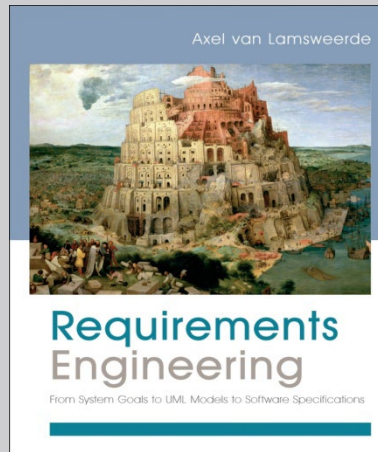




Building System Models for RE

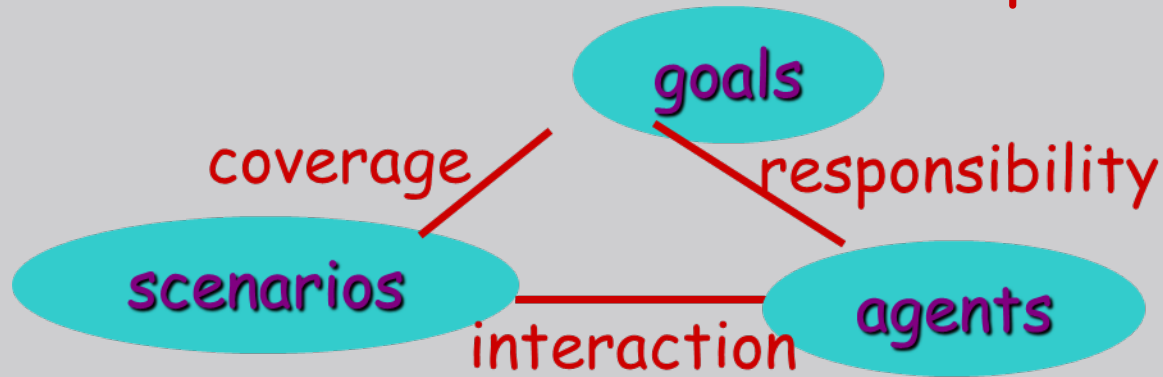
Modeling System Objectives with Goal Diagrams

Amaca Yönelik Gereksinim Tanımlamalarında Kavramların Farkı

- ◆ Amaca yönelik olarak gereksinimlerin tanımlanması çözümü yukarıdan aşağıya doğru bir ağaç olarak kurmaz. Aksine amacın soyutlamaları (betimlemeleri) aşağıdan yukarı doğru tanımlanarak sonuçta kök hedefe (amaca) ulaşılır.
- ◆ Goal-oriented $\neq$ top-down (bottom -up specification)
- ◆ Amaca yönelik olarak gereksinimlerin betimlenmesi yeni bir kavram **agent** sözcüğünü kullanır ve gereksinimbildirimlerinin çözümlemeleri **agent - oriented** model olarak adlandırılır.
- ◆ Goal-oriented $\Rightarrow$ agent-oriented,
scenario-oriented

Gereksinim Mühendisliğinde farklı bir Yaklaşım

Amaca -Yönelik Gereksinimler Spesifikasyonu



Amaca Yönelik RE tüm çözüm aşamalarında **amaçları** kullanır. Amaç, **prescriptive** bir ifadedir.

- ◆ Sistem amacını, agent'ların ilişkileri /birlikteliği sağlar.
- ◆ İfade çevresel faktörler (environmental phenomena) ile birlikte düzenlenir.
- ◆ Agent, amacın betimlenmesinde önemli bir rol üstlenir.
- ◆ Sistemin uygunluğu için **agent** davranışlarını **kısıtlamalıdır**.

Amaca Yönelik Tasarım

- ◆ Amaçlar yüksek düzeyli, stratejik, kaba tanecikli (coarse grained) nesnelerdir.

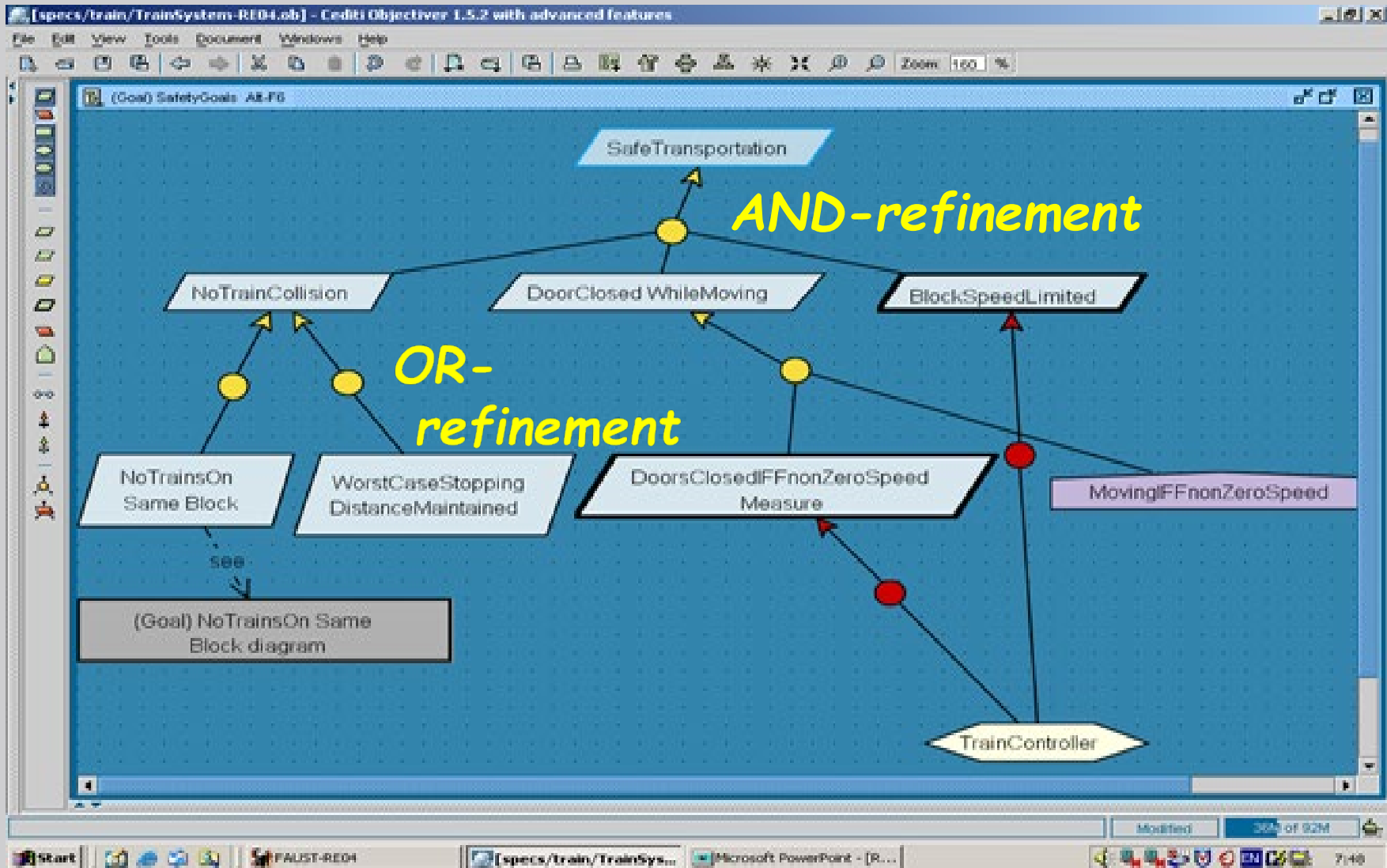
The system's transportation capacity shall be increased by 50%.

- ◆ Daha sonra sistemin daha düşük düzeydeki ince tanecikli (fine-grained) prescriptive önermeleri tasarlanır.

Acceleration commands shall be send to every train every 3 seconds

- ◆ İnce tanecikli (fine-grained) amaçlar için daha az sayıda «agent» ların sağlanması gerekeceği sonucu çıkarılır. Çünkü çözümleme detaylandırılmıştır.
- ◆ Bu nedenle de amaca yönelik RE aşağıdan yukarıya doğru gerçekleştirilir.
- ◆ Çözüm yolu , aşağıdan itibaren çözümü ince ince işleyerek amaca ulaşmaktır.

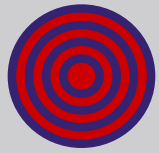
GOALS



«Why» sorusu cevaplanır

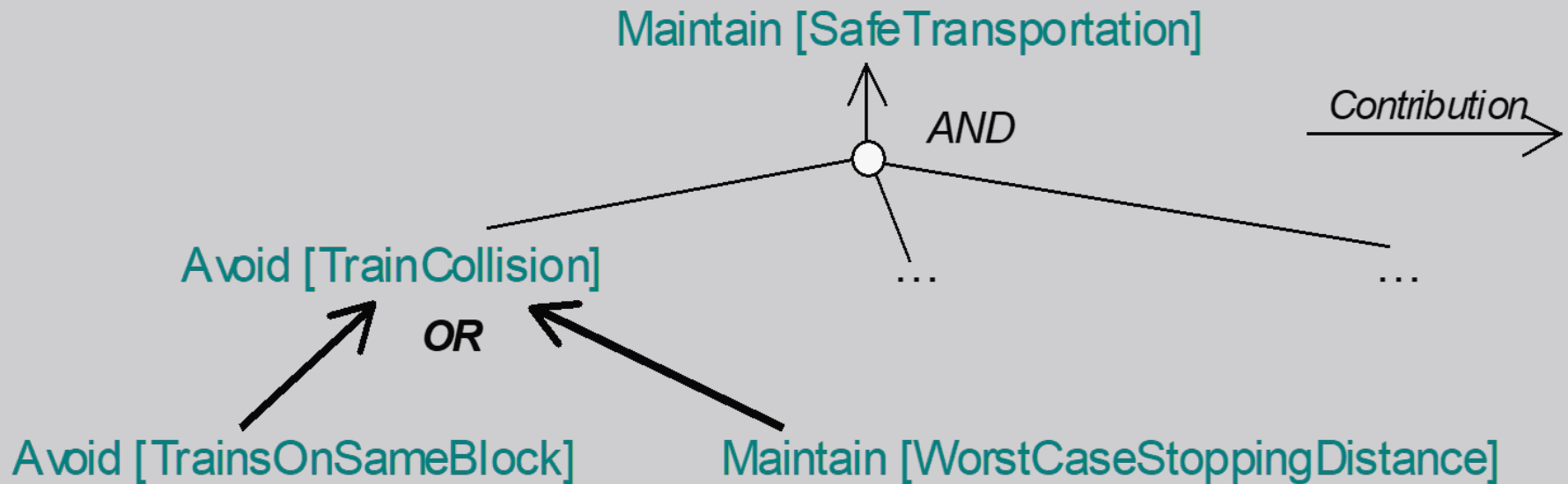
Amaç Modeli ve Amaç Diyagramları

- ◆ Amaç modeli sistemin hem fonksiyonel hem de fonksiyonel olmayan amaçlarını yazılım gereksinimleri ve çevresel varsayımlar/kabuller(environmental assumptions) ile birlikte yapar.
- ◆ Çözüm için alternatif yollar ise bazı amaçların yeniden düzenlemesi /elenmesi ve çatışma/ uyuşmazlık (conflict) durumlarının da ele alınması demektir.
- ◆ Sistem modelinin oluşturulması **grafik olarak** düzenlendiğinde ise **AND/OR grafi** olarak adlandırılır.
- ◆ Böylece gereksinimler mühendisliğinde sistem modellerinin tasarımı gerçekleştirilir.
- ◆ Model oluşturulması için amaca yönelik bir yaklaşım da WHY-WHAT-WHO boyutları ile incelenecektir.

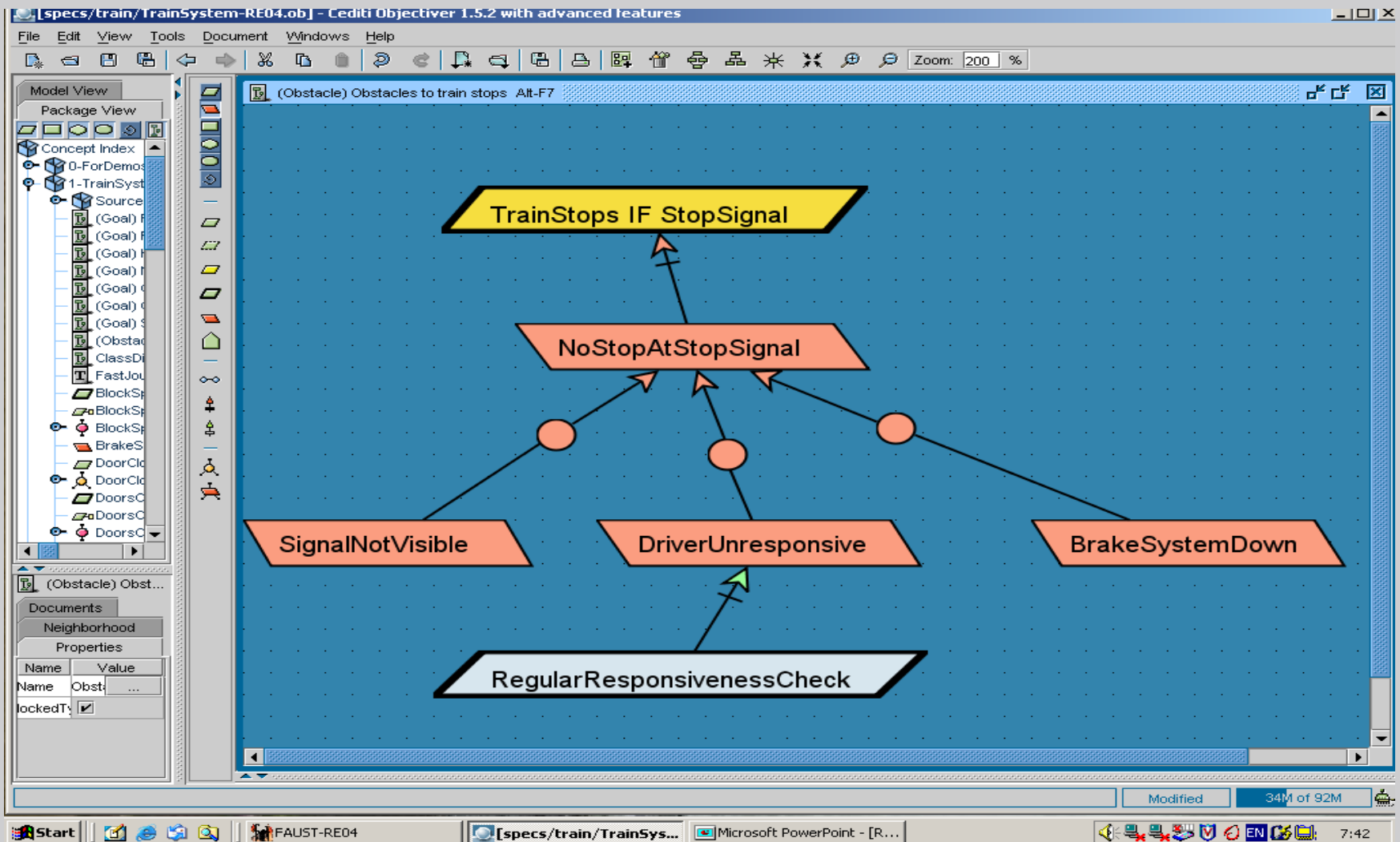


Amaçların Gereksinimler Mühendisliğindeki Rolü

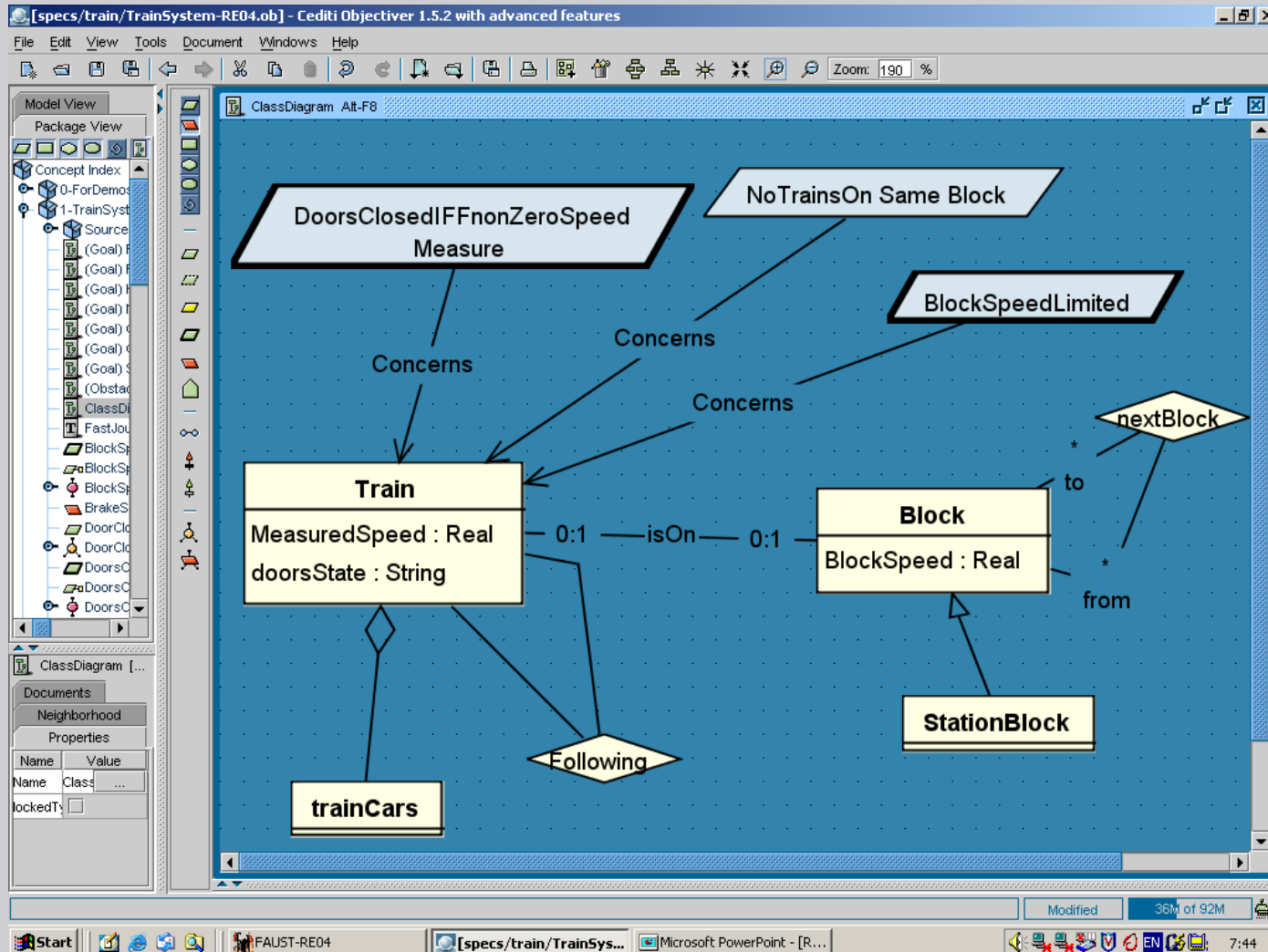
- ◆ Goal **OR**-refinement → capture of alternative options



RISKS

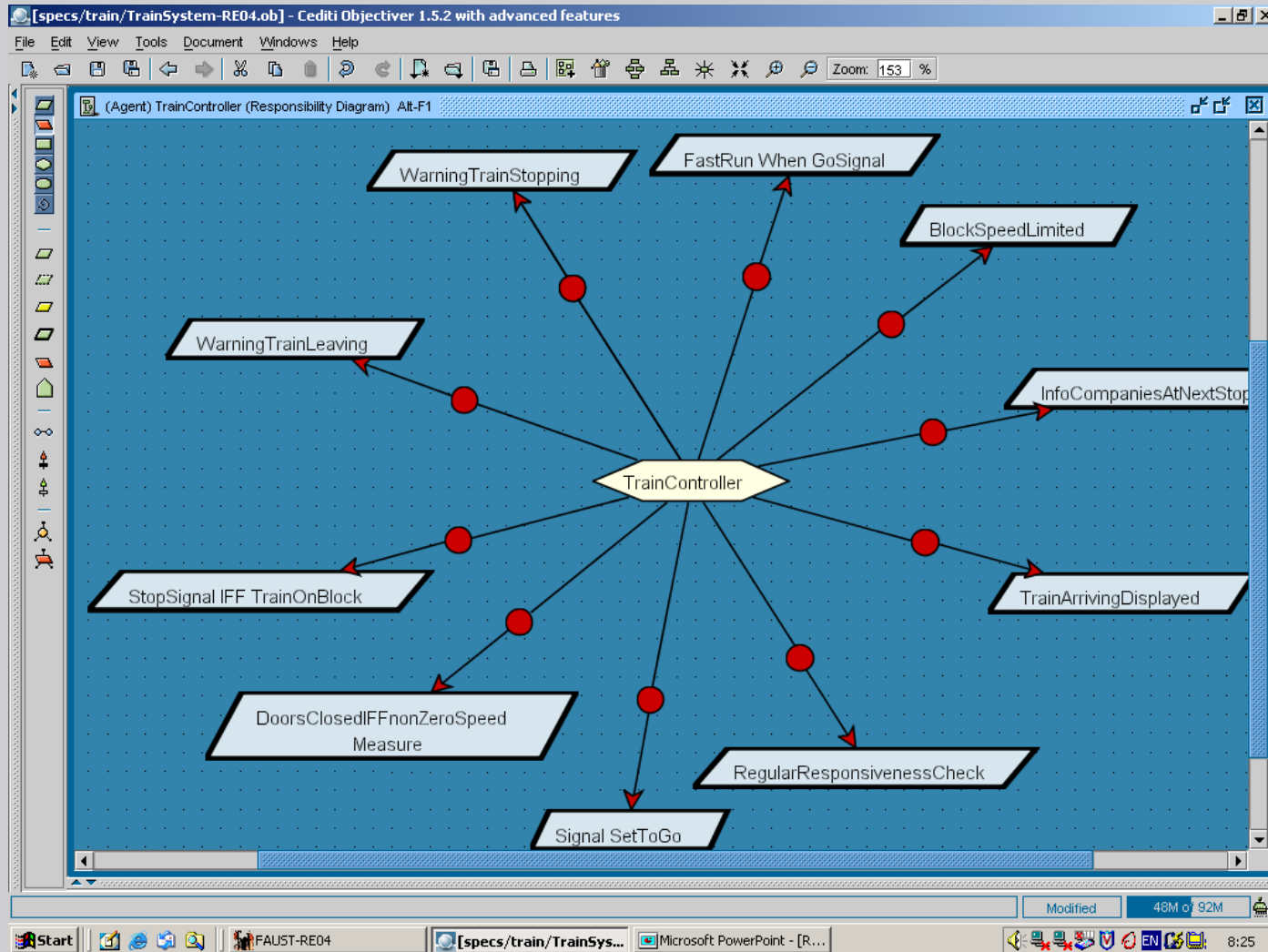


CONCEPTUAL OBJECTS



«on what» sorusu cevaplanır

AGENTS



«who» sorusu cevaplanır

«Agent» Nedir?

- ◆ Agent, aktif bir sistem bileşenidir. Amacın gerçekleştirilmesinde aktif rolü vardır.
- ◆ Amacın başarı ile gerçekleştirilmesi sistem «agent» larının işbirliği ile gerçekleşecektir.
- ◆ Bu nedenle amaç agent lar arasında gerçekleşen ortak (paylaşılmış) olgulara/olaylara (phenomana) göre tanımlanmalıdır.
- ◆ Sistem «agent» ları sistemin hedefini (scope of goal) belirler.

Amaç odaklı Sistem Tasarımında Agent Tipleri

- ◆ İnsan (human) «agent» lar organizasyonel birimler, operatörler, son kullanıcılar gibi belirli roller üstlenir.
- ◆ Aygıtlar (devices) sensörler, alıcılar (actuators), ölçüm cihazları, iletişim ortamı (communication media) olabilir.
- ◆ Mevcut sistemin yazılımına ait bileşenler, software-to-be yi oluşturacak yeni yazılım bileşenleri «agent» tipleridir.



Package View Model View

1-TrainSystemModel

- SourceDocuments
 - (Agent) TrainController (Responsibility Diagram)
 - (Agent) TrainDriver (Responsibility Diagram)
 - (Goal) FastJourney diagram
 - (Goal) FunctionalGoals
 - (Goal) High-level goals
 - (Goal) NoTrainsOn Same Block diagram
 - (Goal) Operationalization
 - (Goal) QoS-Goals
 - (Goal) SafetyGoals
 - (Obstacle) Obstacles to train stops
 - ClassDiagram
 - FastJourney diagram (Text Explanation)
 - BlockSpeedLimited
 - BlockSpeedLimited "Concerns" Train
 - BlockSpeedLimited "Responsibility" TrainContro
 - BrakeSystemDown
 - DoorClosed WhileMoving
 - DoorClosed WhileMoving "Refinement" DoorsCl
 - DoorsClosedIFFnonZeroSpeedMeasure
 - DoorsClosedIFFnonZeroSpeedMeasure "Conce

NoTrainsOn Same Block [Goal]

Properties Neighborhood Documents

Name	Value
Name	NoTrainsOn Same Block
Def	Each block may contain at most
Issue	
Pattern	Avoid
Category	Safety
Priority	High
FormalDef	

(Goal) NoTrainsOn Same Block diagram Alt-F1

(Goal) SafetyGoals

see

NoTrainsOn Same Block

TrainStops IF StopSignal

TrainDriver

Def of NoTrainsOn Same Block

Each block may contain at most one train at any point in time

Close

TrainCo

Modified

34M of 89M

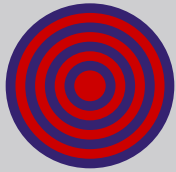


Goal categories

Amaçların Sınıflandırılması



•



Amaçların Gereksinimler Mühendisliğindeki Rolü

- ◆ $Req, Expectation, Domain \models G$,
- ◆ $SubG, Expectations, Domain \models G$

“Alan özellikleri (domain properties) bağlamında *Dom*, gereksinimler/alt amaçlar *Req/SubG* olarak ifade edilir. ve bunlar G is amacını *Exp* (expectational) koşulu ile sağlar

R: doorsState = ‘closed’ **if** measuredSpeed $\neq 0$

E: Doors are closed **iff** doorsState = ‘closed’ ($\leftrightarrow$ door actuators)
measuredSpeed = physicalSpeed ($\leftrightarrow$ speedometer)

D: Train is moving **iff** physicalSpeed $\neq 0$

G: Doors are closed **if** train is moving

- ◆ Önceki slaytta gereksinimler Amaç Odaklı tanımlandığında bunun bir betimleniş biçimi örnek olarak verildi
- ◆ Sonraki slaytta ise sistem gereksinimleri ile yazılım gereksinimlerinin ifadesini ve birbirlerinden farklarının açık olarak gösterimi genel gereksinim tanımlamalarından çıkarılmaktadır.
- ◆ Her iki gereksinim bildirimi farklı bir bakış açısından kaynaklanmış aynı hedefe ulaşmayı gösterir.

Sistem Gereksinimlerinin Yazılım Gereksinimlerine Dönüşümü farklı argümanlar içerir

$$\text{SOFREQ, ASM, DOM} \models \text{SysReq}$$

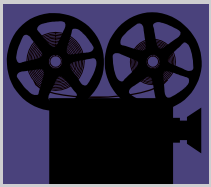
“SOFREQ ‘nin içerdiği yazılım gereksinimleri ASM varsayımlarını ve alan özelliklerini içeriyorsa istenilenlerin tümü uygun olarak gerçekleştirilmiştir; böylece sistem gereksinimleri , *SysReq* belirlenmiştir” (if-then cümlesi)

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

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

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

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

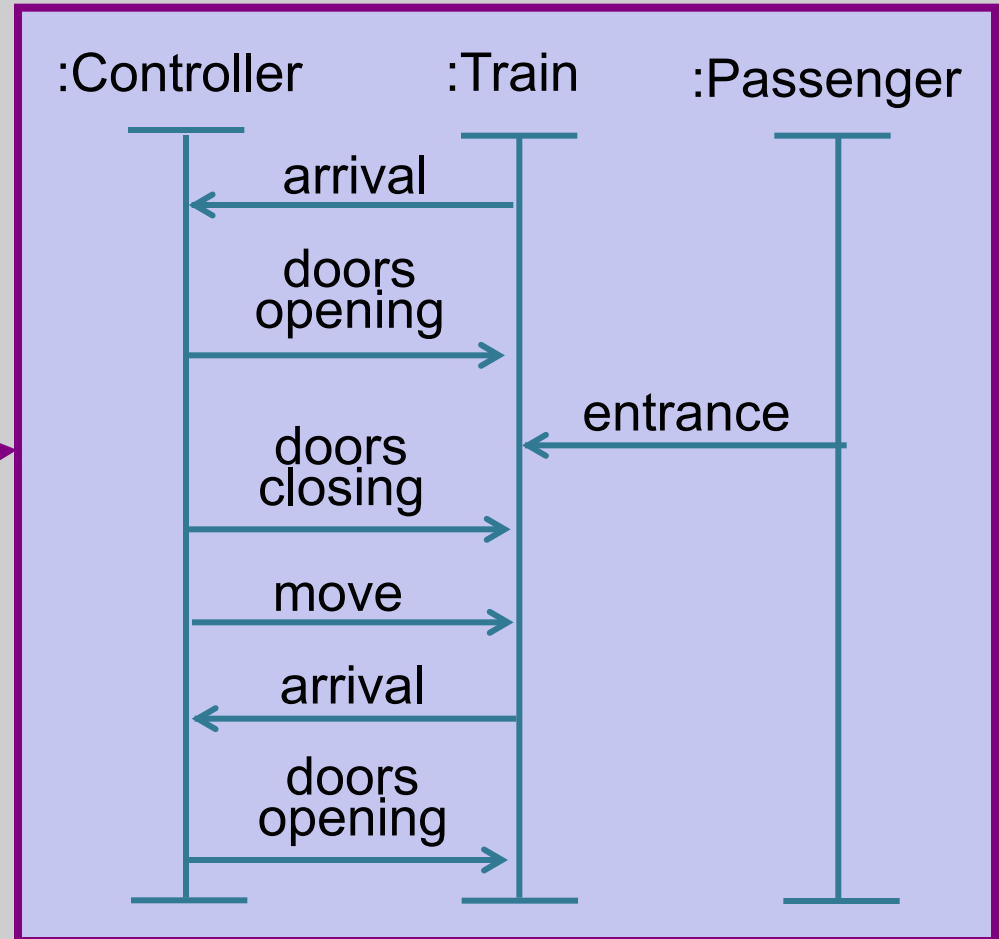


Senaryolar Somut Simgelenişlerdir ve Amacın Belirlenmesini (Anlaşılmasını) sağlarlar (*elicitation/validation*)

easy to get from or validate with stakeholders

DoorsClosed
WhileMoving

G covers Sc:
Sc is subhistory in set of behaviors prescribed by G





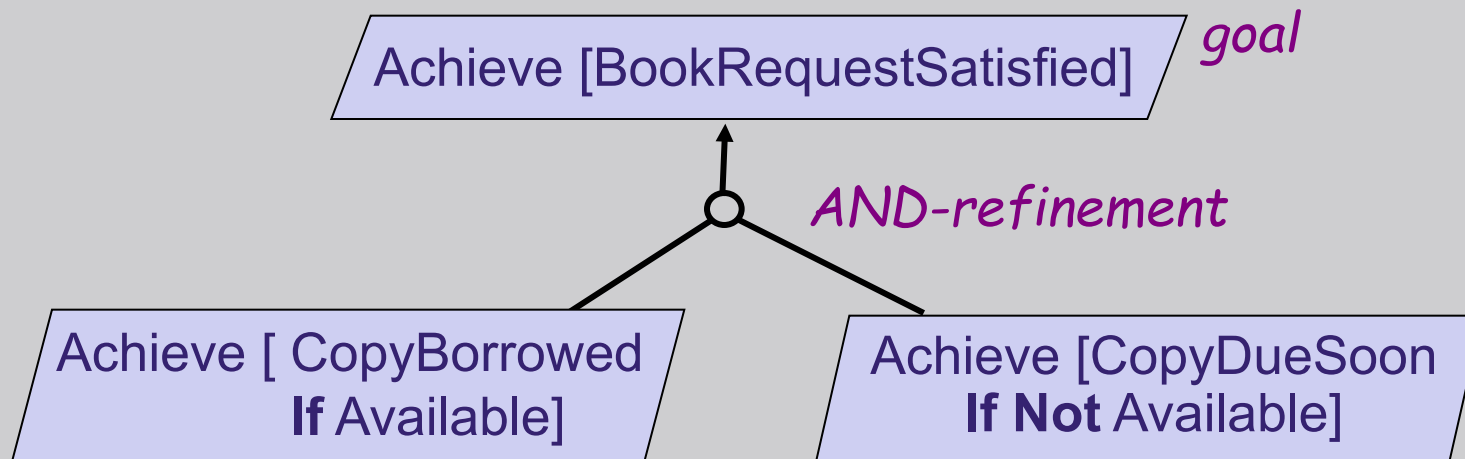
Goal refinement

Amacın Ayrıntılandırılması

- ◆ An **AND-refinement** of goal G into subgoals $G_1, \dots, G_n$ states that G can be satisfied by satisfying $G_1, \dots, G_n$

The set $\{G_1, \dots, G_n\}$ is called **refinement** of G

Subgoal G_i is said to **contribute positively** to G



Def *In case a requested book has no copy available for check out, a copy of that book shall be made available within 2 weeks for check out by the requesting patron.*

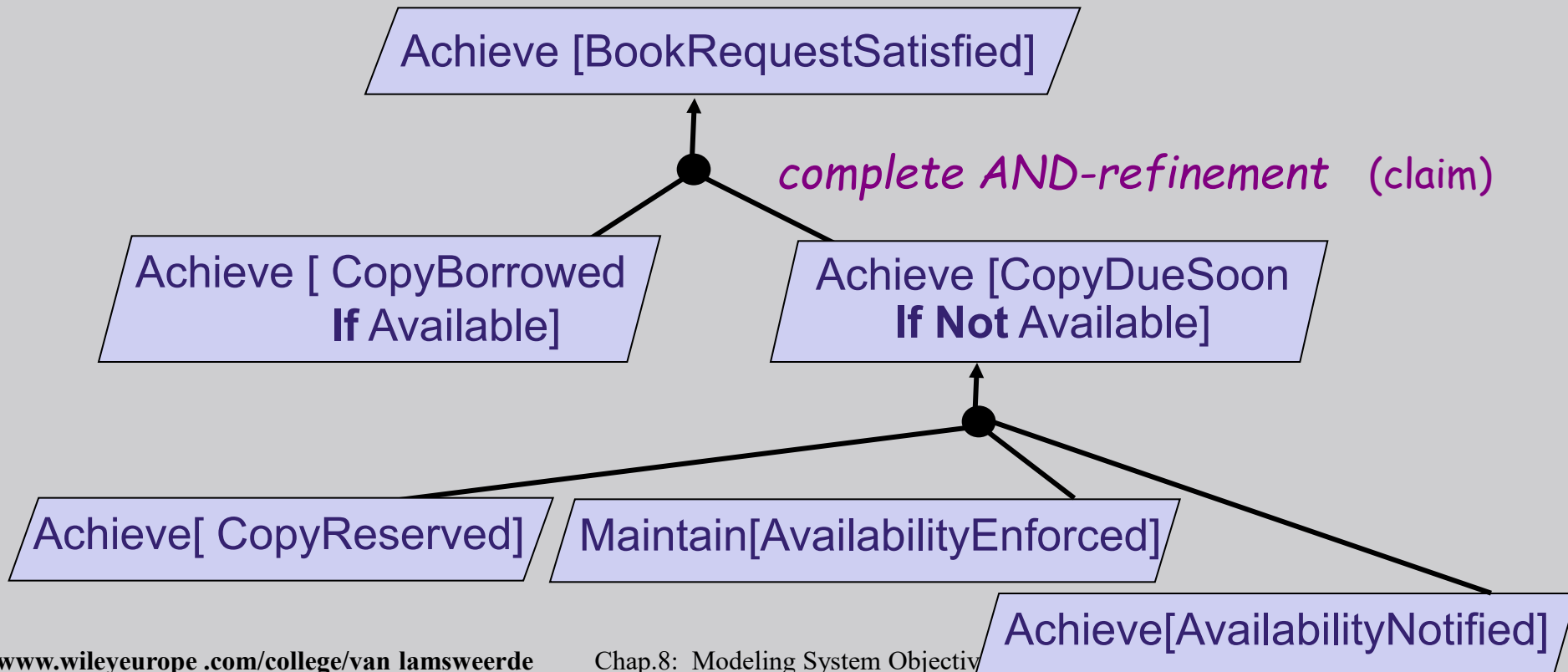


AND-refinements

amacı tümüyle içerecek şekilde genişletilmelidir

- ◆ $\{G_1, \dots, G_n\}$ is a sufficient for satisfying G in view of known domain properties

$$\{G_1, \dots, G_n, \text{Dom}\} \models G$$



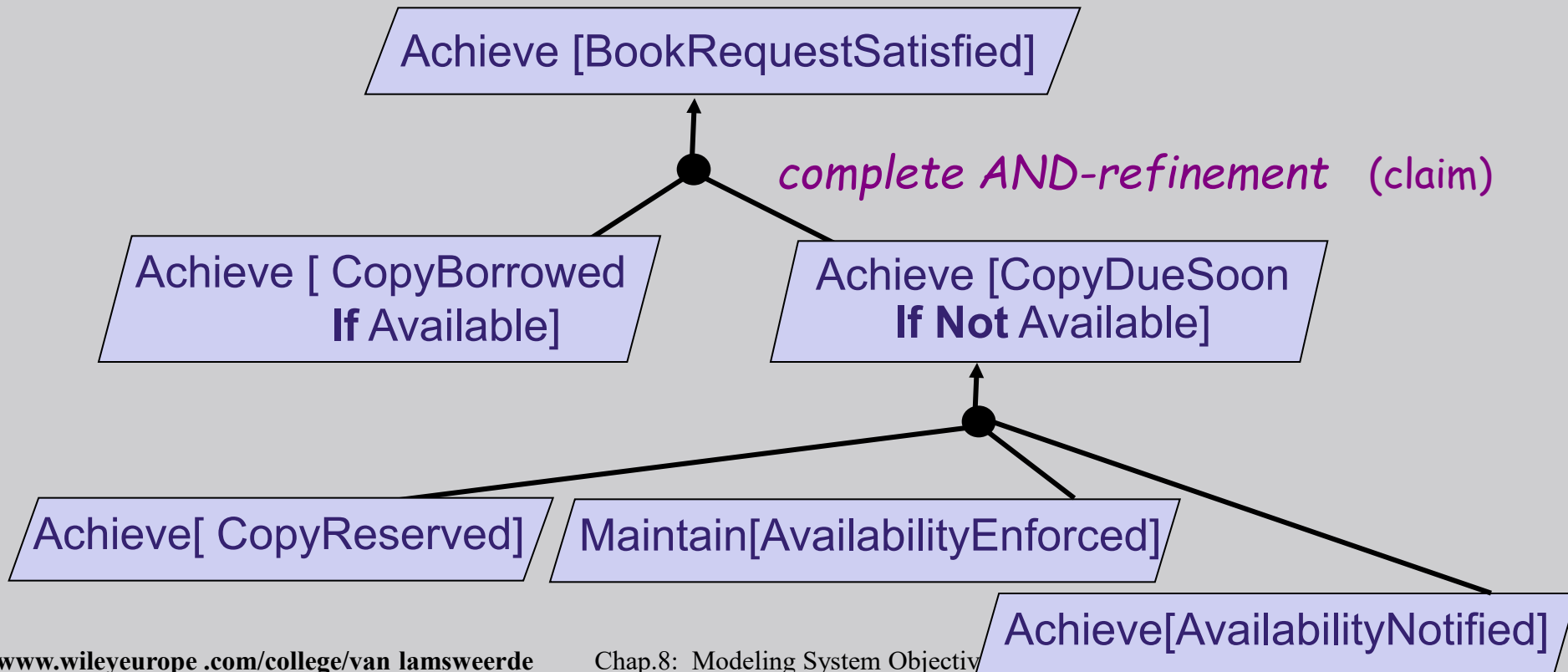


AND-refinements

amacı tümüyle içerecek şekilde genişletilmelidir

- ◆ $\{G_1, \dots, G_n\}$ is a sufficient for satisfying G in view of known domain properties

$$\{G_1, \dots, G_n, \text{Dom}\} \models G$$





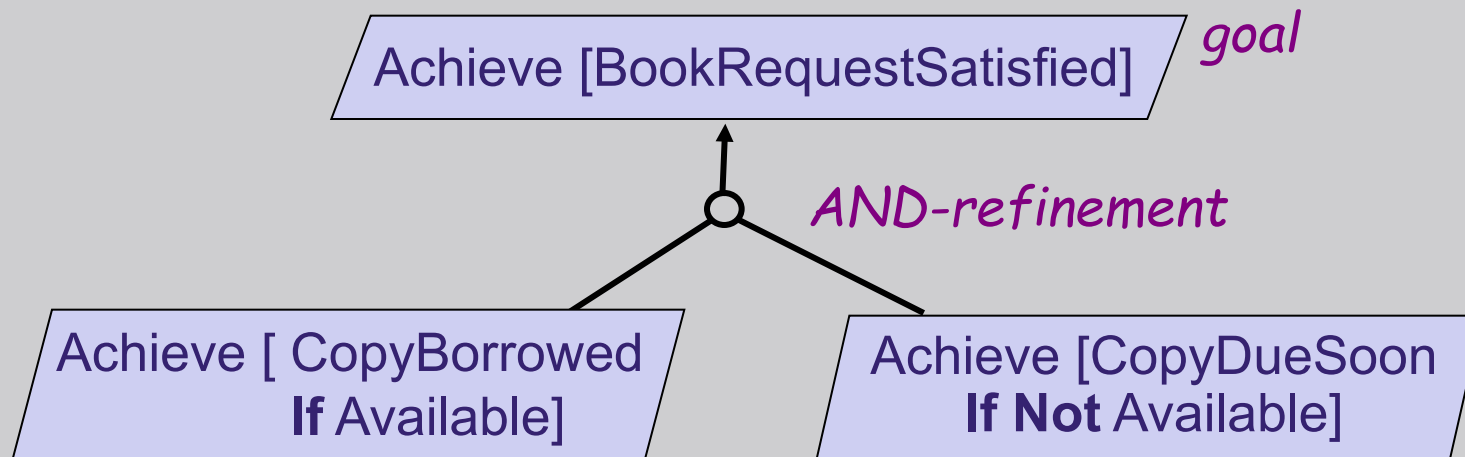
Goal refinement

Amacın Ayrıntılandırılması

- ◆ An **AND-refinement** of goal G into subgoals $G_1, \dots, G_n$ states that G can be satisfied by satisfying $G_1, \dots, G_n$

The set $\{G_1, \dots, G_n\}$ is called **refinement** of G

Subgoal G_i is said to **contribute positively** to G



Def *In case a requested book has no copy available for check out, a copy of that book shall be made available within 2 weeks for check out by the requesting patron.*



AND-refinementd ne zaman tamamlanır?

◆ requirements completeness ne zaman gerçekleşir?

Bu ayrıntılı açıklamaları içeren genişlemeler gereksinimlerin tamamlanmasına karar vermede önemlidir.

◆ «Domain properties» davranışsal amaçların ayrıntılandırılmasının tamamlandığı ile ilgili bir hedef olabilir.

- classified as ...

• domain invariants: known to hold in every state

"train doors are either open or closed"

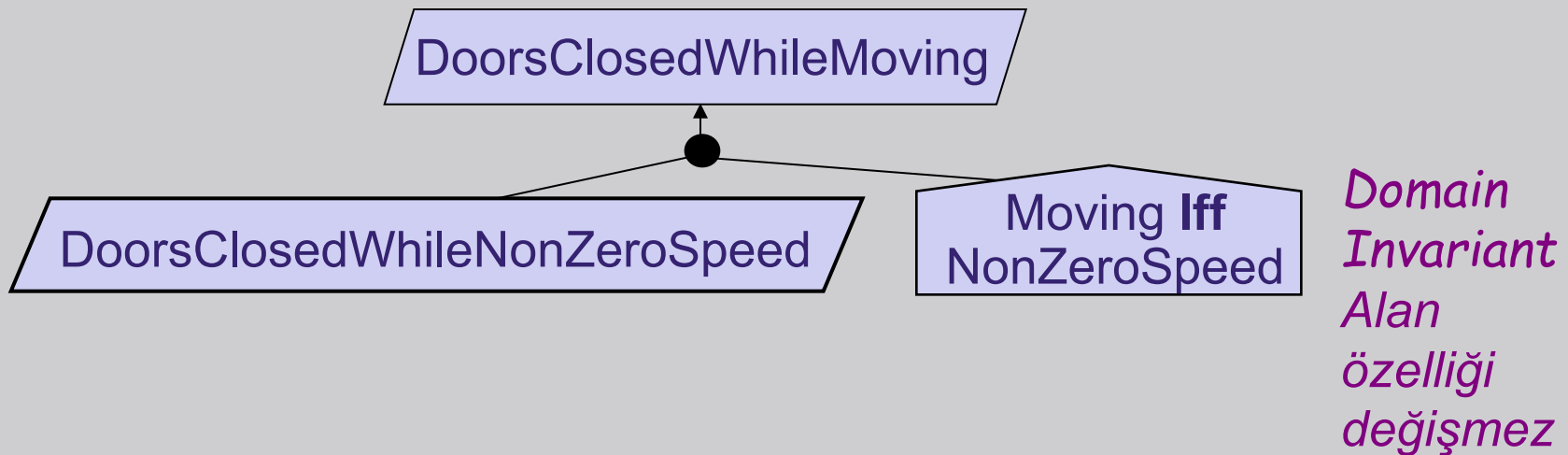
• domain hypotheses: assumed to hold in specific states

"railway tracks are in good conditions ..."

- attached to conceptual objects in the object model



Domain properties in AND-refinements AND-refinements ile Alan Özelliklerinin İfadesi

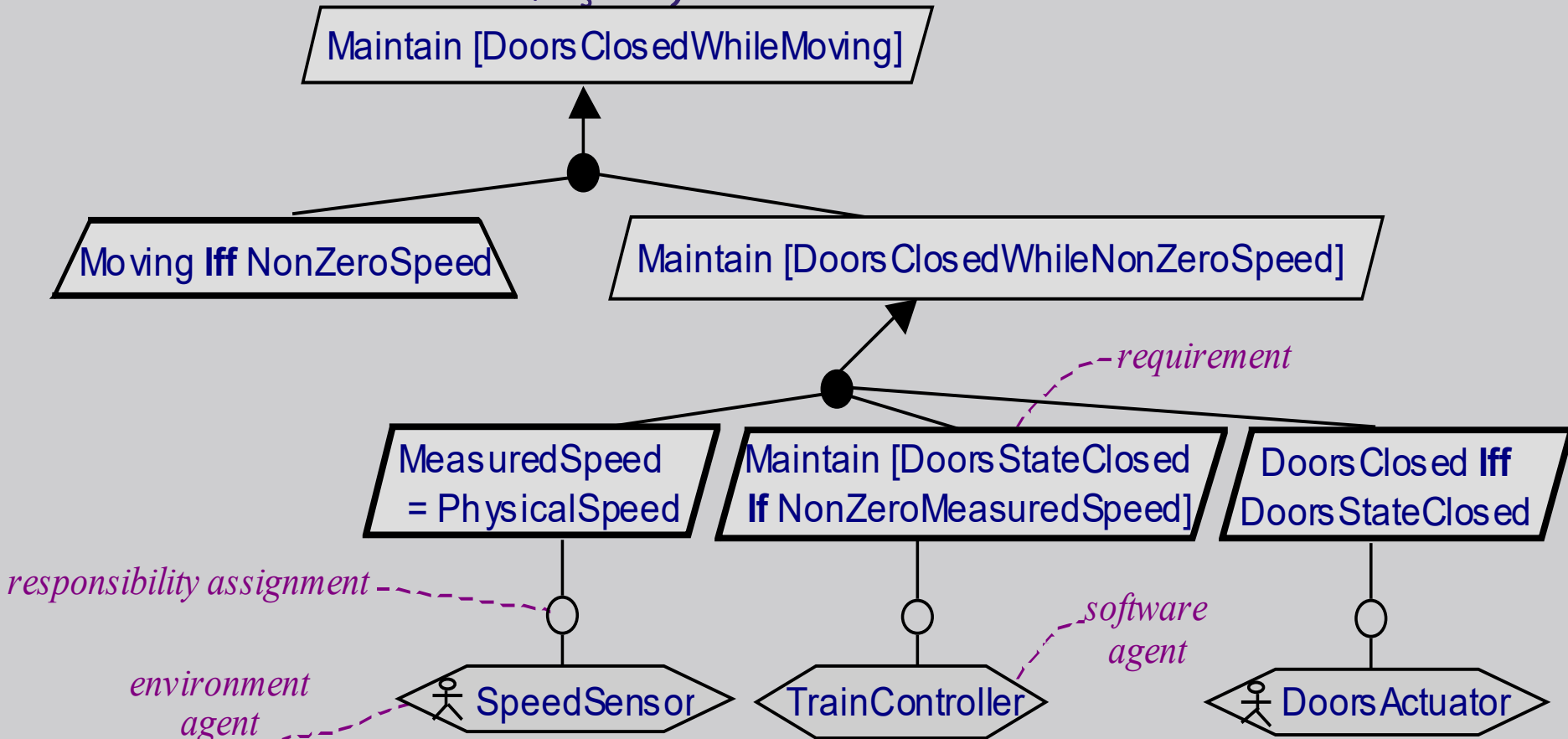


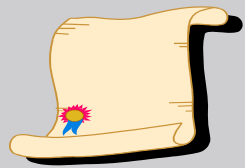


Refinement trees

Ayrıntılandırma Ağaçları

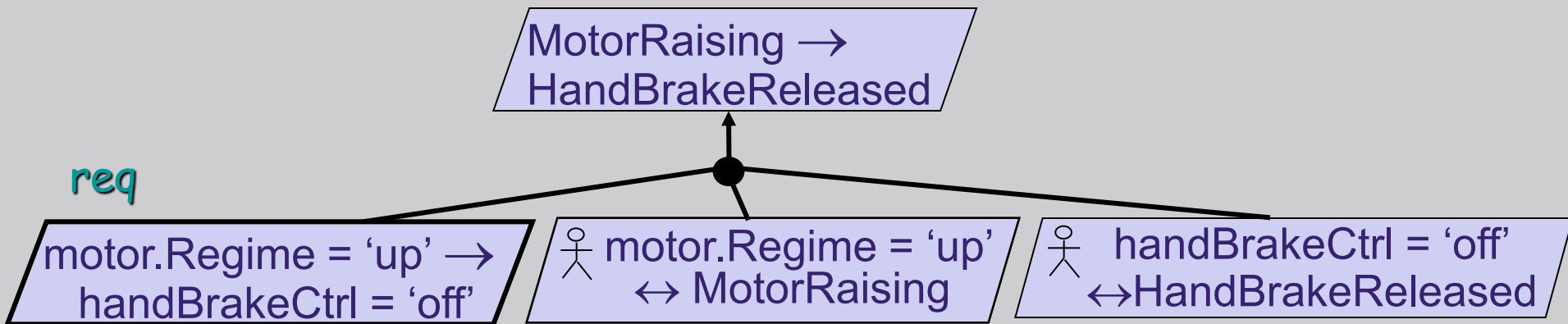
- ◆ Amaçlar rekürsif olarak ayrıntılandırılabilir.
(Yaprak düğümler) Leaf nodes = goals assignable to single system agents (tek bir sistem etkenine (agent) atanabilen amaçlar)





Ayrıntılandırma Ağacının Adım adım oluşturulması

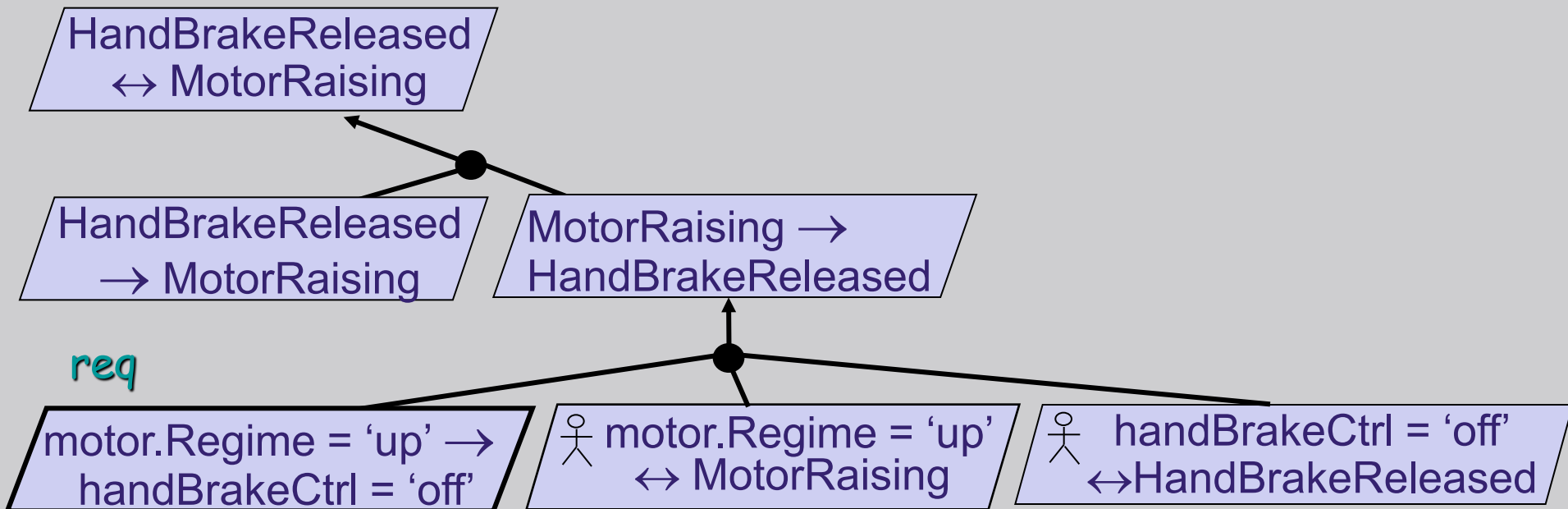
Gereksinimlerin yüksek düzeydeki etkilere (diğer bir ifade ile çözümlemelerine) adım adım erişilmesi





Ayrıntılandırma Ağacının Adım Adım Oluşturulması

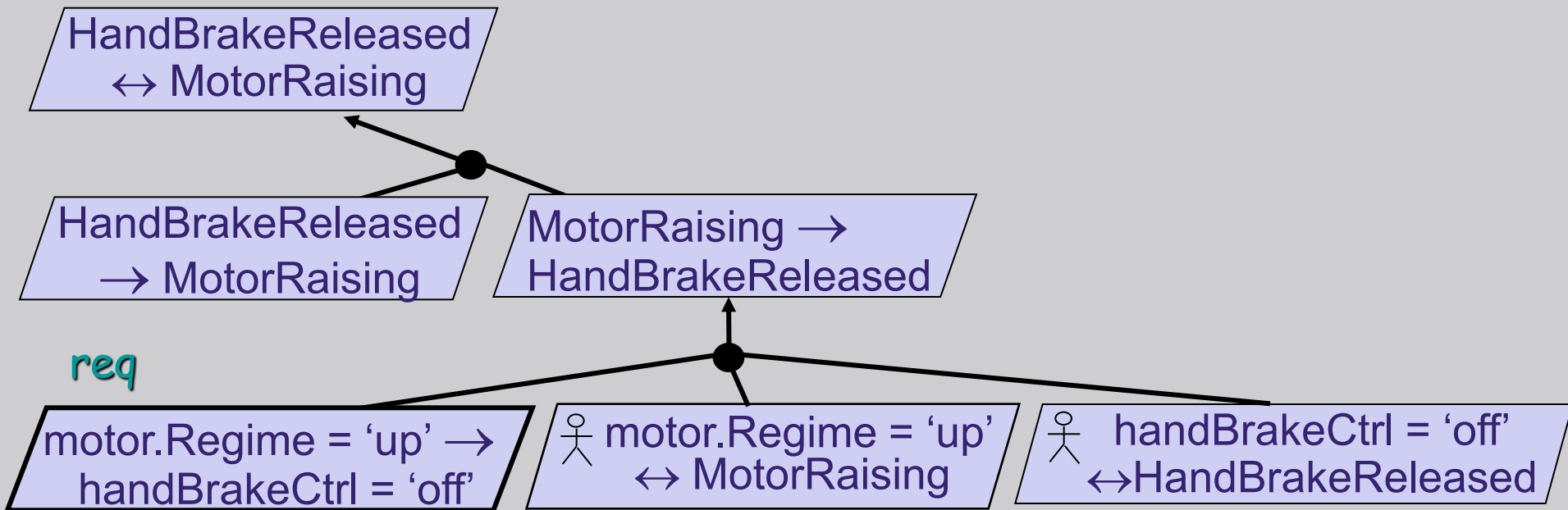
Gereksinimlerin yüksek düzeydeki etkilere (diğer bir ifade ile çözümlemelerine) adım adım erişilmesi





Ayrıntılandırma Ağacının Adım Adım Oluşturulması

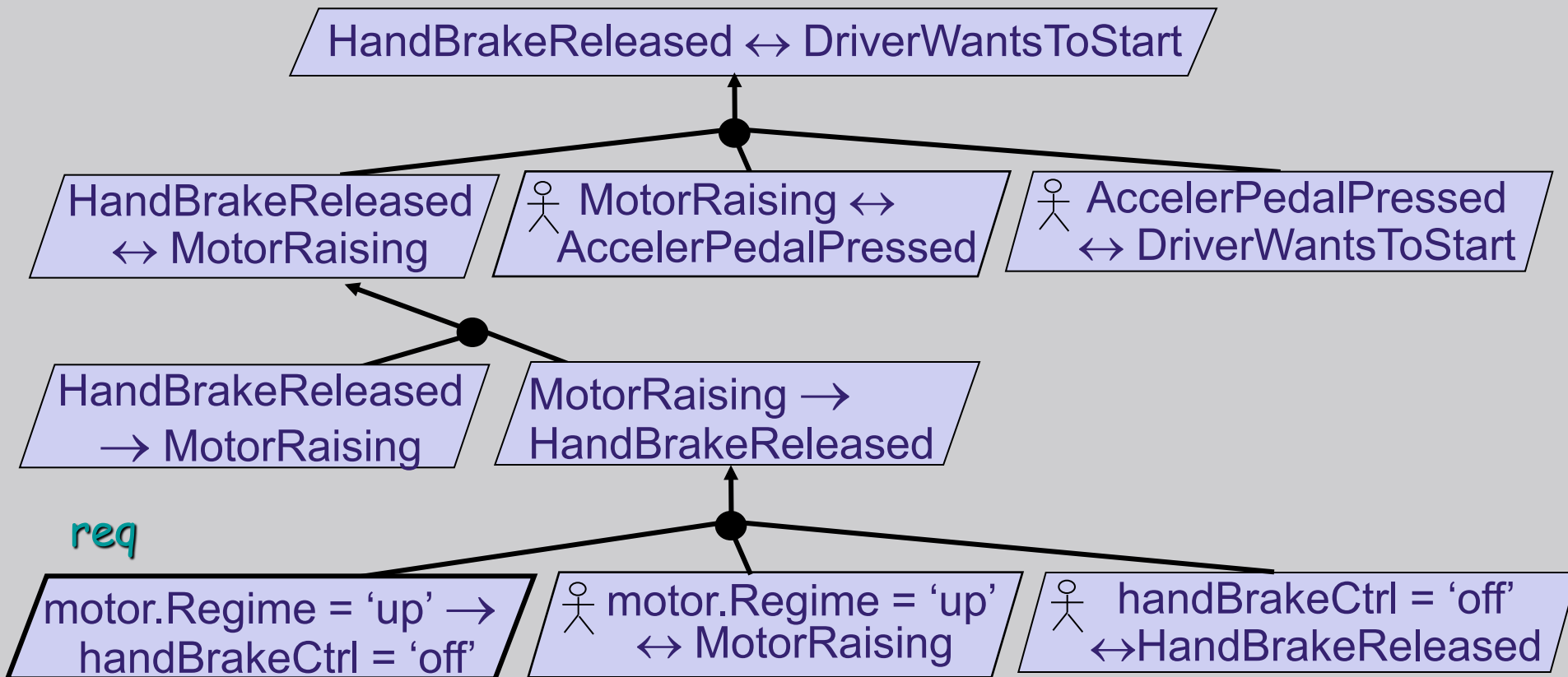
Gereksinimlerin yüksek düzeydeki etkilere (diğer bir ifade ile çözümlemelerine) adım adım erişilmesi





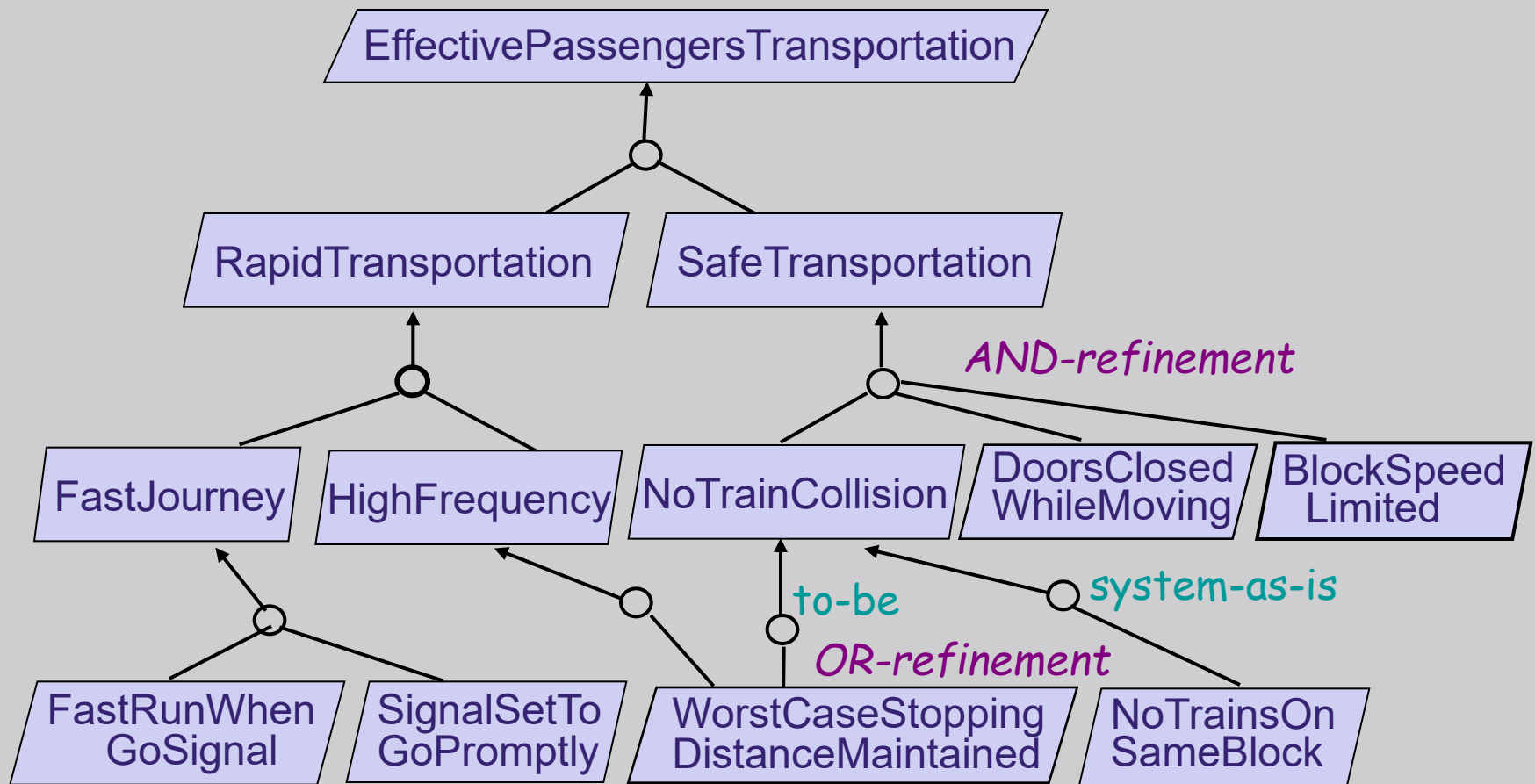
Ayrıntılandırma Ağacının Adım Adım Oluşturulması

Gereksinimlerin yüksek düzeydeki etkilere (diğer bir ifade ile çözümlemelerine) adım adım erişilmesi





Amaca Yönelik Diyagram AND/OR graf





AND-refinement ne zaman tamamlanır?

◆ requirements completeness ne zaman gerçekleşir?

Bu ayrıntılı açıklamaları içeren genişlemeler gereksinimlerin tamamlanmasına karar vermede önemlidir.

◆ «Domain properties» davranışsal amaçların ayrıntılandırılmasının tamamlandığı ile ilgili bir hedef olabilir.

- classified as ...

• domain invariants: known to hold in every state

"train doors are either open or closed"

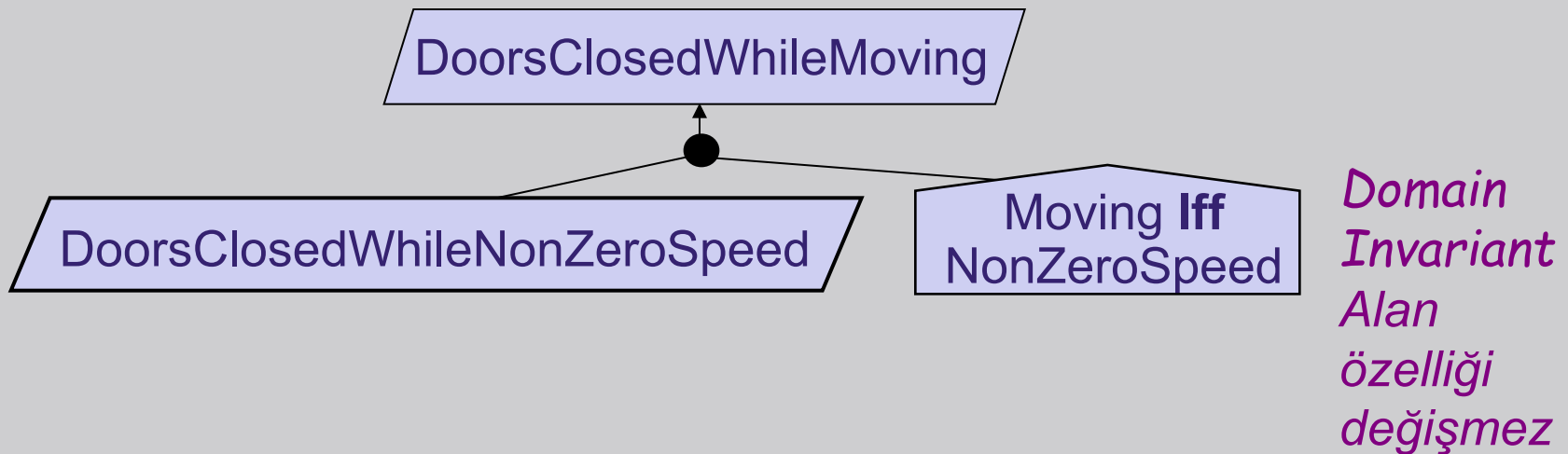
• domain hypotheses: assumed to hold in specific states

"railway tracks are in good conditions ..."

- attached to conceptual objects in the object model



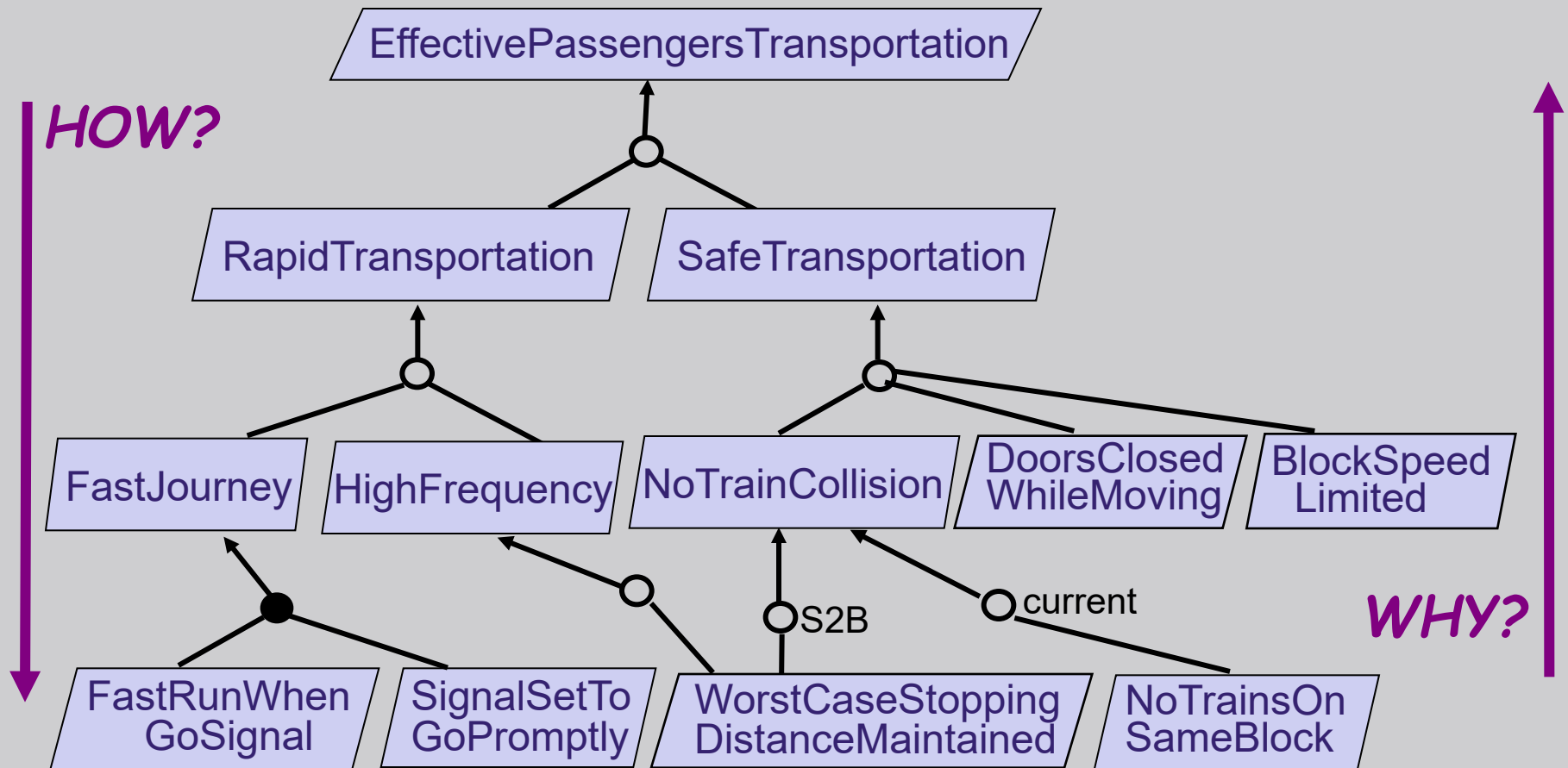
Domain properties in AND-refinements AND-refinements ile Alan Özelliklerinin İfadesi





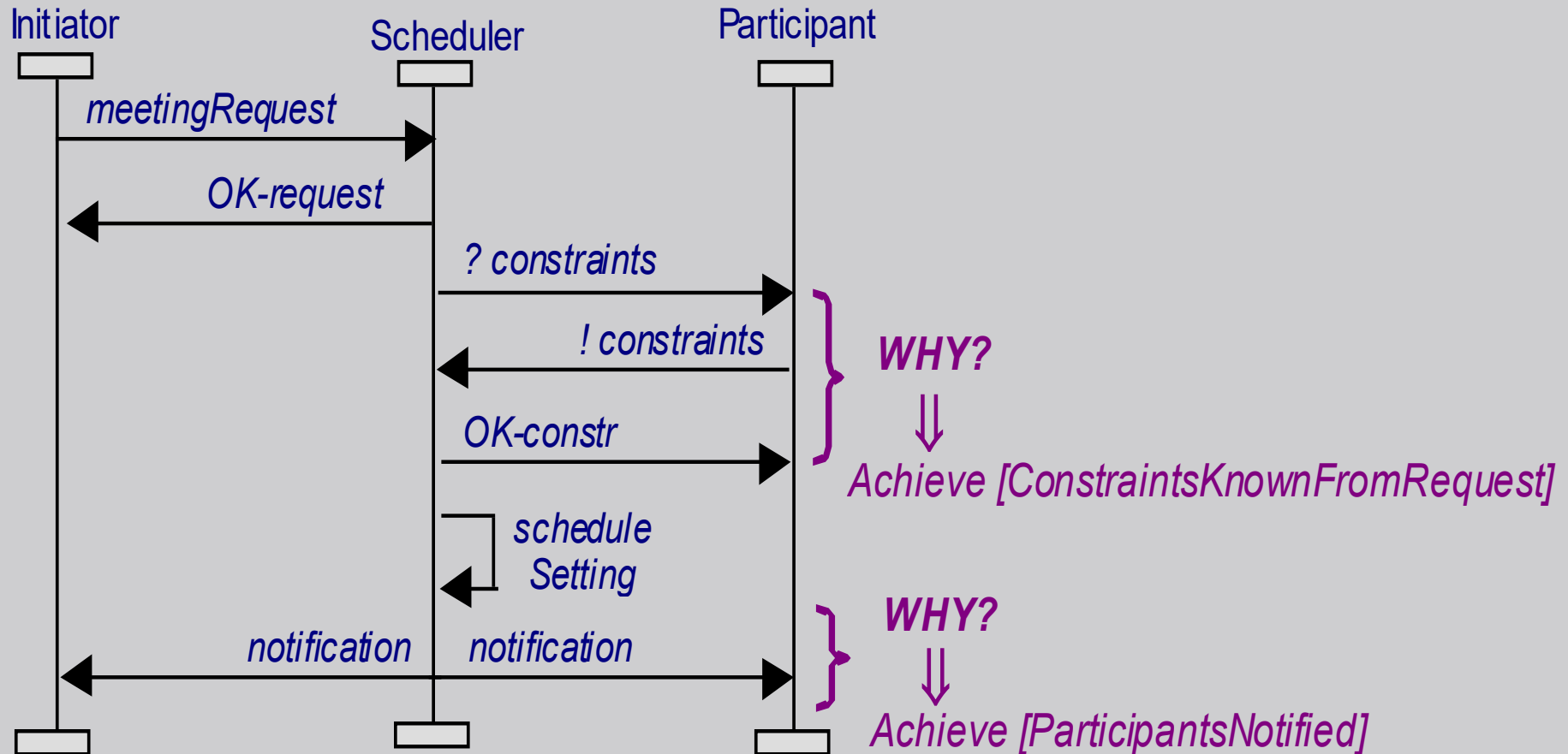
Amaca Yönelik Modellerin Oluşturulması

HOW ve WHY soruları





Amaçların WHY sorularını içeren senaryolardan tanımlanması

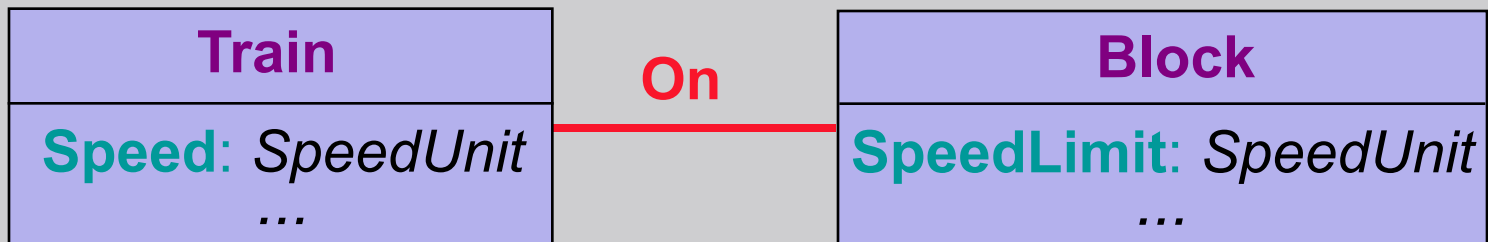




Objects, Associations, Attributes from Goal Specifications

Goal Maintain [BlockSpeedLimited]

Def The *speed* of a *train* *on* a *block* may never exceed the *limit* associated with that block

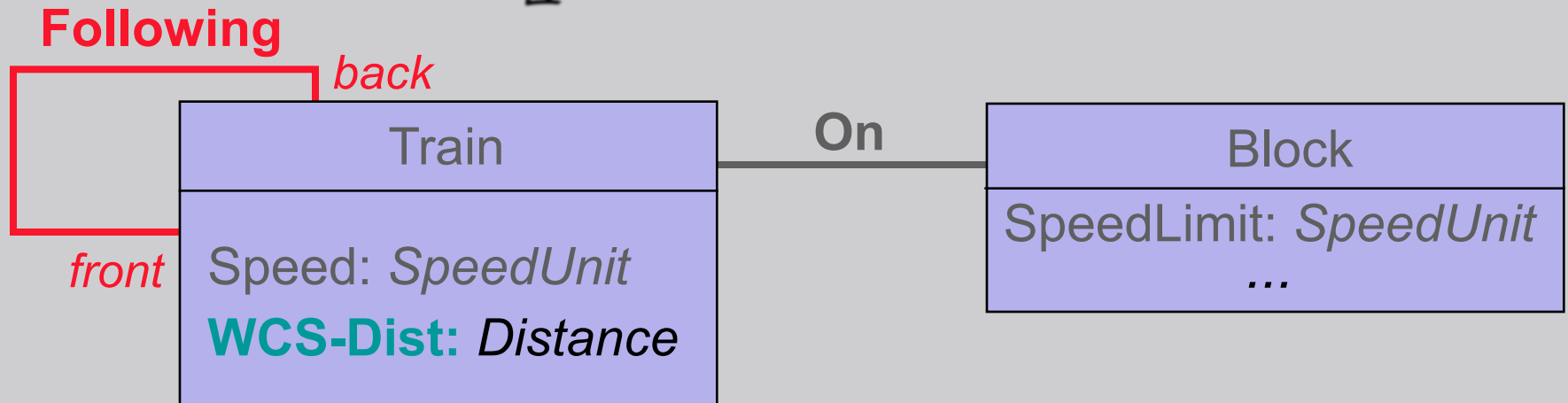




Objects, Associations, Attributes from Goal Specifications

Goal Maintain [WorstCaseStoppingDistance]

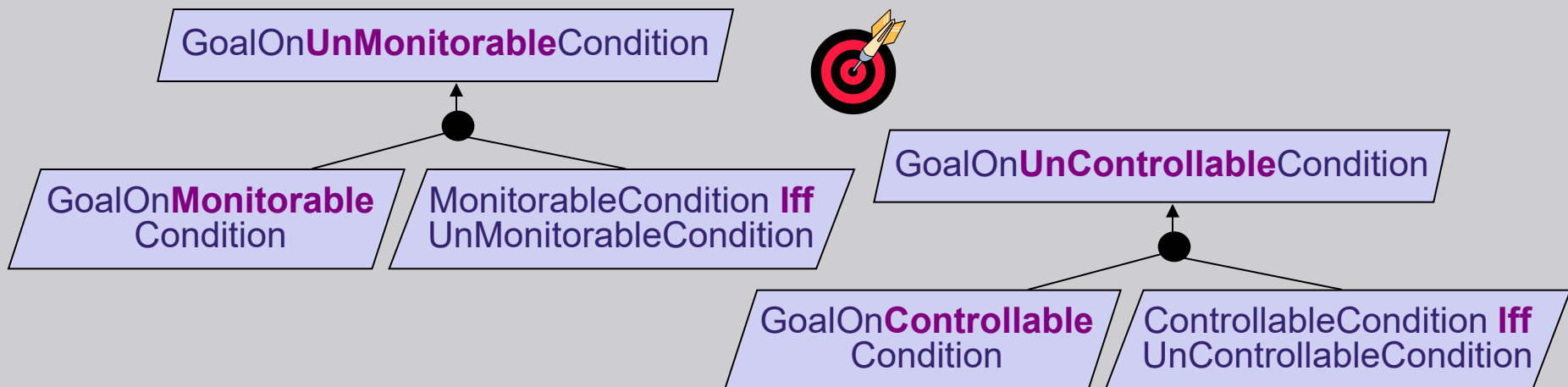
Def The *distance* between two trains *following* each other shall be sufficient to prevent the back train from hitting the front train in case the latter stops suddenly





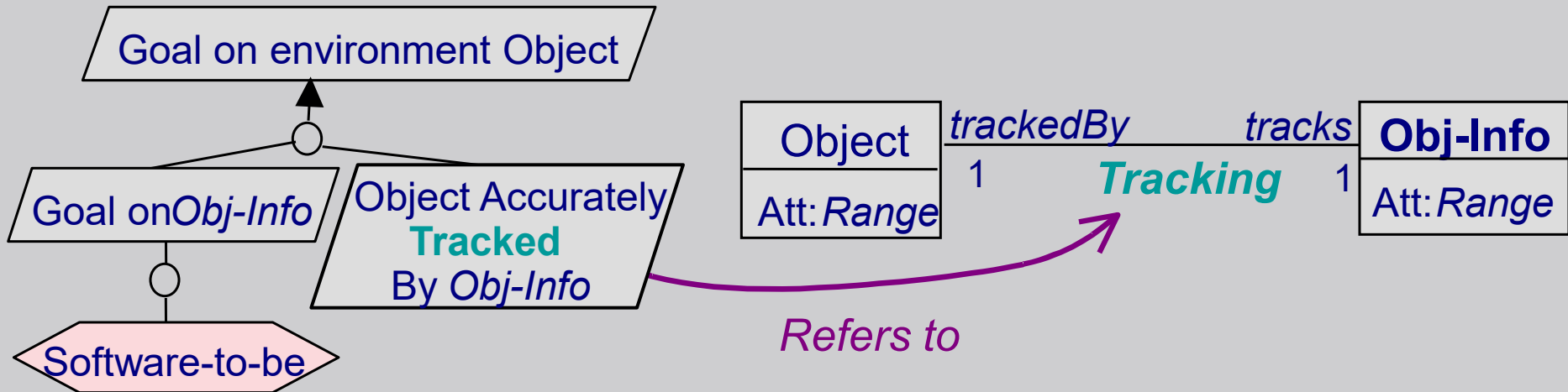
Introducing software-environment tracking associations

- ☞ For goal assignment to software-to-be, we must introduce shared “images” of environment objects referenced by the goal
- the shared object tracking its environment counterpart must accurately reflect it ($\Rightarrow$ new accuracy goal)
 - e.g. TrainInfo(Speed, Position) **tracking** Train(Speed, Position)
LoanInfo **tracking** Loan, PatronInfo **tracking** Patron
 - cf. goal refinement pattern seen before:





Introducing software-environment tracking associations: a general pattern





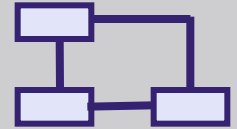
Identifying associations from domain invariants on multiple objects

- ◆ Domain properties in goal refinements are to be defined in annotations of the object model (as seen before)
 - invariants on single objects they constrain
- ◆ Invariant seeming to constrain multiple objects ...
 - ? => ? constrains **missing association** among these ?
 - e.g. “A platform cannot accommodate more than one train at a time”
 - ... **constrains** platforms ? ... **or** trains ?
 - => to be attached to missing association **At linking**
Train **and** Platform





Amaç model ile Nesne model arasındaki türetmeler karşılıklı (bidirectional) ilişkilidir



- ◆ Amaçlardan nesnelere ilişkiler incelendi
- ◆ Nesnelerden amaçlara ilişkiler ise alana ilişkin kavramlardır (domain concepts). Yani, nesne (object) model içerisinde kesinlikle mevcut olmalıdır.

Domain concepts that should “obviously” appear in object model ...

WHY? => amaç modelde olmayan **modeldir**

İlişkilerdeki çokluk bildirimleri ile bir birliktelik kurulur.

prescriptive? => amaç modelde olmayan **amaçlardır**

WHY? => parent goals

HOW? => subgoals





Entity, association, agent, or event ?

Herhangi bir X için: amaç betimlemelerini (specifications) verecek kavramsal nesne instances of X are defined in one single state

⇒ event e.g. StartTrain

- X 'in örnekleri aktivasyon belirtir: Diğer nesne örneklerinin kontrol davranışları

⇒ agent e.g. DoorsActuator

- X 'in örnekleri passive (etkilenmez), autonomous (bağımsız) özelliktedir.

⇒ entity e.g. Train

- X 'in örnekleri pasif, diğer nesnenin örnekleri ile bağlantılıdır.

⇒ association e.g. Following (Train, Train)



Her N parçasının N -li ilişkileri nesneler olarak düşünülür. Her bir ilişki araştırılır.