

SEAGENT ÇOKLU ETMEN GELİŞTİRME ÇERÇEVESİ

Oğuz Dikenelli

Bilgisayar Mühendisliği Bölümü, Ege Üniversitesi, İzmir
e-posta: oguz.dikenelli@ege.edu.tr

Özetçe

SEAGENT rol tabanlı çoklu etmen geliştirme çerçevesi, Ege Üniversitesi Çoklu Etmen Sistemleri Araştırma Grubu tarafından geliştirilmiştir. SEAGENT çoklu etmen geliştirme çerçevesini diğer çoklu etmen geliştirme çerçevelerine göre fark yaratan iki özelliği vardır. Birincisi etkileşimli geliştirim ortamına sahip olmasıdır. Diğeri ise anlamsal veb uyumlu olmasıdır. Bu çalışmada SEAGENT çoklu etmen geliştirme çerçevesini oluşturan temel bileşenler ve çerçevenin genel mimarisi ele alınarak detaylı olarak incelenmiştir.

1. Giriş

SEAGENT rol tabanlı çoklu-etmen geliştirme çerçevesi, Ege Üniversitesi Çoklu Etmen Sistemleri Araştırma Grubu tarafından geliştirilmiştir. SEAGENT projesi resmen 2002 yılında geliştirilmeye başlandı. Projenin başlangıcından 2006 yılına kadar evrilerek gelişen mimarisi daha sonra tüm araştırma grubu tarafından kabul görerek etkin olarak kullanılmaya başlandı.

SEAGENT çerçevesini [1] diğer çoklu etmen geliştirme çerçevelerine göre fark yaratan iki özelliği vardır. Birincisi etkileşimli geliştirim ortamına sahip olmasıdır. Bu geliştirim ortamı, çoklu-etmen sistemlerinin farklı bakış açılarından tümleşik olarak geliştirilmesini sağlar. Geliştirme ortamı, çoklu etmen sistemi geliştiricilerinin organizasyon, hedef, plan gibi farklı bakış açılarından çoklu etmen sistemlerini geliştirmelerine olanak sağlamaktadır.

SEAGENT çerçevesinin fark yaratan diğer özelliği ise Anlamsal Veb desteğidir. SEAGENT bu desteği iki boyutta sağlamaktadır. Birincisi SEAGENT etmenlerin organizasyon, hedef ve plan bilgileri gibi iç bilgilerini OWL ontolojileri kullanılarak temsil edilmektedir. Diğer yandan anlamsal veb servislerinin planlayıcı seviyesinde kullanılabilmesini destekler [2]. Anlamsal veb servislerinin OWL-S ontolojileri etmenin bilgi tabanında saklanır. Etmen dış anlamsal servisi çalıştırmak istediğinde planlayıcısı bu ontolojileri kullanarak servisleri çalıştırabilir.

Bildirinin geri kalanı şu şekilde düzenlenmiştir: 2. bölümde SEAGENT Çoklu etmen geliştirme çerçevesini oluşturan geliştirme ve çalışma ortamları açıklanmıştır. 2. Bölümün alt bölümlerinde ise Çalıştırma ve geliştirme ortamlarını oluşturan alt bileşenler, mimarileri, bileşenlerin aralarındaki etkileşimler ile detaylandırılmıştır.

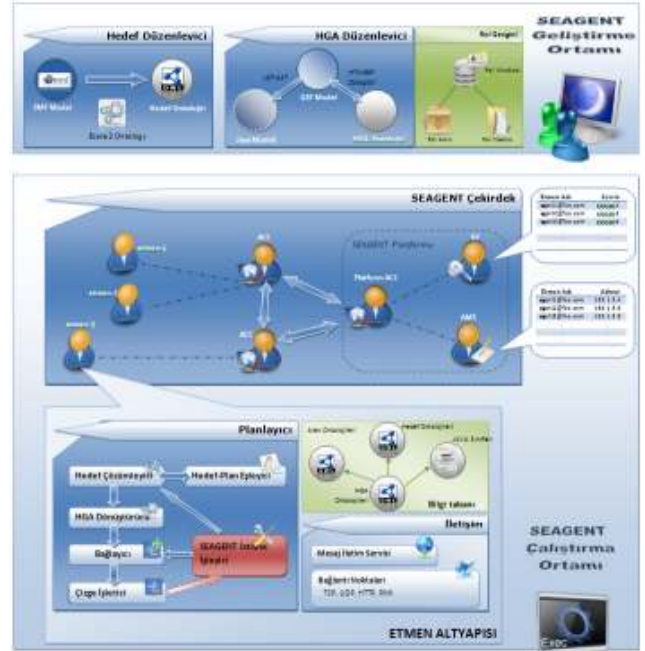
2. SEAGENT Çoklu Etmen Geliştirme Çerçevesi

SEAGENT Çoklu Etmen Geliştirme çerçevesi iki ortamın birleşiminden oluşmaktadır: SEAGENT geliştiricilerine

görsel geliştirim ortamı sağlayan Geliştirme Ortamı (SGO) ve altyapı elemanlarını içeren Çalıştırma Ortamı (SÇO). Bu ortamlar Şekil-1'de gösterilmektedir. Takip eden bölümlerde ise SEAGENT çerçevesini oluşturan elemanlar ve mimarisi detaylı olarak açıklanmaktadır.

2.1. SEAGENT Geliştirme Ortamı

Yazılım platformlarının kullanılabilirliğinin artması için yazılım geliştirilen ortamda çalıştırılacak olan yazılımın kullanıcı tarafından kolay üretilmesi gereklidir. Bu nedenle SEAGENT Çoklu Etmen Platformu'nda kolay yazılım geliştirmek için SEAGENT Geliştirme Ortamı (SGO) ortaya çıkarılmıştır. SGO, SEAGENT çoklu etmen platformuna girdi olarak verilecek ürünlerin geliştirilmesini ve yeniden kullanılabilirliğini sağlamayı hedeflemektedir. Bunlarla birlikte ortam bağımsız ve kolay geliştirim sağlanması için SGO, Eclipse tak-çıkart mimarisine uygun biçimde gerçekleştirilmiştir. Eclipse geliştirme ortamına tümleşik olarak gerçekleştirilen bu araçlar, görsel arayüzler tarafından desteklenerek kullanıcıların daha rahat yazılım geliştirmelerini sağlamıştır.



Şekil 1: SEAGENT Mimarisi

SEAGENT platformunda, etmen tabanlı bir uygulama geliştirmek için şu adımlar izlenmektedir. Önce bir SEAGENT projesi oluşturulur. Oluşturulan projede bir etmen organizasyonunu modellemek için sağlanan farklı bakış

açılarından modellenebilmektedir. Sırasıyla organizasyon, hedef, plan modelleri hazırlanır. Hazırlanan bu modeller SGO araçları tarafından sağlanan destek ile ontolojilere çevrilir. Ayrıca planların detaylı işletim bilgisini sağlayan Java sınıfları da oluşturularak planlarla ilişkilendirilmelidir. Tüm bu işlemlerin ardından, bütün bu ürünlerin daha kolay dağıtılabilmesi ve yeniden kullanılabilirliğin sağlanması için belli bir düzen içinde arşivleme yapılır. Bu arşivleme şekliyle SEAGENT çoklu etmen platformu geliştiricilere daha modüler ve hızlı bir yazılım geliştirme süreci sunar.

SGO, çoklu etmen sistemleri geliştirmek isteyen geliştiriciler tarafından kullanılan bir ortamdır ve genel olarak üç geliştirim aracından oluşur. Bu araçlar: *HGA Düzenleyici*, *Rol Gezgini* ve *Hedef Düzenleyici*'dir. HGA (Hiyerarşik Görev Ağı) Düzenleyici; etmenlerin davranışlarını ve eylemlerini, yani kısacası planlarını görsel olarak gösterebilen ve aralarında ilişki kurulmasına olanak sağlar. Rol gezgini, etmenlere ait olan davranışların yeniden kullanılabilirliğini sağlar. Hedef düzenleyici ise, organizasyonların ve rollerin hedeflerinin geliştirilmesine olanak sağlar.

2.1.1. HGA Düzenleyici

SEAGENT çoklu etmen çerçevesinde, etmenlerin davranışları HGA planlama yaklaşımı [3] ile temsil edilir. HGA düzenleyici, etmenlerin gerçekleştireceği davranışları modellemek için, geliştiricilere görevler ve parametre bağlantıları gibi basit HGA yapılarını sunar. Böylece geliştiriciler HGA yapılarını kullanarak etmenler üzerinde çalıştırmak istedikleri davranışları biçimleyebilirler. HGA düzenleyici bu biçimleme altyapısını gerçekleştirmek için kendi içeriğinde GEF çerçevesi ile tümleşik şekilde çalışmaktadır. GEF gibi bir modelleme çerçevesinin kullanılması sayesinde HGA düzenleyici çabuk ve yapısal bir şekilde geliştirilmiştir. Bununla birlikte geliştirme sonucunda oluşan HGA düzenleyicisine ait olan alan nesneleri ile arayüz nesneleri arasında belli bir soyutlama sağlanmıştır.

HGA düzenleyici, istenilen davranış görsel olarak tasvir etmenin dışında davranış geliştirme sürecinde davranışa ait olan eylemlere işlevsellik özelliğini katan Java sınıflarının JDIT (Java Development Tool) ile otomatik üretimini de sağlar. Taslak halinde üretilen bu Java sınıfları oluşturulduktan sonra gerçekleştirilmek istenilen görevler sınıfların içine dizgelenir.

Bütün eylemlerin Java sınıfları sırasıyla oluşturulduktan sonra, geliştirilen davranışın eylemlerine ait olan Java sınıflarına referans içeren bir HGA ontolojisi oluşturulur. HGA Düzenleyicisinin çıktıları olan HGA ontolojisi ve Java sınıfları, SGO için girdi olarak kullanılır.

2.1.2. Rol Gezgini

Rol gezgini, SEAGENT çoklu etmen geliştirme çerçevesinde yeniden kullanılabilirliği sağlayan bir SGO bileşenidir. Rol gezgini daha önceden geliştirilmiş planların HGA ontolojilerini kullanarak geliştiricinin o anda geliştirdiği davranışa, var olan davranışları kolayca eklemesine ya da yeniden düzenlemesine yardımcı olur. Davranış deposu olarak tabir edebileceğimiz Rol Gezgini aracı değişik kaynakları kullanarak davranışların yeniden kullanılabilmesini sağlar. Bu kaynaklar; bir klasör, bir rol arşivi veya daha önceden oluşturulmuş bir SEAGENT projesi olabilir.

2.1.3. Hedef Düzenleyici

Hedef Düzenleyici, SEAGENT çoklu etmen geliştirme çerçevesinde organizasyonların, etmenlerin gerçekleştirmek istediği hedeflerin ve rollerin geliştirilmesi için kullanılan diğer bir SGO bileşenidir. Bu aracı geliştirirken GMF çerçevesi kullanılmıştır.

GMF gibi bir modelleme çerçevesini Hedef düzenleyicide kullanılmasının sebebi; arayüz vasıtasıyla kullanıcı ile etkileşim sağlayan düzenleyicilerin, model için kullandıkları ana nesneleri hem süreç içindeki işlemlerden hem de arayüz nesnelerinden soyutlama isteğidir. Bu soyutlama sayesinde; düzenleyicide yapılabilecek herhangi bir artırımda ana süreçler ve arayüz üzerinde fazla bir değişikliğe gidilmeyecek, sadece model üzerinde değişiklikler yapılacağı için yeni gereksinimler düzenleyicilere daha çabuk yansıtılabilecektir.

Arayüz yardımıyla hedef, rol ve bu rolleri hangi etmenlerin kullanacağını gösteren basit kavramların eklenmesiyle bir etmen organizasyonu tamamlanmış olur. Etmen organizasyonundaki kavramları biraz açmak gerekirse;

1. *Etmen*: Organizasyondaki hedefleri gerçekleştirmek için geliştirilmiş akıllı, sosyal ve iş birlikçi yazılım parçacıklarıdır.
2. *Rol*: SEAGENT etmen platformunda, etmenlerin gerçekleştirmek istedikleri hedefler için oynadıkları senaryolara göre değişen mantıksal iletişim taraflarıdır.
3. *Hedef*: Bir etmen organizasyonundaki etmenlerin gerçekleştirmek istediği davranışların soyut tanımlanmış şeklidir.

Bu geliştirilen araç yardımıyla tanımlanmış olan etmen organizasyonu bir etmen organizasyon ontolojisine çevrilebilmektedir. SGO sayesinde, geliştiriciler hızlı ve kolayca SEAGENT çerçevesi üzerinde etmen tabanlı bir yazılım geliştirebilirler.

2.2. SEAGENT Çalıştırma Ortamı

SEAGENT çalışma zamanı ortamı (SÇO), SEAGENT çoklu etmen çerçevesinin altyapı elemanlarını birleştiren bir bileşendir. SÇO, iki bakış açısına göre ele alınır; etmen ve çoklu etmen bakış açılarıdır.

2.2.1. Etmen Bakış Açısı

Etmen bakış açısı, tek bir etmenin içyapısını kapsamaktadır. Etmenin alt yapısı, etmenin yaşamsal aktivitelerini yöneten bir işletim sistemi gibi düşünülebilir. Aynı işletim sistemlerinde olduğu gibi altyapı; hangi görevlerin yerine getirileceğinden, bu görevlerin çalışma sırasından ve çalışma süresince oluşabilecek hataların ele alınmasından sorumludur. SEAGENT etmen altyapısı bu bağlamda üç temel bileşenden oluşur: *Planlayıcı*, *İletişim* ve *Bilgi Tabanı*.

2.2.1.1 Planlayıcı

Planlayıcı, etmenin hedeflerini eylemlere dönüştürebilen bir birimdir. Etmen, bir hedefi yerine getirmeye karar verdiği zaman kendi planlayıcısına ilgili hedefi bildirir. Planlayıcının Eşleştirici bileşeni, hedefi gerçekleştirebilecek planı, plan kütüphanesini tarayarak belirler. SEAGENT Çoklu etmen geliştirim çerçevesinde planlar ontolojiler ile temsil edilir. Dolayısıyla bir planın hangi alt görevlerden oluştuğu, bu alt görevler arasında nasıl bir veri akışı bulunduğu gibi bilgiler de ontolojilerde tutulmaktadır. Bu yüzden planlar, doğrudan çalıştırılabilir yapılar değildir. Planlayıcı, etmenlerin işletmek istediği planları ya da anlamsal servisleri iş akışına dönüştürerek çalıştırır. Planlayıcının HGA Dönüştürücü bileşeni, plan ontolojisini kullanarak plana ait bir çizge (iş-akışı) oluşturur. Oluşturulan bu çizge, Çizge İşletici bileşeni tarafından çalıştırılır. Çizge İşletici bileşeni, JVM (Java Virtual Machine)'yi altyapı olarak kullandığı için etmen geliştiricisinin müdahalesi olmadan birçok çizgeyi eş zamanlı olarak çalıştırabilmektedir. Bu sayede; etmenler birkaç hedefi birden gerçekleştirebilir.

2.2.1.2 İletişim

Etmenlerin sosyal bir davranış ortaya koyabilmesi, eş güdümlü ve işbirlikçi olabilmesi için birbirleriyle iletişimde bulunabilmesi gerekmektedir. SEAGENT çoklu etmen çerçevesinin diğer bir bileşeni olan İletişim Katmanı, etmenler arasındaki birebir iletişimi sağlamak için gerekli altyapıyı sağlar. Bu birim fiziksel mesajlaşma için TCP, UDP ve RMI protokollerini desteklemektedir. İletişim birimi, karmaşık mesaj protokollerini veya özel mesaj tiplerini desteklemez. Temel olarak sadece çeşitli iletişim protokolleri üzerinden mesaj alır ve gönderir. Dolayısıyla protokol yönetimi gibi karmaşık sorumlulukları daha üst katmanlara bırakır.

2.2.1.3 Bilgi Tabanı

Bir etmenin yaşamını sürdürebilmesi için yaşadığı alanı, sahip olduğu hedef ve planları, diğer etmenlerle iletişime geçebilmesi için gereken iletişim kurallarını ve yaşam döngüsü boyunca edindiği deneyimleri bilmesi ve yönetmesi gerekmektedir. Bilgi tabanı bileşeni, etmenin bilgi kümesini saklamasını ve yönetebilmesini sağlar. Ayrıca bilgi tabanı bileşenindeki bilgilerin çeşitli temsil dilleri ile tanımlanarak saklanması, etmenin olası bir sorundan sonra (etmenin ölmesi gibi) kaldığı yerden yaşamına devam edebilmesine olanak tanır.

2.2.2. Çoklu Etmen Bakış Açısı

Çoklu etmen bakış açısı, FIPA uyumlu bir etmen platformunda bulunması gereken temel elemanları kapsar. Bu bakış açısında incelenen bileşenler, Çekirdek bileşeninde gerçekleşmişlerdir. SEAGENT etmen platformundaki bu temel elemanlar diğer etmen sistemlerinden farklı olarak geliştirilmişlerdir. Diğer etmen platformlarında AMS (Agent Management System) ve DF (Directory Facilitator) bir etmen olarak tanımlanmıştır. SEAGENT etmen platformunda ise bu temel elemanlar birer rol olarak tanımlanmıştır. SEAGENT etmen platformu ilklendirildiğinde temel roller de ilklendirilir. Bu roller şunlardır:

- **AMS:** AMS rolü; her etmenin tanımlayıcı (AID), adres ve durum bilgilerini kaydederek platforma beyaz sayfalar

servisini sunar. İlklendirilen her etmen, geçerli bir tanımlayıcı alabilmek ve diğer etmenlerle iletişime geçebilmek için AMS' ye kayıtlanmak zorundadır.

- **DF:** DF rolü, etmenlerin servislerini yayınlamasına ve ihtiyaç duyduğu servislerin sağlayıcılarını bulmasına olanak tanıyarak platforma sarı sayfalar hizmetini sunar.
- **ACC:** Aynı hedefi güden veya eş güdümlü çalışması gereken etmenler bir ACC (Agent Communication Channel) etrafında bir araya gelerek bir organizasyonu oluştururlar. ACC' de, İletişim bileşeni tarafından desteklenen her bir iletişim protokolü için bir tane MTS (Message Transport Service) bulunmaktadır. Normal bir etmen, etmenler arası iletişim kurmasını sağlayacak yeteneklere sahip değildir. Bu yetenekleri bağlı olduğu ACC' den edinir, yani mesajlar fiziksel olarak sadece ACC' ler arasında yol alırlar.

Örnekleme gerekirse, bir etmen mesaj göndermek istediği zaman bunu bağlı bulunduğu ACC' ye iletir. ACC, mesaj alıcının da desteklediği bir protokol seçerek ilgili MTS üzerinden mesajı iletir. Mesaj alıcı etmenin bağlı olduğu ACC, ilgili MTS' den bir mesaj alır ve bu mesajı alıcı etmene iletir. Bu yaklaşım, aynı organizasyon içindeki etmenlerin birbiriyle daha hızlı iletişim kurabilmesine ve gerçekleşen tüm iletişimlerin kontrol altında olmasına olanak tanımaktadır.

3. Teşekkür

2002'den bu yana SEAGENT çoklu etmen geliştirme çerçevesinin geliştirilmesinde katkıda bulunan Ege Üniversitesi Bilgisayar Mühendisliği Çoklu Etmen Sistemleri Araştırma grubunun eski ve şimdiki üyeleri; Rıza Cenk Erdur, Erdem Eser Ekinci, Önder Gürcan, Ali Murat Tiryaki, İnanç Seylan, Özgür Gümüş, İbrahim Çakırlar, Övünç Çetin, Hüseyin Kır ve A. Burak Eliaçık'a teşekkür ederiz.

Ayrıca bu çalışma, Türkiye Bilimsel ve Teknolojik Araştırma Kurumu (TÜBİTAK), Elektrik, Elektronik ve Enformatik Araştırma Grubu (EEEAG) tarafından 1008E142 projesiyle desteklenmektedir.

4. Kaynakça

- [1] O. Dikenelli, "SEAGENT MAS Platform Development Environment", AAMAS (Demos): 1671-1672, 2008.
- [2] E. E. Ekinci, A. M. Tiryaki, O. Gurcan, and O. Dikenelli. "A Planner Infrastructure for Semantic Web Enabled Agents.", OTM 2007 Workshops : 95-104, 2007.
- [3] K. Sycara, M. Williamson, and K. Decker, "Unified information and control flow in hierarchical task networks.", AAAI-96 Workshop, 1996.