

## ETMEN TABANLI SİMÜLASYON ONTOLOJİSİ

**M. Fatih Hocaođlu** <sup>(a, b)</sup>

<sup>(a)</sup> İstanbul Medeniyet Üniversitesi, Mühendislik ve Mimarlık Fakültesi, İstanbul  
mfatih.hocaoglu@medeniyet.edu.tr

<sup>(b)</sup> Agena Bilişim ve Savunma Teknolojileri, 34912 İstanbul Teknopark /İstanbul  
hocaoglu@agenabst.com

### ÖZ

Bu çalışmada, etmenlerin simülasyonda var olmalarının ontolojik arkaplanı incelenirken, EtSiS'in etmen davranışına verdiği cevap irdelenmiştir. EtSiS'in kozmolojisi ve ontoloji tanımı ele alınmış, ontolojisini oluşturan her bir paradigmanın davranış yönetimi ile olan ilişkileri incelenmiştir. Muhakeme mekanizması, varlık sorgusu ve dual dünya tasvirinin davranış yönetimi ile olan ilişkileri bir çözüm yaklaşımı olarak sunulmuştur.

**Anahtar Kelimeler:** Etmen, Etmen güdümlü simülasyon, EtSiS, Kozmoloji, Ontoloji

## AGENT DRIVEN SIMULATION ONTOLOGY

### ABSTRACT

In this study, while the fact that ontological background of accommodating agents in simulation studies is examined, the answer given by AdSiF- Agent driven Simulation Framework to the question is scrutinized. The cosmology that AdSiF lays down and AdSiF's ontological commitment and the relations of the paradigms constituting the ontology with behavioral management are examined and also the connection of reasoning mechanism, acquisitions for existence, and dual world representation with behavioral management are provided as a solution approach to agent driven simulation.

**Keywords:** AdSiF, Agents, Agent driven simulation, Cosmology, Ontology

## 1. GİRİŞ

Etmen tabanlı Simülasyon Sistemi (EtSiS) içerdği çoklu programlama paradigmasını kuşatan ve tümleştiren durum tabanlı programlama paradigması ile ontoloji tabanlı modelleme için bir modelleme dili sunmaktadır. Dil ile ontoloji tabanlı modellemenin ana bileşenlerinden “Bu nedir?” sorusuna, mantık programlama ile ilişkilendirilmiş ve farklı cevap çözümlüğü kademeleri için davranış yapılarının tanımlandığı bir çözüm sunulurken, bir diğer önemli bileşen olan, varlık ilişkileri için doğrudan davranış yapıları tanımlanmasına imkân veren bir ontoloji betimlemesi sunulmuştur.

Varoluşa ilişkin sorulara verilen cevaplar mantık programlama tabanlı olarak formüle edilmiş ve model davranış yapıları ile tümleştirilmiştir. Çözüm aynı zamanda varlığın geçmişine dönük kararların ve ilgili davranış modellerinin gerçekleştirilmesi için bir programlama ortamı sunmaktadır. Bununla, özellikle zaman gecikimli sistemlerde modelleme ve davranış yönetimi için bir dil ve ontoloji desteği sağlanmıştır. Çalışma hareket noktası olan EtSiS ontolojik betimlemesini, varlık felsefesi ile olan ilişkilendirilmesini, ortaya konulan dil tanımındaki karşılıklarını ve farklı programlama paradigmasını tümleştirmiş olduğu durum tabanlı programlama paradigmasının ontoloji arkaplanını içerir. Ontolojik betimlemenin dil grameri ile tümleştirilmesi, yalnızca modelleme düzeyinde kalmayan, aynı zamanda işletim aşamasına kadar uzayan, etkin ontoloji tabanlı bir modelleme ve işletim aracı tanımlamıştır.

Etmen tabanlı modelleme ontoloji tanımı üç temel kavramın varlık davranış yönetimleri ile ilişkilendirilmesi üzerinde tanımlanmıştır. Bunlar, varoluş tanımına dönük, ne olduğu sorusuna alınan cevaplar, varlıkların aralarındaki kurulan ilişki tanımları ve dünya tasviri ve muhakeme yapıları olarak belirlenir. Bu çalışma ayrıca, bir etmenin ontolojisinde varolan amaç güdümü ve reaktif olma özelliklerinin nasıl modellendiğine ve ontoloji kavramının temel soruları ile nasıl ilişkilendirileceğine eğilmiştir.

Makalenin takip eden bölümlerinde ontoloji tanımı ve etmen tabanlı simülasyonun temel aldığı kavramların, uygulama karşılıkları ele alınacaktır.

## 2. Ontoloji

Ontoloji konusundaki felsefi yaklaşımlar iki temel soru etrafında şekillendiğini görmekteyiz [1]. Bunlardan ilki bir varlığa dönük “nedir?” [2] sorusu ve ikinci kısım ise, varlığın temel özelliklerine ve ilintili olarak ilişkilerinin sorgulanması olarak karşımıza çıkmaktadır [3]. Varlık tanımı, varlığın tanımlayıcı özellikleri ve ilişkileri kavramsal model ve kavramsal modelin dayandırıldığı paradigmlar ile sıkı bir şekilde ilişkilidir. Kavramsal model kurgusunda varlıkların farklı paradigma bakışları ile yorumlanması mümkündür. Bu bakış ile etmen tabanlı modellemede varlık betimlemesinin birden fazla paradigmayı kuşatan bir tanımlama bakışı ile ele alınması tasvirin esnekliğine ve tasarım güçlenmesine destek verecektir. Çalışmanın merkezini oluşturan EtSiS’in kavramsal model tanımlamasında varlık ontolojisi birden fazla

paradigma ile gerçekleştirilirken, tüm paradigmalardan tümlenildiği bir paradigma ve paradigmaya ait bir dil ile tanımlanmaları sağlanmıştır [4], [5]. Farklı paradigmalardan birleştirilmesi bir ontoloji metriği olan ifade zenginliğini [1] güçlendiren bir özelliktir.

Simülasyonda bir varlık zaman eksenini, olay etkileşimleri, durumlar, durumlar ile oluşturulmuş olan davranışlar, bir muhakeme mekanizması ve muhakeme mekanizmasının girdilerini oluşturan bilgi yapılarına sahiptir. EtSiS’de varlığın semantik yapısı davranış yapıları ile temsil edilirler. Bir etmen programlama, simülasyon ve hesaplama altyapısı olarak EtSiS, iyi tanımlı bir gramere sahiptir ve gerek dil yapısından gerekse de ontolojik betimlemesinden kaynaklı bir seri tasarım kuralları, bir programlama paradigması ve tasarım şablonları sunar.

### **3. ETMEN TABANLI SİMÜLASYON ONTOLOJİSİ**

Etmen tabanlı Simülasyon sistemi Sistemi (EtSiS) ontolojik betimlemesi aşağıdaki gibi ifade edilir.

“Bir varlık kendisini diğer varlıklardan ayırt eden özniteliklere ve bu özniteliklerin değerlerini yöneten eylemlere sahiptir. Varlığın yürüttüğü her eylem bir durum tanımı ile sarılırlar ve eylemler durumlar arası geçişler ile sağlanır. Durum geçişleri ve ilgili eylemler, anlamlı bir bütün oluşturacak şekilde yürütülürler ve bu anlamlı sıralama davranış olarak tanımlanır. Varlıklar bir çevre içerisinde bulunurlar ve diğer varlıklar ile olay etkileşiminde bulunurlar. Her varlık bulunduğu çevreyi algılar ve çevreye ilişkin bir iç tasvir oluşturur. Tasvir çevreye ait geçmiş bilgileri, doğru olduğu kabul edilen bilgileri ve bu bilgi yapısını esas alan karar yordamlarını içerir. Karar yordamları ile çevresi ile olan etkileşimleri yöneten ve kendi durumu tanımını belirleyen davranışları yönetir.”

Bir varlığı bir diğerinden ayırt eden öznitelikler, varlığın, ontolojinin temel sorularından biri olan, “Bu nedir?” sorusunun, bir soyutlama seviyesinde cevabı olan, bir üst modelden türemiş olabilmesi, sahip olduğu eylem ve özniteliklere erişim prosedürlerinin tanımlanabilmesi, ontoloji betimlemesinin nesne yönelimli arka planını oluşturmaktadır. EtSiS ontolojik betimlemesi mantık programlama, nesne yönelimli programlama, etmen tabanlı programlama ve cephe tabanlı programlama paradigmasını içerir ve tüm bu paradigmaları durum tabanlı programlama paradigmasında tümlendirir. Durum tabanlı programlama paradigması varlıklara ilişkin, durumları ve davranışlarına göre bir betimleme, bir sınıflandırma gerçekleştirir. Paradigmanın temel kavramlarını, durumlar, davranışlar ve davranış cepheleri (aspects) oluşturur. Varlığın, bir etmen olarak, çevresini gözlemlemesi ve kendisi için tanımlı bilgi yapılarının bilgi tabanında zaman eksenli olarak güncellemesi ve genişletmesi, bilgi yapısını kullanan karar algoritmaları ile ürettiği karar sonuçlarını kullanarak davranışlarını yönetmesi, mantık programlamanın durum tabanlı programlama ile olan tümleniminin ana eksenini oluşturur.

Bir etmenin davranış yapısı ardışık durumlar ile oluşturulmuş davranışlar ve durum geçişleri ile yürütülen eylemler ile temsil edilirler. Bir davranışın yönetilmesi aşağıdaki faz geçişlerinin yönetilmesi ile ilgilidir, bunlar; 1) Aktive edilmesi, 2) İptal edilmesi, 3) Askıya alınması, 4) Tekrar işletilmesi ve 5) Tamamlanması olarak tanımlanır. Bir davranış olay etkileşimi ile aktive edilebilir. Aktive edilme faz geçişi dahil diğer tüm faz geçişleri bir başka davranış tarafından yürütülebilir ve etmenin bulunduğu ortam durum tanımlamalarına göre tanımlanmış şartların sağlanması ile sağlanabilir.

Cephe tabanlı programlama etmenin bulunduğu çevre veya kendi şartına bağlı olarak davranış karakteristiğini değiştirebilme yeteneğini tanımlar [6] ve davranışların farklı şart ve duruma göre kategorik olarak aktive edilmelerini yönetir. Bu şart değişimlerine bağlı olarak aktive edilen davranışlar kümesi etmenin bir davranış cephesini sergiler.

Bildirimsel programlama yaklaşımı ile EtSiS insan benzeri problem çözme yaklaşımının modellenmesinde önemli bir çözüm sağlamaktadır. Problem çözümünde yaklaşım bir varlığın ve ilintili olarak, varlığa ait bilgi yapılarının, ilişkilerinin, kısacası, ontoloji tanımının doğrudan bir algoritmik rutinler olarak değil, problemin, çözümün ve problemi çözecek olan varlığın tanımlanması tabanlı bir çözüm arayışı mevcuttur. Bu yapı mantık programlama yaklaşımı ile tam olarak örtüşür ve etmenleri tepki üreten varlıklar olmanın ötesine taşıyarak, bir amaç tanımlama ve amacı gütmeye yeteneği kazanmasına sahip olmasını sağlar. Bir etmenin amacının olması, etmen yazılımlarını yalnızca bir insan benzeri yazılım olmaktan öte, bir işi olan insan benzeri olmalarını sağlar. Amaç güdümü ve strateji uygulama etmenleri, klasik yazılımlardan daha fazla insan benzeri davranışa yaklaştırır ve otonom, reaktif olma özellikleri bu özelliği güçlendirir. Klasik yaklaşım ile geliştirilen bir yazılım, tanımlı prosedürleri takip ederek işlevi sonlandırır. Bir etmen ise, bir çevre içerisinde bulunduğu için çevresi var olduğu sürece, yürütecek bir amacı bulunmasa veya başarılı olmuş olsa bile, kendisi için tanımlı olan olaylara ve durum değişimlerine reaksiyon gösterir. Çevresinde oluşan olaylara veya durum değişimlerine gösterilen reaksiyon (reaktif olma özelliği) davranış olarak modellenir ve davranış iletimi sonucunda durumsal farklılık (dolaylı etkileşim) ve/veya olay yayılması (doğrudan etkileşim) gerçekleşir [7].

#### **4. Kozmoloji: Zaman ve Evren**

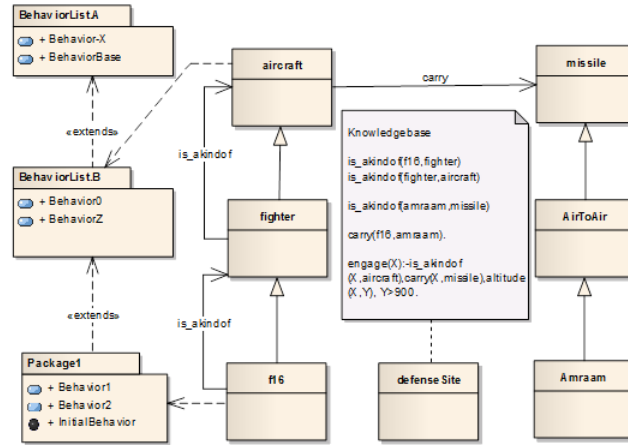
Zaman kavramı EtSiS'in etmen tabanlı ontoloji betimlemesinde özel bir yere sahiptir. Zaman kurgulanan kavramsal model bazında, tüm kavramsal model varlıkları tarafından oluşturulan bir boyuttur ve kavramsal dünyaya, Euclidean zaman tanımı gibi, bir varsayılan bilgi olarak dışardan giren bir bilgi değildir. Zaman kavramsal dünyayı oluşturan varlıkların devinimi ile göreceli olarak belirlenen bir bileşendir ve her kavramsal model, boş bir uzay ile başlar. Uzaya dâhil edilen bir varlığın durum geçişi ve bağlantılı olarak bir olay etkileşimi ile gerçekleştirdiği devinim uzayda zaman boyutunun oluştuğu zaman ve uzay tümleşik tanımına ulaşırız. Tanımlama izafiyet

teorisinin temelini oluşturan zaman tanımlamasıdır. EtSiS kavramsal uzayında varlıklar eylemlerini zamanın ilerleyişi ile gerçekleştirmezler, eylemlerle uzaydaki zamanın oluşumu sağlanır. Bu bakış ile, EtSiS zaman tanımının, izafi zaman teorisi esaslı olarak tanımlandığını söyleyebiliriz. Uzayda belirli bir durum tanımı içerisinde sonsuz zaman için bekleyen varlık veya varlıklar için bir ilk zamandan, şimdiki zamandan veya önceki zamandan bahsetmek mümkün olmaz. Bir zaman noktasının oluşumu ancak ve ancak varlıkların bir devinim göstermesine bağlıdır.

Bu kozmolojik tanımlama bizi, sürekli ve kesikli olay tasvirinde önemli bir tümlenmeye taşır. Durum geçişleri uzayda bir olay oluşumu ile açıklanabilir ve tersine bir yorum ile olayların oluşumları varlıklarda durum geçişlerini tetikler. Bu iki yönlü bakış ile, varlığın kendi içindeki devinimi sebebiyle gerçekleştirdiği durum geçişleri, uzayda bir olaya karşılık gelirken (gerçek veya sanal), algılanan her bir olay ilgili varlıklarda durum geçişlerine sebep olur, -ki durum geçişleri durum tabanlı programlamada eylemlere ilişkin fazları içerir. Bir varlıktan bir eylem isteğinde bulunmayan sanal olaylar uzayda sadece zaman noktasına ilişkin tanımlama oluştururlar ve varlığın kendi iç dinamiği ile gerçekleştirdiği bir durum vektörü değişimi sözkonusudur. Bu bakış ile EtSiS ontolojisi sürekli veya kesikli olay tanımlamasından bağımsız olarak, her ikisini de kuşatan, bir üst zaman ve olay tanımı gerçekleştirir ve bu tanımlama ile sürekli olay düzenli durum geçişlerini evrende zaman noktalarına resmederken, evrende gerçekleştirilen zaman noktalarını kesikli zaman noktaları olarak tanımlar.

## 5. VARLIK SORGUSU İLE DAVRANIŞ YÖNETİMİ

Ontolojinin temel sorusu olan “Bu nedir?” sorusuna alınan her bir cevap, varlığın tanımlamasını bir üst soyutlama seviyesine taşır ve taşınan her bir soyutlama seviyesi için tanımlı bilgi yapıları, karar yordamları ve ilişkili üst seviye davranış tanımlamaları mevcuttur. Şekilden de görülebileceği gibi bir F16 için sorulan “Bu nedir?” sorusuna alınan cevaplar, modelin daha üst bir varlık tanımı ile tanımlanması ve her üst varlık tanımına daha alt seviye varlık davranış tanımında da geçerli olan, davranış tanımlamaları ile ilişkilendirilmiş olduğu görülmektedir. F16’nın ne olduğuna verilen ilk cevap, onun bir savaş uçağı (fighter) olduğu, savaş uçağının ne olduğuna verilen cevap ise onun bir hava aracı olduğu şeklindedir. Hava araçları seviyesinde tanımlanan *BehaviorList.B* davranış kümesi hava aracından türeyen tüm varlık tipleri için geçerlidir ve tüm hava araçları için tanımlı olan temel davranışları içerir. Benzer bakışla *Package1* isimli davranış kümesi F16 için tanımlanmıştır ve hava aracı seviyesinden alınan davranışları genişletir. Tanımlama modelin durumsal tanımlamalarının da daha üst soyutlama düzeyinde yapılabildiğini gösterir.



Şekil 1: Varlık Sorgusu ile Davranış Güdümü

## 6. AMAÇ GÜDÜMÜ VE TEPKİSEL DAVRANIŞLAR

Bir etmen olarak tasarlanan simülasyon modelleri bilgi tabanında güncellenen bilgi yapıları ve muhakeme yordamları (kurallar) ile davranışlarını belirli bir hedefi sağlayacak şekilde yönlendirirler. Bir amaç başarımlı şartı, iptal şartı, askıya alınma şartı, tekrar işletilme şartı ve bir grup seri ve paralel işletilen eylemler olarak tanımlanır. Eylemler serisi amacı sağlayacak bir plan olarak belirlenir ve planlama etmeni [8] olarak tanımlanan etmen tasarımı, her eylem sonucunda amaca yaklaşma derecesini ölçerek, alternatif planları değerlendirir. Değerlendirme ve planlama eylemleri mantık programlama paradigmasının etmen görev işletiminde üstlendiği bir görev parçasıdır. Bu aşamada EtSiS etmen tasarımı durum tabanlı programlama yaklaşımı ile, atomik çözünürlüğü belirleyen, her bir durum geçişi amaç değerlendirmesi ve planlama için iyi tanımlı bir faz tanımlamıştır [9]. Görevin yürütülmesi esnasında çevresinde gerçekleşen ve kendisini ilgilendiren olaylara karşı tepkisel davranışlar üretir ve bu davranışlar eğer yürütmekte olduğu göreve ilişkin davranışlar ile çelişmiyor ise paralel olarak yürütülürler. Bir çelişki olması durumu yeni bir planlamayı veya görev davranışlarını askıya alarak uygulama şartlarının tekrar sağlanmasını takiben (tepki davranışlarının işletiminin tamamlanması ile) işletilmelerine devam edilir. Bir amaç güdümü için genel amaçlı bir planlama davranış şablonu tasarlanabilir. Bu özellik varlık ontoloji tanımlamasında yer alan birden fazla paradigmayı tümleştiren durum tabanlı dil gramerinden gelmektedir. Tüm görev yapıları ve ilişkili şartlar (Activation, cancel, suspend, resume ve finish) EtSiS'in davranış gramerine uygun olarak gerçekleştirilir.

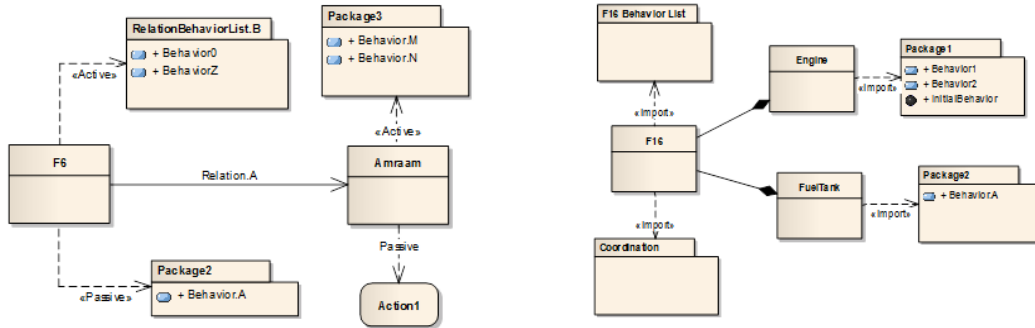
## 7. MUHAKEME İLE DAVRANIŞ YÖNETİMİ

Muhakeme bileşenleri çevreye ilişkin sahip olunan bilgiler (beliefs) ve önerme tanımlamaları (predicates) olarak ele alınır. Muhakeme önermelerin doğruluk değerlerinin ve/veya önerme çıktı parametrelerinin değerlerinin mantıksal veya matematiksel operatörler ile kullanılarak bir doğruluk değeri üretmesi ve bu doğruluk

değerlerinin davranış güdümünde istihdamı sağlanır. Varlık bilgilerine bir örnek Şekil 1’de görülmektedir. Bilgi yapısı fl6’nın bir savaş uçağı, savaş uçağının bir hava aracı olduğu bilgisine (bu nedir sorusuna verilen cevaplar ile oluşturulmuş olan bilgi yapısı) ek olarak fl6’nın ilişkide olduğu füze tipi (*carry(fl6, amraam)* – F16 Amraam füzesi taşımaktadır. Amraam füzesi ise bir tür havadan havaya füze olarak tanımlanmıştır. Burada ilişki *carry* olarak tanımlanmıştır) bilgi tabanında yer almakta ve bilgi yapısına dayalı olarak tanımlanmış angajman emri kuralı görülmektedir. Söz konusu bilgi yapısı yere konuşlu savunma unsuru modeline (*defenseSite*) ait bir bilgi tabanı, diğer bir ifade ile, modele ait bir iç dual dünya tasviridir. Bilgi tabanı çevreden sensörler aracılığıyla veya önceden tanımlı bilgiler ile (*beliefs*) oluşturulur ve sensör bilgisinin doğruluk düzeyine bağlı olarak ve çıkarım mekanizması doğruluğuna bağlı olarak oluşturulur. Şekilde görülen kural “*Amraam füzesi taşıyan bir hava unsuruna angajman emri verilmesini içerir.*” Emir verilmesine ilişkin kuralın doğruluk değeri angajman davranışını tetikleyen güdüm koşulu olarak ilişkilendirilir.

## 8. İLİŞKİ KAVRAMI VE DAVRANIŞ YÖNETİMİ

Varlık ontolojisi, ilişki kavramını tümleşen, durumsal ve eylem ilişkisi olarak üç ilişki kategorisi altında ele alır. İki varlık arasında kurulan durumsal ve eylemsel ilişkiler, yönlüdür ve kurulma ve koparılma (kaldırılma) fazlarından oluşur. İlişkiler statik olarak tanımlanan ve değişmez yapılar değildir, bu sebeple, kurulma ve koparılma fazları varlıkların yaşam döngüsü içerisinde, defalarca gerçekleşebilir [3], [9], [10]. Şekil 2 (a)’da görüldüğü üzere, *Relation.A* isimli ilişkinin kurulması durumunda *RelationBehaviorList.B* isimli davranış seti aktive edilmektedir. Koparılması durumunda ise *Package2* isimli davranış kümesi yürütülmektedir. Benzer tanımlama ilişkinin diğer tarafında yer alan varlık için de yapılmıştır. İlişkinin kurulması durumunda *Package3* isimli davranış kümesi aktive edilmekte, koparılması durumunda ise *Action1* isimli atomik fonksiyon yürütülmektedir. Tanımlama görüldüğü gibi belirtimsel yapıdadır ve doğrudan ontoloji tanımının uygulamaya olan yansımasıdır.



(a)

(b)

Şekil 2: İlişki ve Tümleşik Model Tanımlama

Bir varlık birden fazla varlığın tümleşimi ile oluşabilir. Bu durumda birden fazla varlığı içeren ve bir anlamda tümleşik varlığı temsil eden varlık, iç yapısında içerdiği varlıkları ayrı olarak muhafaza etse de, onların dış dünya ile olan etkileşimlerini üstlenir. **Hata! Başvuru kaynağı bulunamadı.**'de iç modellerin davranış yapılarını, bilgi tabanlarını ve ilişkili kural setlerini muhafaza ettikleri, olay dağıtımlarını ve zaman koordinasyonlarını iç modelleri çevreleyen, tümleşik model üstlenir. Şekil 2(b)'de yakıt tankı ve motorun ayrı modeller olarak tanımlanarak, f16 modeli ile tümleştirilmiş bir tanımlama görülmektedir. Motor ve yakıt tankı modelleri kendi davranış yapılarına sahiptirler fakat dış dünya ile etkileşimleri, bağlantılı olarak zaman yönetimleri F16 modeli tarafından yürütülmektedir. Bu işlemler F16 modeli davranış kümesi, *F16BehaviorList*'i genişleten *Coordination* davranış kümesi ile yürütülmektedir.

## 9. Diğer Ontoloji Dilleri ile Kıyaslama

AdSiF'in sunmuş olduğu modelleme ve simülasyon dil yeteneği ile ontoloji tabanlı modelleme yaklaşımı diğer, simülasyon paket ve dilleri ile ve ontoloji geliştirme metodolojileriyle değerlendirilmiştir. Değerlendirme kriterleri ve yorumlar aşağıdaki tablolarda ve tablo dipnotlarında kısaca özetlenmiştir .

**Tablo 1: Ontoloji Geliştirme Metodolojileri**

	DUJET	OIEd	Onto Edit	Ontolingua	OntoSaurus	Prote'ge'	AdSiF
Mimari	P1	S	S,C	C	C	S	x2
Genişletilebilirlik	-	P	P	-	-	P	x3
Ontoloji depolama	-	D4	D/Vt	D	D	D, Vt	D, Vt
Bilgi Gösterimi	NYP Sınıf	DL(DAM+OIL)	F+FOL	F+FOL	DL (LOOM)	F+FOL+ MC5	SA+F OL+F
Metodoloji	R	-	O	-	-	-	EA
Muhakeme mekanizması	Hayır	FaCT	Ob	Hayır	Evet	PAL	x6
İşletilebilirlik	-	-	-	-	-	-	X

R: Rational Rose, EA: Enterprise Architect, O: Onto-Knowledge, Ob: Onto-Broker

**Tablo 2: Simülasyon Paket/Dilleri**

	DEVS	Siman	Mozart/Oz	AdSiF
Ontoloji	Yok	Yok	Yok	X
Zaman yönetimi	X	X	Yok	X <sup>7</sup>
Olay Yönetimi	X	X	Yok	X <sup>8</sup>

<sup>1</sup> P: Yazılım uzantısı (plugin), S: Standalone, C: Client/server

<sup>2</sup> Betik Dil, Plugin, Olay güdümlü, Katmanlı

<sup>3</sup> İşletim zamanlı ve Tasarım zamanlı

<sup>4</sup> D: Dosya, Vt: Veritabanı

<sup>5</sup> F: Frames, FOL: First Order Logic, MC: Meta class, SA: Durum otomatı

<sup>6</sup> Predicate logic, First order logic, Higher order logic

<sup>7</sup> Mevcut zaman yönetiminden farklı senkronizasyon ve zaman yönetim algoritmaları AdSiF'in programlama alt yapısı olan durum tabanlı programlama ile genişletilebilir. Farklı algoritmalar işletim zamanlı olarak değiştirilerek kullanılabilir. Değişiklik kararı bir yapay zeka algoritma ile muhakeme sonucunda ele edilebilir.



USMOS 2015 ODTÜ, ANKARA

	DEVS	Siman	Mozart/Oz	AdSiF
Tümleşik Kavramsal Model Desteği	Yok	Yok	Yok	X <sup>9</sup>
Senaryo Doğrulama	Yok	Yok	Yok	X
Kavramsal model doğrulama	Yok	Yok	Yok	X
Yığın koşumlar	Yok	Yok	Yok	X
Yinelemeler	Yok	Yok	Yok	X
Desteklenen Paradigmalar	NYP	NYP	NYP	NYP, CtP, MP, EtP <sup>10</sup>
Sahip olduğu paradigma	x <sup>11</sup>	x <sup>12</sup>	x <sup>12</sup>	SOP
Model tabanlı Simülasyon Yönetimi	Yok	Yok	Mümkün <sup>13</sup>	X <sup>14</sup>
Rastsal Sayı	X	X	X <sup>kodlama düzeyinde</sup>	X <sup>15</sup>
Kaynak Kod	Yok	Yok	Yok	Var
iletişim	Model Kimliğine	Varlık iletimi	Mesajlaşma	X <sup>16</sup>
Soyutlama	Gen, Agg, Ass	Yok	Gen, Agg, Ass	Cla, Gen, Agg, Ass <sup>17</sup>
İşletim zamanlı genişletme	Yok	Yok	Yok	x <sup>18</sup>
Açıklık (expressiveness)	x <sup>19</sup>	sınırlı	x <sup>19</sup>	x <sup>20</sup>
Basitlik	x <sup>21</sup>	x <sup>22</sup>	x <sup>23</sup>	x <sup>24</sup>
Tamlık (Preciseness)	X	Hayır	X	X
Genişletilebilirlik	Java ve c++	C ve Fortran	Plugin	x <sup>25</sup>
İşletim zamanlı genişletme	Yok	Yok	Yok	x <sup>26</sup>
Doğrulama	Çevrim dışı	Çevrim dışı	Çevrim dışı	x <sup>27</sup>
Dil entegrasyonları	C++/Java	C/Fortran		C++, Piton, Prolog
Miras mekanizması	X <sup>sınıf düzeyinde</sup>	Yok	X	x <sup>28</sup>
Modülerlik	X -İyi	Yok	X-İyi	X- Yüksek <sup>29</sup>

<sup>8</sup> Mevcut algoritmaya ek olarak farklı olay seçim mekanizmaları yürütülebilir, farklı olay kuyruk seçim kuralları işletilebilir. Zaman yönetiminde olduğu gibi durum tabanlı programlama ile olay yönetim algoritmaları bir katman olarak geliştirilebilir.

<sup>9</sup> Case araçları ile sağlanan geniş bir süreç desteği mevcut

<sup>10</sup> NYP: Nesne Yönelimli Programlama, CtP: Cephe tabanlı Programlama, MP: Mantık Programlama, EtP: Etmen tabanlı Programlama

<sup>11</sup> Durumların programlanması

<sup>12</sup> Kendine ait yok

<sup>13</sup> Mümkün fakat hataya açıklık yüksek

<sup>14</sup> Modeller ile simülasyon koşum yönetimi mümkün

<sup>15</sup> Kütüphane mevcut, Plugin mimarisi ile genişletilebilir. Hesaplamalar model geliştirme ile betik düzeyinde tanımlı

<sup>16</sup> Model kimliği ile mesajlaşma, ilişkili varlıklarla mesajlaşma, Model tipi ile mesajlaşma

<sup>17</sup> Cla: Classification, Gen: Generalization, Agg: Aggregation, Ass: Association

<sup>18</sup> Fonksiyon ekleme, Öznitelik ekleme, Davranış ekleme, Yapay zeka alg. Ekleme

<sup>19</sup> Standart programlama özelliklerine sahip.

<sup>20</sup> Grafik programlamaya imkan vermesi sebebiyle görsel programlama imkanı sunar. Gelişmiş davranış durum diyagramları ile güçlü programlama ortamı sağlar

<sup>21</sup> Sunulan temel sınıf fonksiyonlarına erişim

<sup>22</sup> Keyword tabanlı programlama (sınırlı)

<sup>23</sup> Özel bir programlama dilidir

<sup>24</sup> Betik dil, modüler model geliştirme, gevşek bağımlı programlama

<sup>25</sup> C++ plugin, Piton arayüz imkanı

<sup>26</sup> Fonksiyon ekleme, Öznitelik ekleme, Davranış ekleme, Yapay zeka alg. ekleme

<sup>27</sup> Koşum zamanlı analiz ve doğrulama

<sup>28</sup> Sınıf, model ve Kütüphane düzeyli. Model iç tanımında durum, davranış ve davranış listeleri, kütüphane düzeyinde olay ve ilişki tanımlama seviyelerinde yüksek seviye miras mekanizması tanımlıdır.

<sup>29</sup> Model ve alt model düzeyinde ayrık ve zayıf bağımlılık

	DEVS	Siman	Mozart/Oz	AdSiF
Overloading	Sınıf seviyesinde	Yok	Sınıf ve fonk.	x <sup>30</sup>
Bilgi saklama	Sınıf seviyesinde	Yok	x <sup>31</sup>	x <sup>32</sup>
Hata yönetimi	Kullanıcıya bağlı	Sınırlı	Kullanıcıya bağlı	x <sup>33</sup>
Genel Programlama	Evet	Yok	Evet	Evet
Programlama Paradigmaları	NYP	Nesne tabanlı	NYP, Fonk.	x <sup>34</sup>
Ortogonalite	x <sup>35</sup>	Yok	x <sup>35</sup>	Yüksek Destek
Özyinelemeli programlama	Yok	Yok	Var	x <sup>36</sup>
İşletim Sistemi	Windows	Windows	Windows, Linux	x <sup>37</sup>
Yazılım Arayüzleri	Var	Var	Var	Gelişmiş plugin
Hataya Açıklık	x <sup>38</sup>	Açık	x <sup>39</sup>	x <sup>40</sup>
YM Süreci	OOP	Yok	Yok	x <sup>41</sup>
Zaman gecikimli sistem modelleme	Yok	Yok	Yok	Var
Matlab Entegrasyonu	Yok	Yok	Yok	Var
Koşum hızı regülasyonu	Yok	Yok	Yok	Var
Tasarım şablonu	NYP şablonları	Yok	Yok	x <sup>42</sup>

## 10. SONUÇ

EtSiS kavramsal model süreci boş bir uzay ile başlar. Zaman eksenini tanımlama sürekli ve kesikli olay sistemlerinin her ikisini de kuşatacak şekilde, uzay-zaman (spacetime) göre yapan bir kozmolojiye sahiptir. Simülasyon dünyası boş bir uzay, zaman eksenini tanımlama ve varlık tanımlaması ile başlatılır. Tanımlamanın bu kadar atomik düzeyde başlatılması uygulama sahasından ve pratik durumların sınırlandırmalarından uzaklaşmış bir modelleme çözümü elde etmemizle sonuçlanır.

Ontoloji tabanlı modellemede bir varlık ontolojisi tanımlamanın ötesinde, tanımlama bileşenlerinin her biri, varlık için sorulan “bu nedir?” sorusuna alınan cevapların, varlıklar arası ilişkilerin, varlık durum tanımlarının ve varlığın dual dünya tasvirlerinin davranış yapıları ile ilişkilendirmesi kuşatıcı bir çözüm yaklaşımı sunmaktadır.

Modelleme yaklaşımının temelini oluşturan çoklu programlama paradigması ve programlama paradigmalarının arka planındaki dünya tasvirleri, modelleme ve dünya betimlemesi zenginliği sunmaktadır. Nesne yönelimli programlama, mantık

<sup>30</sup> Davranış, durum ve model seviyesinde + sınıf seviyesinde

<sup>31</sup> Dil tanımları ile belirlenmiş öznitelik ve fonksiyon sınırları

<sup>32</sup> Sınıf yapılarında NYP’den gelen bilgi saklamaya ek olarak, model seviyesinde öznitelik ve davranış erişimleri erişim tip (public, protected, private) tanımlamaları ile sınırlandırılır.

<sup>33</sup> Hata logları, Koşum logları, Koşum akış görselleştirme

<sup>34</sup> NYP, Cephe (Aspect Oriented), mantık programlama, etmen tabanlı programlama, Durum tabanlı programlama

<sup>35</sup> Tasarımcı yeteneğine bağlı özel bir destek yok

<sup>36</sup> Davranışlar ve mantık programlama önerme yapıları kullanımı ile sağlanır.

<sup>37</sup> Çekirdek yorumlayıcı linux’da derlenmiştir.

<sup>38</sup> Temel fonksiyonların kullanımı ile kod geliştirildiği için hataya açıktır.

<sup>39</sup> Programlama yapıldığı için hataya açıktır.

<sup>40</sup> Atomik seviyeye indirgenmiş programlama, soyut davranış yapıları ve iyi tanımlanmış bir geliştirme süreci, araç desteği ile en aza indirgenmiştir.

<sup>41</sup> Tanımlı Süreci var ve CASE araçlarında uygulanmıştır.

<sup>42</sup> Paradigmasına ait şablonlar, NYP şablonları

programlama, cephe programlama, etmen tabanlı programlama ve tüm bu paradigmaları birleştiren durum tabanlı programlama paradigması farklı dünya bakış şablonlarını birleştirerek, farklı bakışlar ile varlık modellemesini ve davranış yönetimini sağlar. Dünya tasviri ile oluşturulan bilgi tabanı ve ilişkili kural (predicate-önerme) yapılarından elde edilen kararlar ile davranışların yönetilmesi, benzer şekilde, varlık soyutluk seviyelerinin her birisine ilişkilendirilmiş davranış yapıları ve şartlara, durumlara bağlı varlıkların davranış cephelerini değiştirebilme, varlıklar arası ilişkilerin tanımlı fazları için eylem ve davranış aktivasyonları ontoloji tabanlı modellemede, ontoloji tanımından uygulamaya giden yol haritasını gösterir.

## 11. KAYNAKÇA

- [1] N. Guarino, D. Oberle, and S. Staab, "What is an Ontology?," S. Staab and R. Studer, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 1–17.
- [2] W. V. O. Quine, "On What There Is," *Rev. Metaphys.*, vol. 2, pp. 21–38, 1948.
- [3] M. F. Hocaoglu, "Conceptual Model and Perdurantist Modeling with Reasoning," *Simul. Notes Eur.*, vol. 24, no. 2, pp. 95–104, 2014.
- [4] M. F. Hocaoglu, "AdSiF: Agent Driven Simulation Framework," *Hunsv. Simul. Conf. - HSC2005*, 2005.
- [5] M. F. Hocaoglu, "AdSiF: Agent driven Simulation Framework," in *USMOS 2011- National Defense Application and Modeling & Simulation Conference -(in Turkish)*, 2011.
- [6] M. F. Hocaoglu, "Aspect Oriented Programming Perspective in Agent Programming," in *USMOS 2013- National Defense Application and Modeling & Simulation Conference -(in Turkish)*, 2013.
- [7] M. F. Hocaoglu, İ. Genç, and A. A. Ergin, "AdSiF: Agent Driven Simulation Framework and Model Based Validation and Verification," in *2. Yazılım Kalitesi ve Yazılım Geliştirme Araçları Sempozyumu 2010*, 2010.
- [8] P. N. S. Russel, *Artificial Intelligence: A Modern Approach*. Prentice Hall, 1995.
- [9] Mehmet F. HOCAOGLU, "Agent based Simulation: Lecturer Notes," AUZEF, Istanbul University, Istanbul, 2014.
- [10] M. F. Hocaoglu, "Conceptual Model and Perdurantist Modeling with Reasoning," *ASIM 2014, ASIM Symp. Simul. Tech. (ASIM SST)*, 2014.
- [11] X. Su and L. Ilebrekke, "A comparative study of ontology languages and tools," *Adv. Inf. Syst. Eng. 14th Int. Conf. CAiSE 2002*, pp. 761–765, 2002.