

YAZILIM MÜHENDİSLİĞİ YÖNETİM SÜRECİ ONTOLOJİSİ

Barış Ulu, Banu Diri
Yıldız Teknik Üniversitesi
Bilgisayar Mühendisliği
Bölümü





SUNUM PLANI

- Anlamsal Web ve Ontolojiler
- Anlamsal Yazılım Mühendisliği
 - Yazılım Mühendisliği Süreçleri
 - Anlamsal Süreç Modelleme
- Yazılım Mühendisliği Yönetim Süreci Ontolojisi
- Sonuçlar

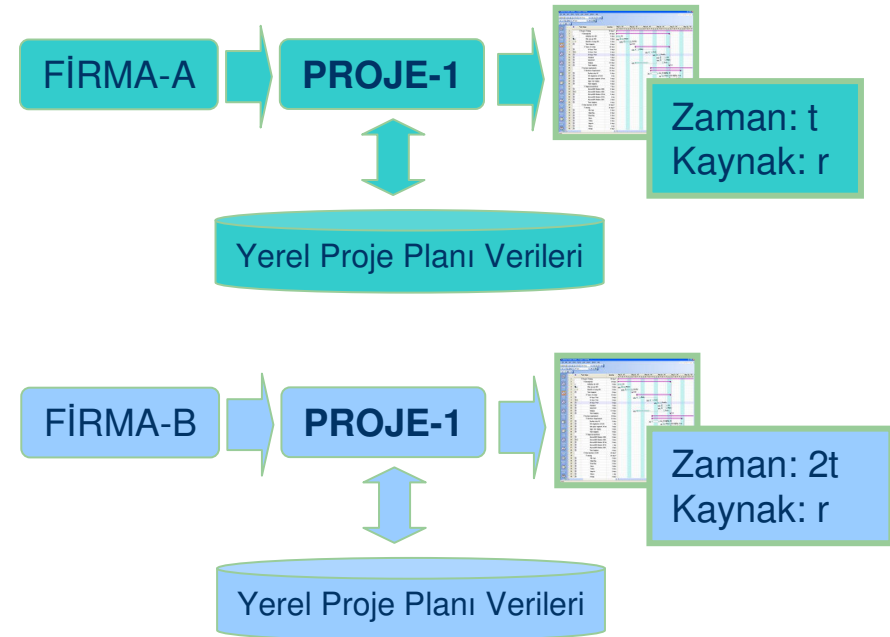


ANLAMSAL WEB VE ONTOLOJİLER

- Günümüzde veri modelleri;
 - Yapısal olmayan, dinamik, dağıtık ve hızla büyüyen bir yığın,
 - Verinin ne anlama geldiğini ifade etmeye çalışan mantıksal işlenebilirlikten uzak, standartlaştırılmış bir artık (garbage),
 - Farklı merkezlerde bulunan verilerin birbirleri arasında bilmediğimiz mantıksal ilişkilerinin kullanıcılara sunulmaması.
- Verinin sunumu önemlidir ve bu sunum, veriler arası anlamsal ilişkilerin yaratılması ve yapılandırılması ile mümkündür.
- Geleceğin veri modelleri;
 - Dağıtık verilerin birbirleri ile olan ilişkileri tanımlanabilirse bilgiye kolaylıkla ulaşılabilir,
 - Veriler arası ilişkiler ve sunum *RDF (Resource Description Framework)* yolu ile yapılmaktadır,
 - RDF tarafından biçimlendirilen ve ilişkilendirilen bu veri kümeleri *Ontolojiler* yolu ile saklanmaktadır,
 - Ontolojilerde tutulan verilerden *çıkarsama* ile **bilgi** elde edilmektedir.

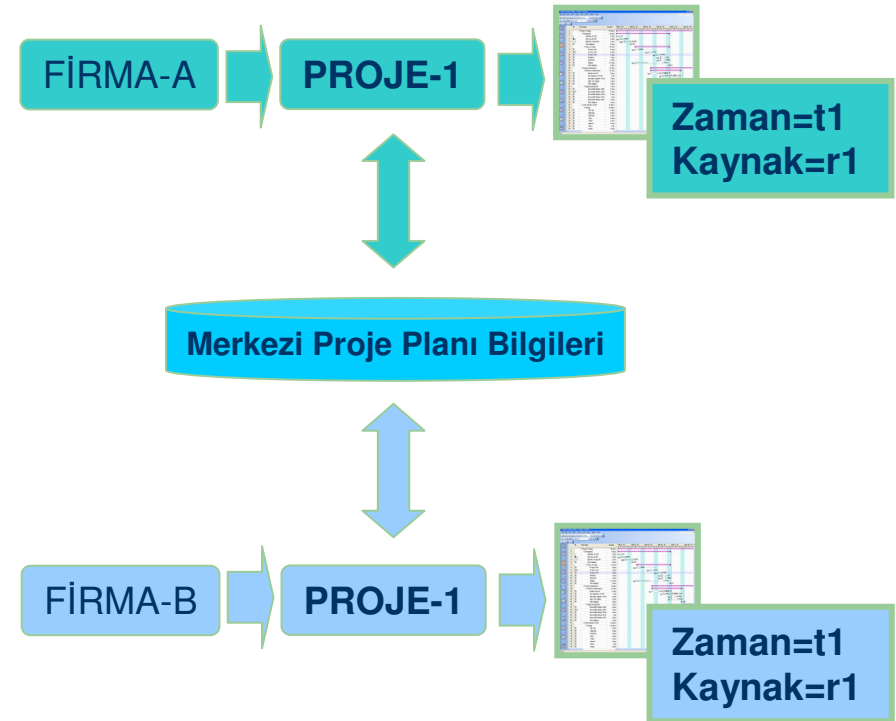
ANLAMSAL YAZILIM MÜHENDİSLİĞİ

- Yazılım Mühendisliği Süreçleri;
 - Bilgi (knowledge) yerine veri bağımlıdır,
 - Yerel veriler ile beslenirler,
 - İyileştirmeye yöneliktirler,
 - Durağan modellerdir,
 - Veriler arası söz-dizimsel ilişkiler mevcuttur,
 - Ne, ne kadar sorularına yanıt bulabilirler,
 - Veriler yeniden kullanılabilir değildir; proje ya da ürün odaklıdır,
 - CMM (Capability Maturity Model), SPICE (Software Process Improvement and Capability dEtermination)

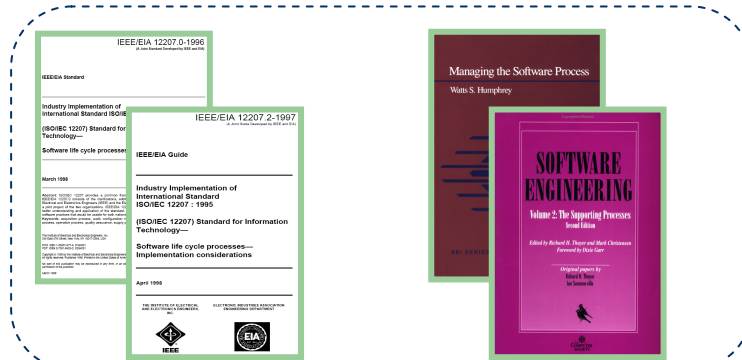


ANLAMSAL YAZILIM MÜHENDİSLİĞİ

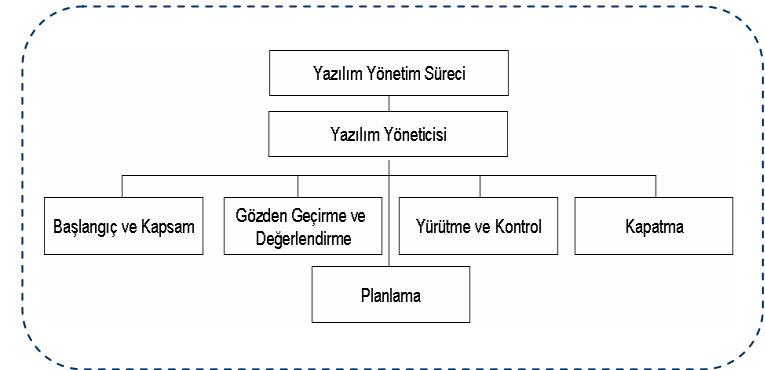
- Anlamsal süreç modelleme;
 - Başarı ile sonuçlanmış yazılım projelerinden bilgi alabilmek,
 - Zaman kaybı olmaksızın proje planlarını hazırlayabilmek,
 - İstatistiksel bilgiler elde edilerek gerekli kaynakların belirlenmesini sağlamak,
 - Doğru ya da doğruya yakın zamanlama çizelgeleri üretebilmek,



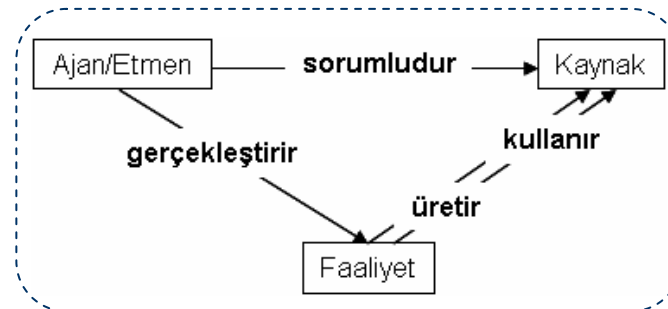
YAZILIM MÜHENDİSLİĞİ YÖNETİM SÜRECİ ONTOLOJİSİ



Yazılım Mühendisliği Yönetim Kural ve Standartları

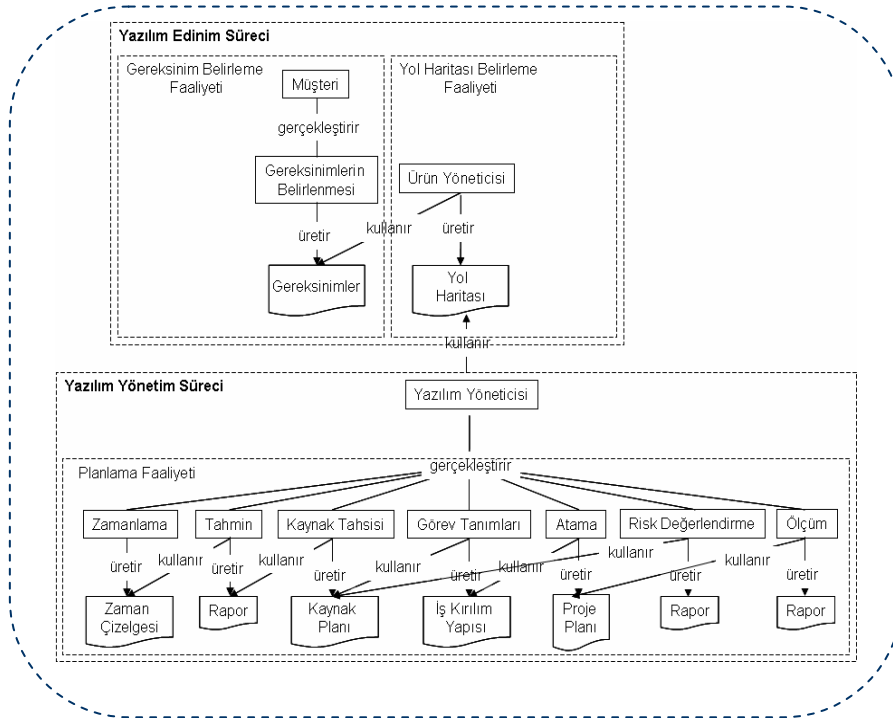


Yazılım Mühendisliği Yönetim Süreci Taksonomisi



Somut Süreç Modeli (SPEM)

YAZILIM MÜHENDİSLİĞİ YÖNETİM SÜRECİ ONTOLOJİSİ



Yazılım Mühendisliği Yönetim Süreci Üst-veri Modeli

Yazılım yönetim süreci taksonomisi kullanılmıştır

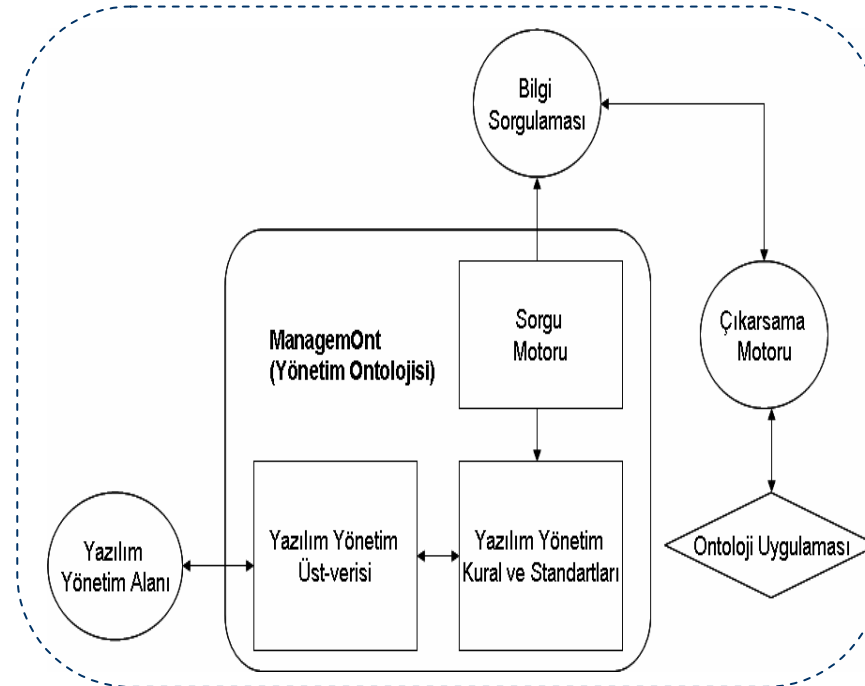
Somut süreç modeli uygulanmıştır

Her faaliyeti işleten bir ajan/etmen bulunmaktadır

Her faaliyetin bir girdi ve çıktısı bulunmaktadır

Her ajan/etmen, faaliyet, girdi ve çıktı (kaynak) yazılım yönetim ontolojisi için bir kavramı ifade etmektedir

YAZILIM MÜHENDİSLİĞİ YÖNETİM SÜRECİ ONTOLOJİSİ



Yazılım Mühendisliği Yönetim Süreci Ontolojisi: ManagemOnt



YAZILIM MÜHENDİSLİĞİ YÖNETİM SÜRECİ ONTOLOJİSİ

ManagemOnt Modelleme Temelleri

ManagemOnt'ta kavramlar nelerdir?

ManagemOnt'ta aktiviteler nelerdir?

ManagemOnt'ta nesnelere nelerdir?

ManagemOnt'ta çıktılar nelerdir?

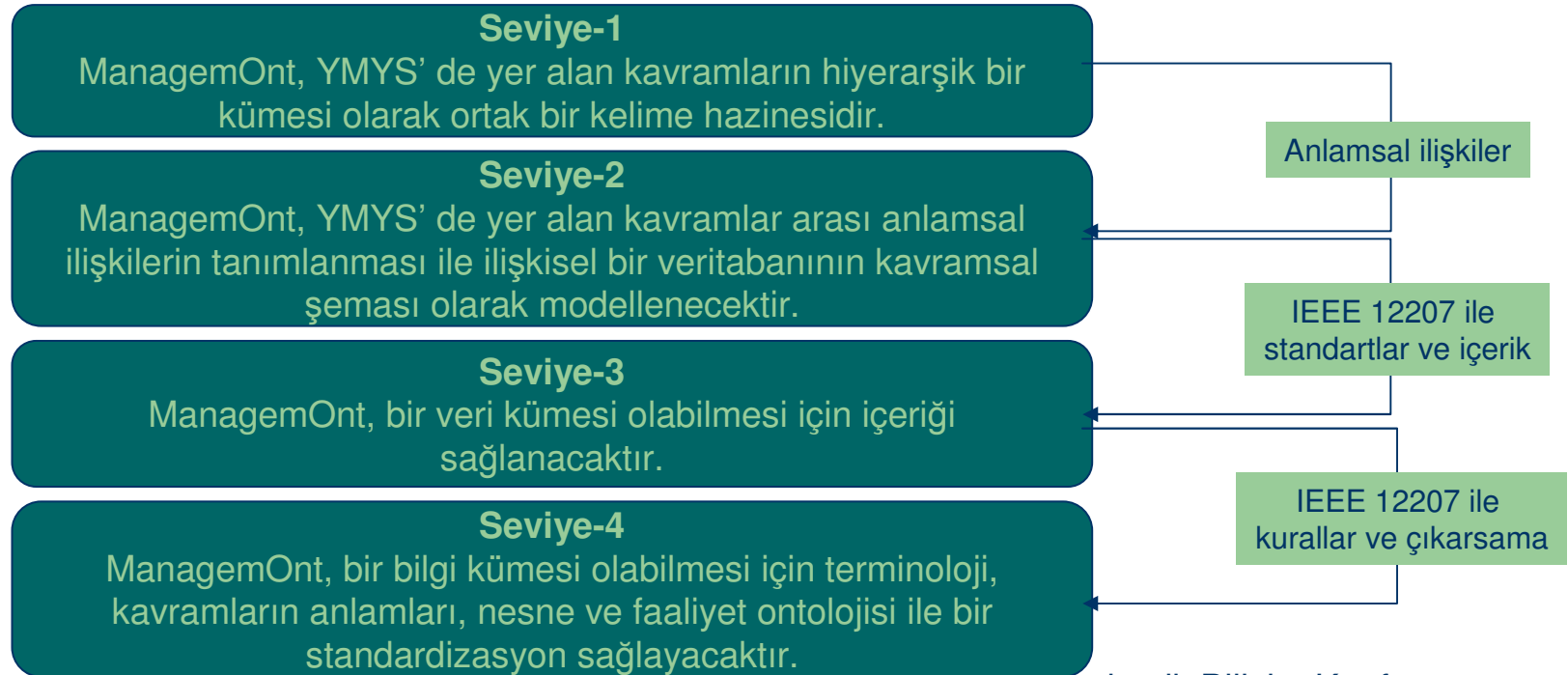
ManagemOnt'ta ajan/etmen nedir?

ManagemOnt'ta "is-a" ve "part-of"
bağlantıları nelerdir?



YAZILIM MÜHENDİSLİĞİ YÖNETİM SÜRECİ ONTOLOJİSİ

ManagemOnt Seviye Analizi





YAZILIM MÜHENDİSLİĞİ YÖNETİM SÜRECİ ONTOLOJİSİ

ManagemOnt Tip Analizi

Alan Ontolojisi

Görev-bağımlı ontoloji

Görev-bağımsız ontoloji

Faaliyet-bağımlı ontoloji

Faaliyet-bağımsız ontoloji

Nesne ontolojisi

Faaliyet ontolojisi

ManagemOnt

Yönetim kavramları

Yönetim aktiviteleri

YAZILIM MÜHENDİSLİĞİ YÖNETİM SÜRECİ ONTOLOJİSİ

ManagemOnt Metodoloji Analizi

Metodoloji

Uschold & King

Amaç

Ontoloji

Değerlendirme

Belgeleme

TOVE

Senaryolar

Uzmanlık soruları

Terimler

Aksiyomlar

Methontology

Proje yönetimi

Ontoloji

Destek faaliyetleri

On-to-Knowledge

Bilgi süreci

Bilgi üst-süreci

AFM

Görev birimlerinin tanımlanması

Faaliyetlerin organizasyonu

Görev yapısının analizi

Alan kavramlarının organizasyonu

IEEE
610-12.1990

ManagemOnt
taksonomisi

ManagemOnt
somut süreç
modeli

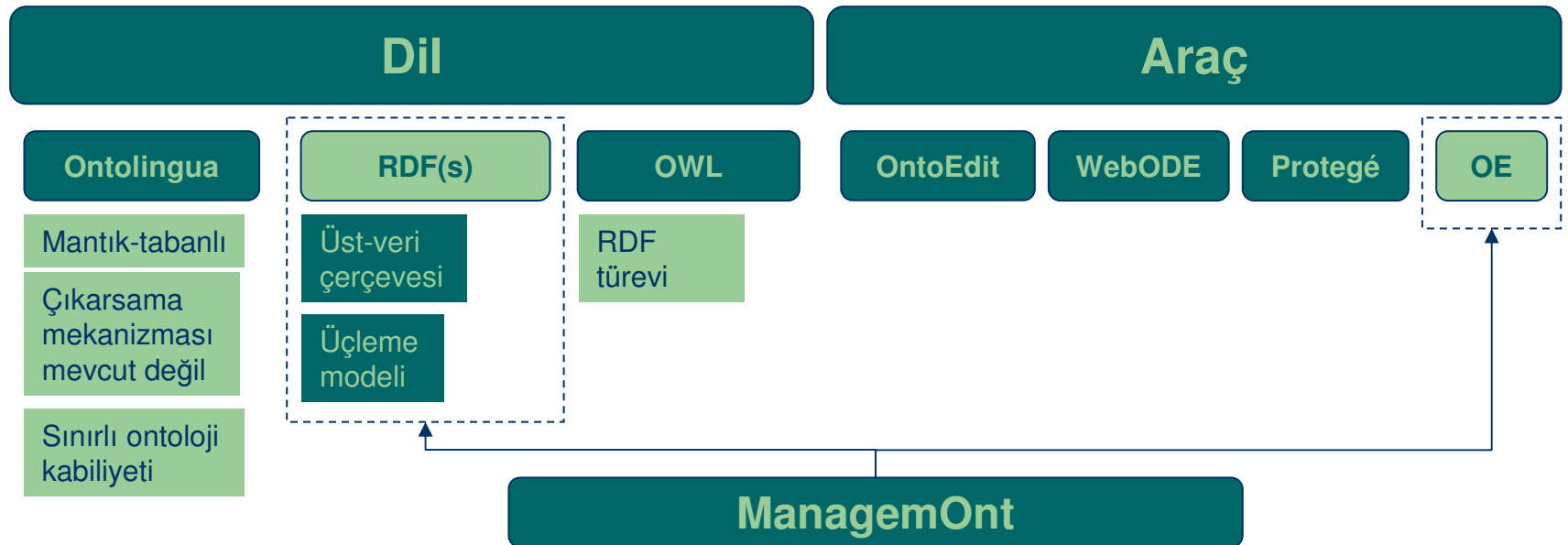
ManagemOnt
üst-verisi ve
ilişkilerin tanımlanması

29.01.2008

X. Akademik Bilişim Konferansı,
Çanakkale, 2008

YAZILIM MÜHENDİSLİĞİ YÖNETİM SÜRECİ ONTOLOJİSİ

ManagemOnt Gerçekleştirim Analizi



YAZILIM MÜHENDİSLİĞİ YÖNETİM SÜRECİ ONTOLOJİSİ

ManagemOnt Kavram Analizi

ManagemOnt: Ana Süreç

İSİM: Sürecin ismi

URL: Bilgi kümesinin adresi

AKIŞ SENARYOSU:
Süreci oluşturan akış senaryosu

Ajan/Etmen: Süreçte Yeralan Aktör

İSİM: Ajan ya da etmenin ismi

URL: Bilgi kümesinin adresi

acts on: Ajan ya da etmenin gerçekleştirdiği eylemler

Kaynak: Girdi ya da çıktılar

İSİM: Kaynağın ismi

URL: Bilgi kümesinin adresi

required by: İşlem için bu kaynağa ihtiyaç duyan eylemler

Faaliyet: Süreçte yer alan eylemler

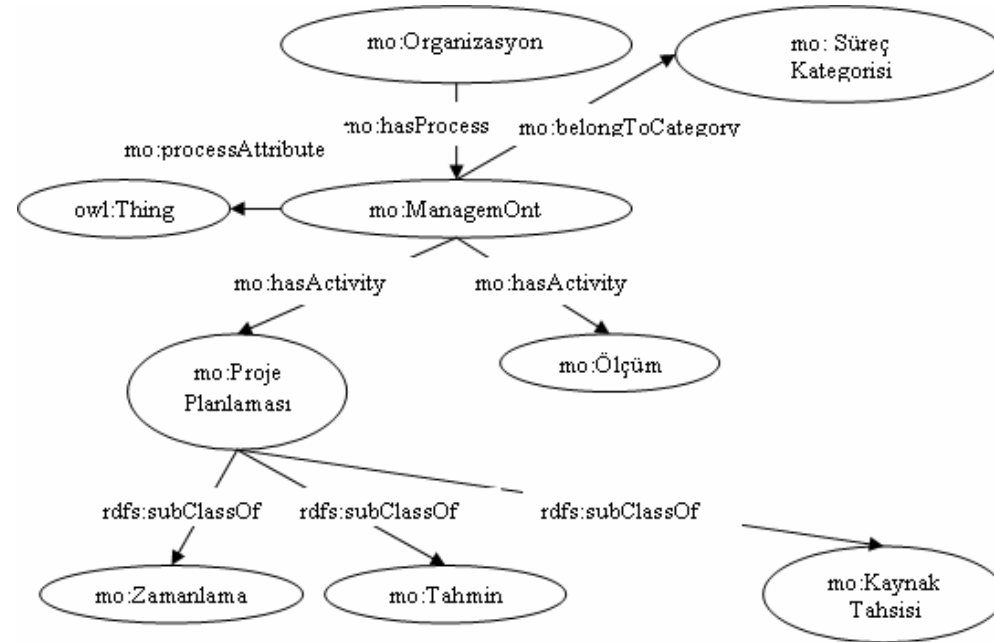
İSİM: Faaliyetin ismi

URL: Bilgi kümesinin adresi

provides: Faaliyet tarafından sağlanan kaynaklar (çıktı)

YAZILIM MÜHENDİSLİĞİ YÖNETİM SÜRECİ ONTOLOJİSİ

ManagemOnt RDF Grafiği (kısmi)



YAZILIM MÜHENDİSLİĞİ YÖNETİM SÜRECİ ONTOLOJİSİ

ManagemOnt RDF Grafiği (kısmi)

```
<rdf_:Ajan rdf:about="&rdf_;OSSD_Process_Model_00076"  
  rdf_:name="Yazılım Yöneticisi"  
  rdfs:label="Yazılım Yöneticisi">  
  <rdf_:acts_on rdf:kaynak=  
    "&rdf_;OSSD_Process_Model_00096"/>  
  <rdf_:acts_on rdf:kaynak=  
    "&rdf_;OSSD_Process_Model_00108"/>  
</rdf_:Ajan>  
<rdf_:Faaliyet rdf:about="&rdf_;OSSD_Process_Model_00096"  
  rdf_:name="Proje Planlama"  
  rdfs:label="Proje Planlama">  
  <rdf_:condition></rdf_:condition>  
  <rdf_:provides rdf:kaynak=  
    "&rdf_;OSSD_Process_Model_00088"/>  
  <rdf_:next_control_flow rdf:kaynak=  
    "&rdf_;OSSD_Process_Model_00108"/>  
</rdf_:Faaliyet>  
<rdf_:Kaynak rdf:about="&rdf_;OSSD_Process_Model_00088"  
  rdf_:name="Project Plan"  
  rdfs:label="Project Plan">  
  <rdf_:required_by rdf:faaliyet=  
    "&rdf_;OSSD_Process_Model_00108"/>  
</rdf_:Kaynak>
```

```
<rdfs:Class rdf:about="&rdf_;Faaliyet"  
  rdfs:comment=""  
  rdfs:label="Faaliyet">  
  <rdfs:subClassOf rdf:kaynak="&rdfs;Kaynak"/>  
</rdfs:Class>  
<rdfs:Class rdf:about="&rdf_;Ajan"  
  rdfs:comment=""  
  rdfs:label="Ajan">  
  <rdfs:subClassOf rdf:kaynak="&rdfs;Kaynak"/>  
</rdfs:Class>  
<rdfs:Class rdf:about="&rdf_;Kaynak"  
  rdfs:comment=""  
  rdfs:label="Kaynak">  
  <rdfs:subClassOf rdf:kaynak="&rdfs;Kaynak"/>  
</rdfs:Class>  
<rdf:Property rdf:about="&rdf_;acts_on"  
  rdfs:comment=""  
  rdfs:label="acts on">  
  <rdfs:range rdf:kaynak="&rdf_;Faaliyet"/>  
  <rdfs:domain rdf:kaynak="&rdf_;Ajan"/>  
</rdf:Property>
```




SONUÇLAR

- Yazılım mühendisliği yönetim sürecinin sahip olduğu veriler tanımlanmıştır.
- Bu verilerin diğer süreç verileri ile ilişkileri modellenmiştir.
- İlişkisel modeller elde edilirken somut süreç modeli ve yazılım mühendisliği taksonomisi kullanılmıştır.
- Yazılım mühendisliği yönetim süreci üst-veri modeli tanımlanmıştır.
- Yazılım yönetim süreci üst-veri modeli **ManagemOnt** verisi olarak kullanılmıştır.



SONUÇLAR

- **ManagemOnt** ile yazılım yönetim sürecine ait üst-verilerden bilgiye erişilebileceği düşünülmektedir.
- **ManagemOnt**'un sağlayacağı bilgiler yolu ile benzer ürün ya da proje geliştirme süreçlerinin dinamik olarak modellenebileceği düşünülmektedir.
- Varolan iş akış ve belgeleme yönetimi modellerinden farklı olarak *doğru veri-doğru bilgi* yerine elde veri olmasa da doğru ya da doğruya yakın bilgiye erişebilmenin mümkün olduğu düşünülmektedir.
- **Ulu, B. ve Diri, B., Software Process Ontology, International MultiConference of Engineers and Computer Scientists 2007, IMECS 2007, 21-23 Mart 2007, Hong Kong, sf. 1110-1115, ISBN: 978-988-98671-4-0**
- **Ulu, B. ve Diri, B., Yazılım Yönetim Ontolojisi, 3. Ulusal Yazılım Mühendisliği Sempozyumu, UYMS 2007, 27-30 Eylül 2007, Ankara, sf. 103-108, ISBN: 978-9944-89-337-4**



TEŞEKKÜRLER

