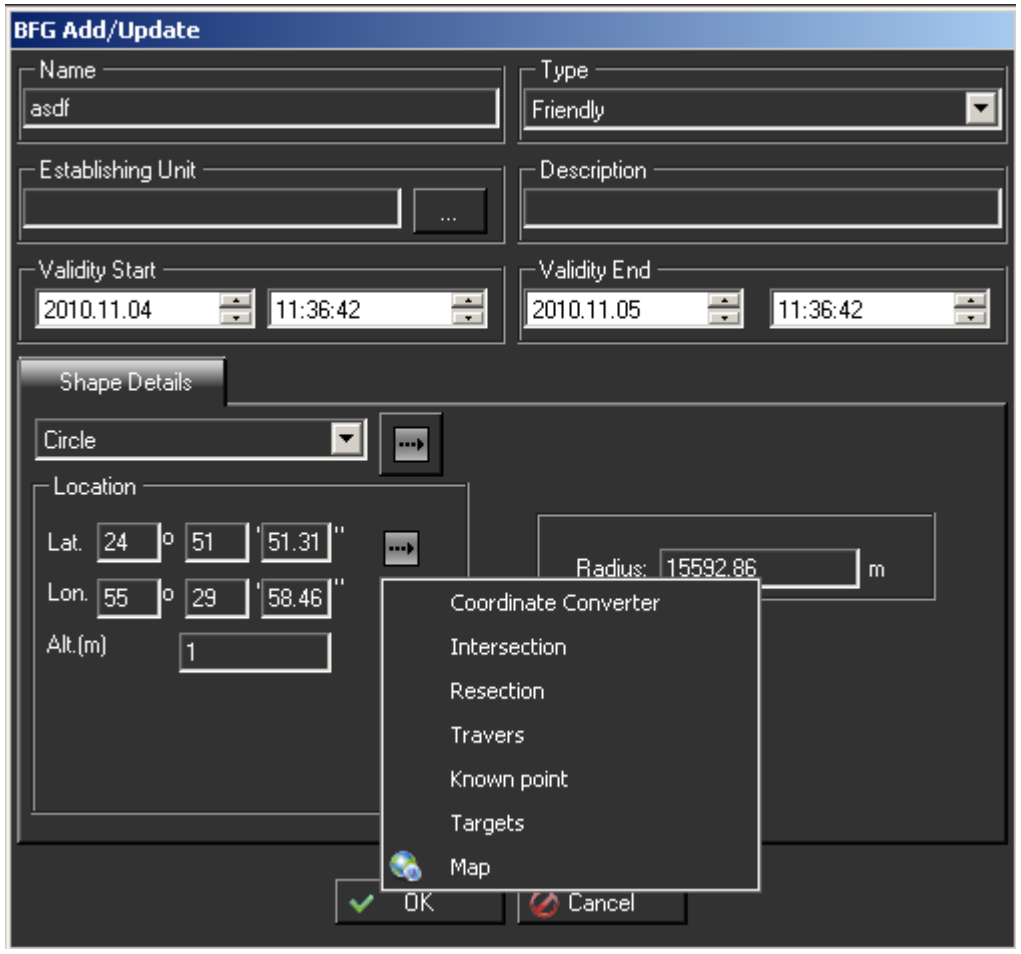
EKRAN GÖRÜNÜMLERİ



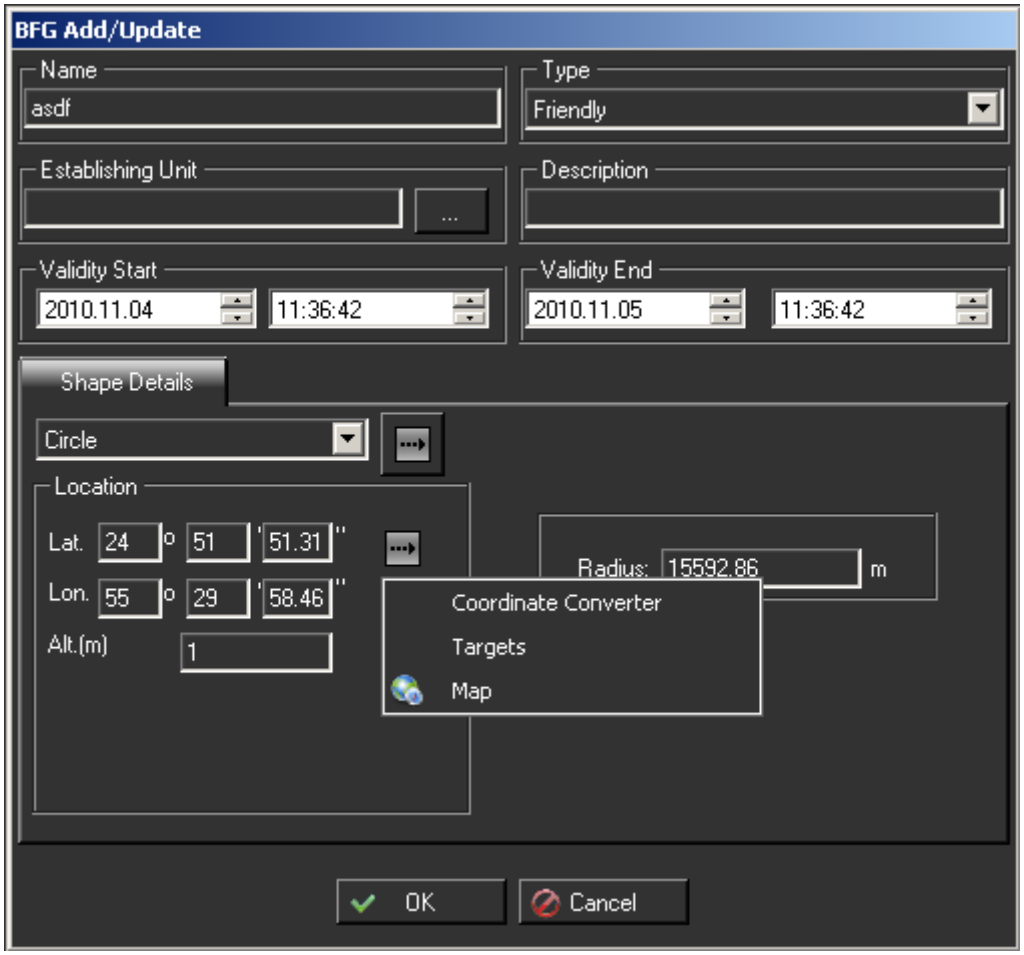
EKRAN GÖRÜNÜMLERİ



EKRAN GÖRÜNÜMLERİ



EKRAN GÖRÜNÜMLERİ



SONUÇ

- Yazılım ürün hattı yaklaşımında
  - Değişkenliğin belirlenmesi
  - Değişkenliğin denetim çevrimi yöntemi ile ele alınması
  - SPRING .NET çatısının kullanımı

- Uygulama örnekleri

aselsan
Savunma Sistem Teknolojileri Grubu

SORULAR

?