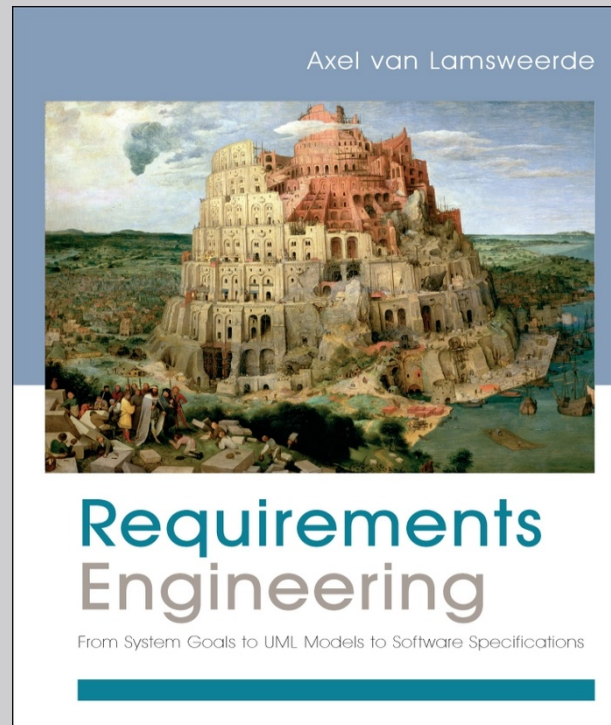
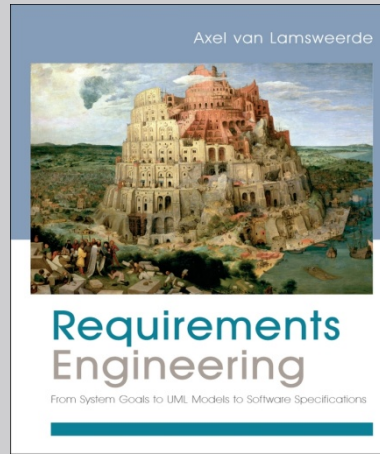


Requirements Engineering

From System Goals
to UML Models
to Software Specifications



Axel Van Lamsweerde



Gereksinimler Mühendisliğinin (GM) Temelleri

Bölüm 1 (devam)

Sistem Gereksinimleri & Yazılım Gereksinimleri

- ◆ Bir yazılım gereksinimi aynı zamanda bir sistem gereksinimidir.
- ◆ Bir sistem gereksinimi aynı zamanda bir yazılım gereksinimi değildir.

Diğer Deyim tipleri

Sistem Gereksinimleri, Yazılım Gereksinimleri

- ◆ **Sistem Gereksinimi**: : Kurallı (*prescriptive*) deyimdir .
Paylaşılması gerek yoktur ve çevre olayına işaret eder.
 - **software-to-be** yapısı ile birlikte diğer sistem bileşenlerini kullanır yürütür.
 - Sözcüksel olarak ifade edildiğinde tüm ayrıntılar verilmiş olur.

TrainMoving → DoorsClosed

- ◆ **Yazılım Gereksinimi** : Kurallı (*prescriptive*) deyimdir. Paylaşılan olaylara işaret eder.
 - Sadece **software-to-be** tarafından yürütülür
 - Yazılı geliştiricilerin ifadeleri ile belirlenir.

measuredSpeed ≠ 0 → doorsState = 'closed'

Diğer Deyim Tipleri

Alan Özelliği & Varsayımlar & Tanımlamalar

- ◆ **Alan Özelliği (Domain property):** dünya fenomeni ile ilgili tanımlayıcı (*descriptive*) deyimdir (herhangi bir software-to-be 'ye bağlı olmaksızın mevcuttur)

$\text{trainAcceleration} > 0 \rightarrow \text{trainSpeed} \neq 0$

- ◆ **Varsayımlar (Assumption):** software-to-be 'nin çevresi tarafından sağlanmış olan deyimdir
 - Çevre olguları (fenomenleri) cinsinden ifade edilir
 - Genellikle kurallı deyimlerdir (sensorler ya da alıcılar (actuators) ile ilgilidir)

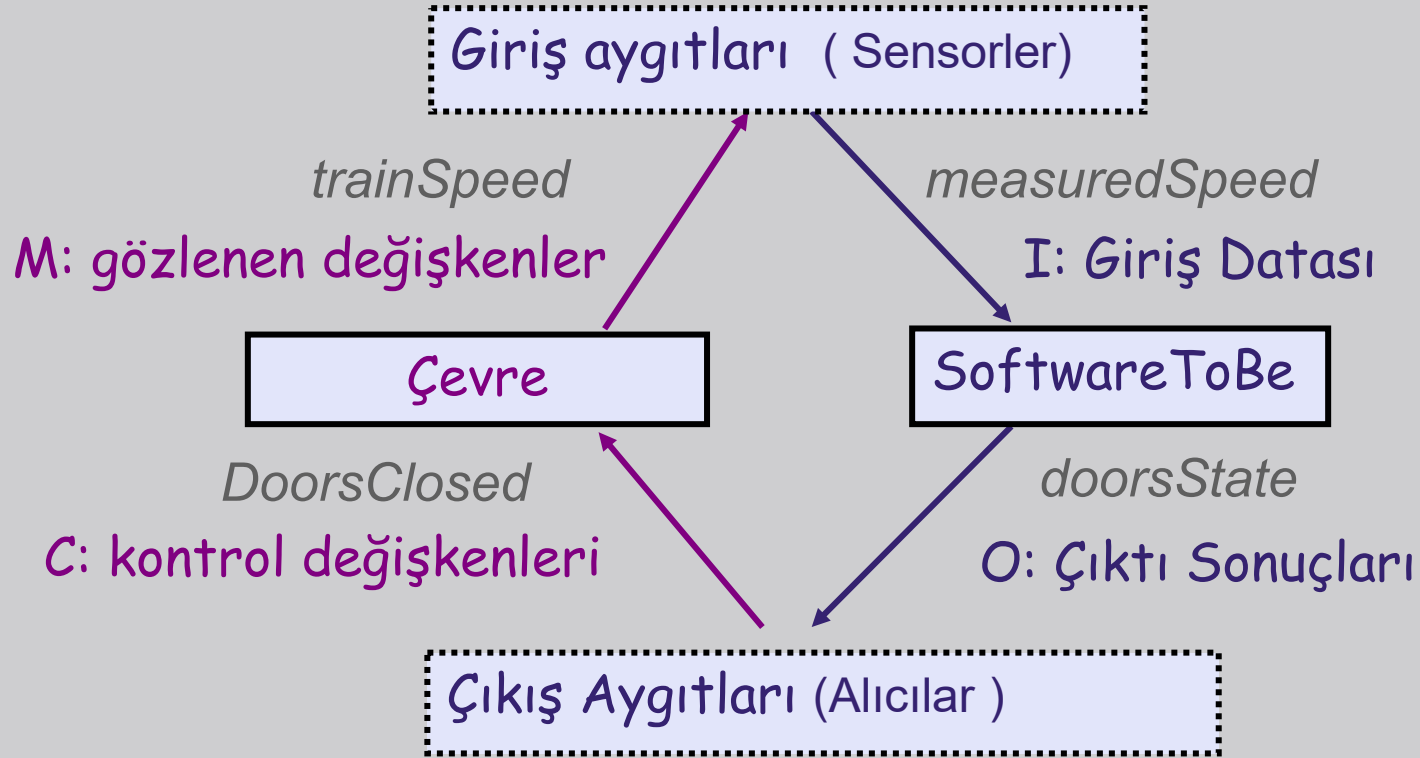
$\text{measuredSpeed} \neq 0 \text{ iff } \text{trainSpeed} \neq 0$

- ◆ **Tanım (Definition):** sistem kavramlarına ya da yardımcı terimlere kesin bir anlam sağlayan deyimdir
 - Doğruluk (truth) değeri yoktur

“measuredSpeed» trenin hız ölçeri tarafından tahmin edilen hızdır.

Yazılım Gereksiniminin Sistem gereksinimi ile ilişkisi :

4-değişken modeli [Parnas95]



$SysReq \subseteq M \times C$ gözlenmiş çevre/kontrol edilmiş değişkenlerin ilişkisi

$SofReq \subseteq I \times O$ relation yazılım giriş/çıkış değişkenlerinin ilişkisi

$SofReq = Map(SysReq, Dom, Asm)$ bağıntısında alan özellikleri ve varsayımları kullanılarak $SysReq$ 'in dönüşümü gerçekleşir

Sistem Gereksinimlerinin Yazılım Gereksinimlerine Dönüşümü farklı argümanlar içerir

SOFREQ, ASM, DOM \models SysReq

“SOFREQ ‘nin içerdiği yazılım gereksinimleri ASM varsayımlarını ve alan özelliklerini içeriyorsa istenilenlerin tümü uygun olarak gerçekleştirilmiştir; böylece sistem gereksinimleri , *SysReq* belirlenmiştir”
(**if-then** cümlesi)

SofReq: $\text{measuredSpeed} \neq 0 \rightarrow \text{doorsState} = \text{'closed'}$

ASM: $\text{measuredSpeed} \neq 0$ iff $\text{trainSpeed} \neq 0$
 $\text{doorsState} = \text{'closed'}$ iff DoorsClosed

Dom: TrainMoving iff $\text{trainSpeed} \neq 0$

SysReq: $\text{TrainMoving} \rightarrow \text{DoorsClosed}$

Gereksinimleri Belirlerken Yapılacaklar

- ◆ Tanımlamalar tamamlandıktan sonra sonra Gereksinimlerin çıkarılması (Elicit)
- ◆ Gereksinimlerin değerlendirilmesi (Evaluate)
- ◆ Gereksinimlerin dokümantasyonu (Document),
- ◆ Uygun Varsayımlar ve Alan Özelliklerinin birleştirilmesi olacaktır.

Gereksinimleri Sınıflandırılması

Fonksiyonel ve Fonksiyonel Olmayan Gereksinimler

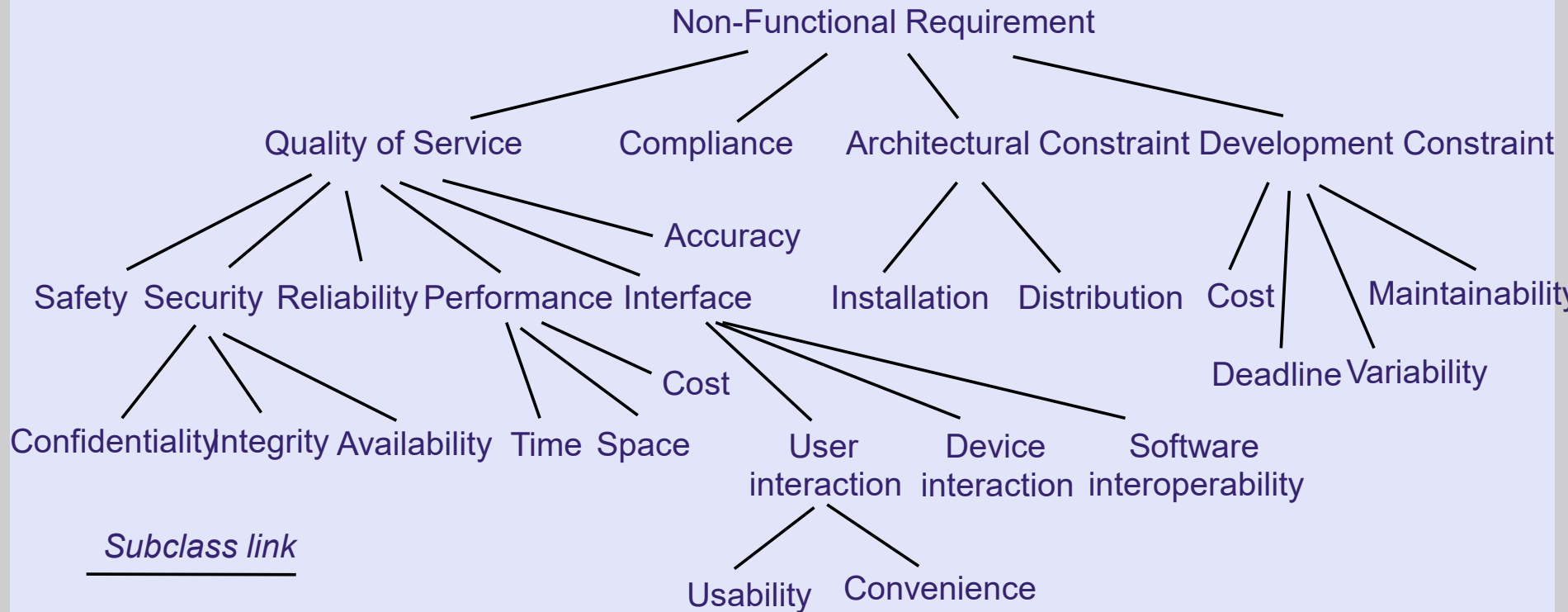
- ◆ **Fonksiyonel Gereksinimler (Functional requirements) :**
software-to-be 'nin hangi hizmetleri (servisleri) sağlayabileceğini (should provide) belirtir.
 - çevre ve uygulanabilirlik koşullarında amaçlanan yazılım etkilerini yakalamak
 - Yazılım operasyonları sonunda birimlerin fonksiyonelliği
- “Yazılım tüm trenlerin hızlanmasını kontrol edecektir. (shall cümlesi) ”

Gereksinimleri Sınıflandırılması

Fonksiyonel ve Fonksiyonel Olmayan Gereksinimler

- ◆ **Fonksiyonel Olmayan gereksinimler (Non-functional requirements):** Hizmetlerin (servislerin) nasıl sağlanabileceğine (should be provided) ilişkin kısıtlardır (constraints)
 - **Nitelik Gereksinimleri: (Quality Requirements):** güvenlik (safety and security, doğruluk (accuracy), zaman/uzam performansı (time/space performance), kullanılabilirlik (usability), ...
 - Diğerleri : Uygunluk (compliance), mimari gereksinimleri (architectural), geliştirme gereksinimleri (development reqs)
 - Sistemin terimleri cinsinden kesin yapılacaktır.
 - “Hız komutları her trn için 3 saniyede bir verilecektir”

Fonksiyonel Olmayan Gereksinimlerin Sınıflandırması



Fonksiyonel / Fonksiyonel olmayan Gereksinimlerin Örtüşmesi

- ◆ Fonksiyonel ve/veya fonksiyonel olmayan gereksinimleri arasında kesin bir ayırım sınırı yoktur; bazı durumlarda örtüşmeleri mümkündür
- ◆ Functional/non-functional: Güvenlik duvarı için **fonksiyonel gereksinim «security»** ile ilgilidir.
- ◆ Non-functional overlaps: "Tren komutlarının yüksek frekansı" ise **performans ve «safety»** ile ilgilidir.

Gereksinimlerin Sınıflandırması -1

Fonksiyonel Gereksinimler

- ◆ **Fonksiyonel Gereksinimler (Functional Requirements) :** software-to-be 'nin hangi hizmetleri (servisleri) sağlayabileceğini belirtir.
 - Çevre (environment) ve uygulanabilirlik koşullarında amaçlanan yazılım etkilerini yakalamak
 - Yazılım operasyonları sonunda birimlerin fonksiyonelliği
- “Yazılım tüm trenlerin hızlanmasını kontrol edecektir.”

Gereksinimlerin Sınıflandırması- 2

Fonksiyonel Olmayan Gereksinimler

- ◆ **Fonksiyonel Olmayan gereksinimler (Non-functional requirements):** Hizmetlerin nasıl sağlanabileceğine ilişkin **kısıtlardır** (constraints).
 - **Nitelik Gereksinimleri: (Quality Requirements):**
 - Güvenlik ve güvenilirlik (safety and security),
 - Doğruluk (accuracy),
 - Zaman/uzam performansı (time/space performance),
 - Kullanılabilirlik (usability), ...
 - **Diğer fonksiyonel olmayan gereksinimler :**
 - Uygunluk (compliance),
 - Mimari Gereksinimleri (architectural reqs.),
 - Geliştirme gereksinimleri (development reqs)
 - **Sistemin terimleri cinsinden kesin olarak gerçekleşecektir.**

“Hız komutları her tren için 3 saniyede bir verilecektir”

«Security and Safety» Fonksiyonel Olmayan Gereksinimleri

- Güvenliğin sağlanması durumu (**security assurance case**) belirlenmiş bir dizi kriterin gerçekleşmesini gerektirir. Böylece sistem kendi güvenlik özelliklerine göre önceden belirlenmiş spesifik beyanları sağlayacaktır (ISO 15026)
- Kanıtların, argümanların ve varsayımların biçimsel olarak betimlenmesidir. Böylece bir sistem ya da bir ürün güvenlik gereksinimlerini (**safety requirements**) sağlayacaktır (Yellow Book)

«Security and Safety» Fonksiyonel Olmayan Gereksinimleri

- ◆ **Safety Requirements:** Nitelik (Quality) gereksinimleridir.

Olumsuz durumlarla sonuçlanabilecek (might) (kaza, bir parçanın aşınarak bozulması , çevreye zarar gibi) faktörlerin yazılıma etkisine ait bildirimlerdir.

Trenlerin kontrol edilebilen hızları en kötü koşulda durma mesafesini (worst-case stopping distance) ard arda giden trenler arasında her zaman koruyacaktır.

- ◆ **Security Requirement :** Nitelik (Quality) gereksinimleridir. Sisteme ait nesnelerin istenmeyen çevre koşullarına göre korunması ile ilgili kuralları belirler (prescribe)

Örnekleri farklı kategorilerde verilir.

Güvenlik (Security) Gereksinimlerinin Ayrıntılı Örnekleri

- ◆ **Gizlilik Gereksinimleri (Confidentially req.)** Yetkisi olmayanların hiçbir şekilde erişemeyeceği bazı kritik bilgilerin olması durumudur.

Hangi kitabın kimin tarafından alındığını personel dışında sistemi kullanan hiç kimse bilemez .

- ◆ **Gizlilik Gereksinimleri (Privacy Req.)** Bazı özel bilgiler o bilginin sahibinin onayı olmadan açılmaz.

Herhangi bir katılımcının günlük kısıtları kendisinin onayı olmadan diğer katılımcılar tarafından görüntülenemez.

Güvenlik (Security) Gereksinimlerinin ayrıntılı örnekleri

◆ Bütünlük (t ümlük, tamamiyet) Gereksinimleri (Integrity Req.)

Bazı bilgiler ancak doğru yapıldığında ve izin ile (authorization) değiştirilebilir.

Kitap kopyalarının iadesi doğru olarak ve yalnızca kütüphane personeli tarafından kodlanacaktır.

◆ Geçerlik gereksinimleri (Availability Req.) Bazı bilgiler ve kaynaklar gerekli olduğu zamanda ve durumda ve kullanımına izin verildiğinde (authorized) kullanılabilir

Trenin bulunduğu pozisyonlar ile ilgili bilgiye acil durum istasyon bilgisayarında her zaman ulaşılacaktır .

Güvenilirlik Gereksinimleri (Reliability Requirements) fonksiyonel olmayan gereksinim

- ◆ Yazılımın işlevliğinin (güncel olarak kullanılabilmesinin) uzun süre devam etmesi beklenir. Yazılımın gerçekleştirecekleri de, bazı istisnaların olması olasılığına rağmen, **doğru (correct) ve güçlü (robust)** olarak sağlanmalıdır.

Tren sevkیات kontrol yazılımının *güncel olarak (işlevsel olarak)* kullanılabileceği ortalama bir süre arızalar da göz önüne alındığında 10^9 saattir.

Doğruluk Gereksinimleri (Accuracy req.)

- ◆ Nitelik gereksinimi olarak yazılım tarafından işlenen bilginin (information) durumunu kısıtlar. Böylece ortamdaki fiziksel bilgiye ait durumun doğru olarak aktarımı gerçekleşir.

1.Tren kontrol birimi tarafından kullanılan tren pozisyonları hakkında bilgi, en çok X metreye kadar olan trenlerin gerçek konumunu doğru olarak yansıtacaktır.

2.Kitabın bir kopyasının alınması , yazılım sistemi tarafından sadece ve sadece kitap kütüphanede (raflarda) mevcutsa mümkün olacaktır.

3.Toplantı çizelgesinde kullanılan kısıtlamalar davet edilen katılımcıların gerçek kısıtlamalarını doğru bir şekilde yansıtmalıdır

Performans Gereksinimleri

- ◆ Nitelik gereksinimleri olarak yazılımın operasyonel koşullarındaki kısıtlarını , zaman ya da mekan (time or space) bakış açısından belirler. Bunlar

aktivitelerin sıklığı (periyodu),

iş çıkarma yeteneği yani verim (throughput) ,

giriş / çıkışların büyüklüğü

olabilir.

1. Her bir trene her 3 saniyede ivme (acceleration) talimatı verilir.

2. Toplantı düzenleyen kişi paralel olarak 3 oturum gerçekleştirebilir.

3. Özgeçmiş sorgularının dönüşü 2 saniyeden az olacaktır.