

TUSAŞ Sistem Entegrasyon Laboratuvarı Doğrulama Yazılımları Altyapısı (SDYA)

Necdet Engin ÖZTUNA*¹, Uğur Melih SÜRME², Orhan UĞURLU³

¹ TUSAŞ, Fethiye Mahallesi Havacılık Bulvarı No:17 Kazan/Ankara

² TUSAŞ, Fethiye Mahallesi Havacılık Bulvarı No:17 Kazan/Ankara

³ TUSAŞ, Fethiye Mahallesi Havacılık Bulvarı No:17 Kazan/Ankara

(Alınış / Received: 23.06.2016, Kabul / Accepted: 12.08.2016,
Online Yayınlanma / Published Online: 09.01.2017)

Anahtar Kelimeler
Aviyonik Veri Yolları,
CMMI,
Çapraz Platform,
Dağıtık Mimari,
Gerçek Zamanlı Sistem Entegrasyon Laboratuvarı

Özet: TUSAŞ'ta Emniyet kritik/güvenilir uçuş yazılımlarının (On-Board Software vb.) entegrasyon ve test aşamalarında kullanılan simülasyon yazılımları ve modellerinin geliştirilmesine olanak sağlaması amacıyla Özgün Sistem Entegrasyon Laboratuvarı Doğrulama Yazılımları Alt yapısı (SDYA) geliştirilmiştir. SDYA'nın geliştirilmesi sürecinde CMMI Seviye-3 uyumlu TUSAŞ Uzay Sistemleri Grup Başkanlığı (USGB) süreçleri rehber alınmıştır. Bu makalede, SDYA'nın geliştirilmesinde platform bağımsızlık, dağıtık mimari, gerçek zamanlı simülasyon altyapısı, grafik kullanıcı arayüzü, aviyonik veri yolları arayüzleri (MIL-STD-1553, Ethernet, Seri vb) için oluşturulan teknik çözüm, uygulanan süreçler, kullanılan teknolojiler ile SDYA'nın uydu alanındaki projelerde kullanım potansiyelinden bahsedilmektedir.

TAI System Integration Laboratory Software Verification Infrastructure(SDYA)

Keywords
Avionic data buses,
CMMI,
Cross platform,
Distributed architecture,
Real time, System Integration Laboratory

Abstract: In TAI, SIL Verification Software Infrastructure ("SDYA") is developed to be used in the development of simulation software required for system and software at System Integration Laboratory for safety critical/reliable flight software and surrounding avionic equipment for various platforms like manned/unmanned aerial vehicles and satellite systems. During the development of SDYA, CMMI Level-3 compatible TAI Space Systems Group processes are used. In this article, the technical solution, applied processes and selected technologies for fulfilling the requirements about cross platform, distributed architecture, real time simulation framework, graphical user interface, avionic data buses capabilities are explained.

*Sorumlu Yazar: eoztuna@tai.com.tr

1. Giriş

SDYA, laboratuvar ortamında uçuş ve yer istasyonu komuta kontrol yazılımlarının doğrulanması için TUSAŞ USGB Yazılım Ekibi tarafından geliştirilmiş genel amaçlı, gerçek zamanlı ve dağıtık bir simülasyon altyapı yazılımıdır. Gerçek aviyonik ekipmanların yerine geçiş fonksiyonel davranışlarını taklit ederek uçuş yazılımı ve yer istasyonu komuta kontrol yazılımının gerçek aviyonik arayüzleriyle haberleşmektedir. Bütünleşik test senaryolarını SEL’de uygulamak için kullanılmaktadır. TUSAŞ tesislerindeki Uydu Sistemleri Entegrasyon Laboratuvarı Şekil-1’de gösterilmektedir.

TUSAŞ; platform entegrasyon sorumluluğu üstlendiği birçok projede geliştirdiği uçuş yazılımlarını laboratuvar ortamında test etmek için doğrulama yazılımlarından yararlanmaktadır. SEL ortamındaki doğrulama, aviyonik sistem ve mimari ile ilgili hataların tespit edilmesine yardımcı olmakta, böylece proje risklerini ve maliyetini azaltırken, aviyonik sistemin merkezinde yer alan uçuş yazılımlarının olgunluğunu da arttırmaktadır. Platform aviyonik sistem geliştirme sürecinde SEL, simülasyon ortamından nihai sisteme geçişi sağlayan bir test ortamı sağlamaktadır. Karmaşık sistemlerin gereksinimlerini karşılayabilmek için kullanılan kritik bir doğrulama ve geçiş ortamı olarak değerlendirilmektedir.

SDYA, CMMI Seviye 3 uyumlu TUSAŞ USGB süreçlerine uygun olarak geliştirilmiştir. Ayrıca, ESA ECSS[1-3] ve RTCA DO-178B[4]’de belirtilen doğrulama aracı gereksinimlerini de karşılamaktadır. Yazılımdaki değişiklikler sonucu yapılması gereken

regresyon testlerini manuel ve otomatik geliştirmeyi kolaylaştıran bir test altyapısına da sahiptir.

SDYA, kendi kalifikasyon testlerinin yanı sıra, ANKA İnsansız Hava Aracı (İHA) Projesinde ANKA modellerinin geliştirilmesi ve uyarlamasında da kullanılarak, bu platforma özgü modeller ile birlikte de ayrıca kalifiye edilmiştir. Kalifikasyon sürecinde izlenen teknik yaklaşım hakkında detaylı bilgi ve uyum sağlanan standartlar “TUSAŞ Özgün Sistem Entegrasyon Laboratuvarı Doğrulama Yazılımları Alt Yapısı (SDYA) Doğrulama ve Geçerleme Deneyimleri ve Uydu Projelerindeki Kullanımı” makalesinde [5] verilmiştir.

SDYA’nın Fonksiyonel Doğrulama Zincirindeki Yazılım Doğrulama faaliyetlerinde ve/veya Dinamik Uydu Simülatörü Geliştirilmesinde ihtiyaç duyulan uçuş yazılımının ve aviyonik sistemin doğrulanmasında kullanılması da hedeflenmektedir. Ayrıca, uydu kontrol prosedürlerinin ve uydu operatörü eğitimlerinde destek aracı olarak kullanılması da değerlendirilmektedir. Uydu alanındaki kullanımında, tasarım kriteri olarak SMP-2 standardı uyumluluğu benimsenmiştir. SMP-2, komponent tabanlı tasarım ve Model Driven Architecture (MDA) yapıları ile UML ve XML standartlarını temel alarak, simülasyon modelinin platforma özgü ve platform bağımsız özelliklerinin ayrı ayrı değerlendirilmesi prensibini desteklemektedir.

SDYA, uçuş yazılımlarının doğrulanması için simülasyon tabanlı bir altyapı sağlamakta ve kullanıcılar aşağıdaki kritik aktiviteleri gerçekleştirmektedir;



Şekil 1. TUSAŞ Uydu Sistemleri Entegrasyon Laboratuvarı

1. Sistem ekibi: Operasyonel gereksinimlerin doğrulanması
 2. Aviyonik sistem ekibi: Aviyonik ekipman testleri, aviyonik arayüzlerin doğrulanması, kalifikasyon testleri, aviyonik sistem entegrasyonu
 3. Yazılım ekibi: Yazılım geliştirme (simülasyon model geliştirme), yazılım entegrasyonu
- Sistem/Yazılım Doğrulama ekibi: Yazılım ve donanım arayüzlerinin doğrulanması, performans testleri, aviyonik yazılım/donanım kabul testleri

2. Bileşen Tabanlı SDYA Mimarisi ve SDYA Geliştirilmesinde Kullanılan Teknolojilerin Seçimleri

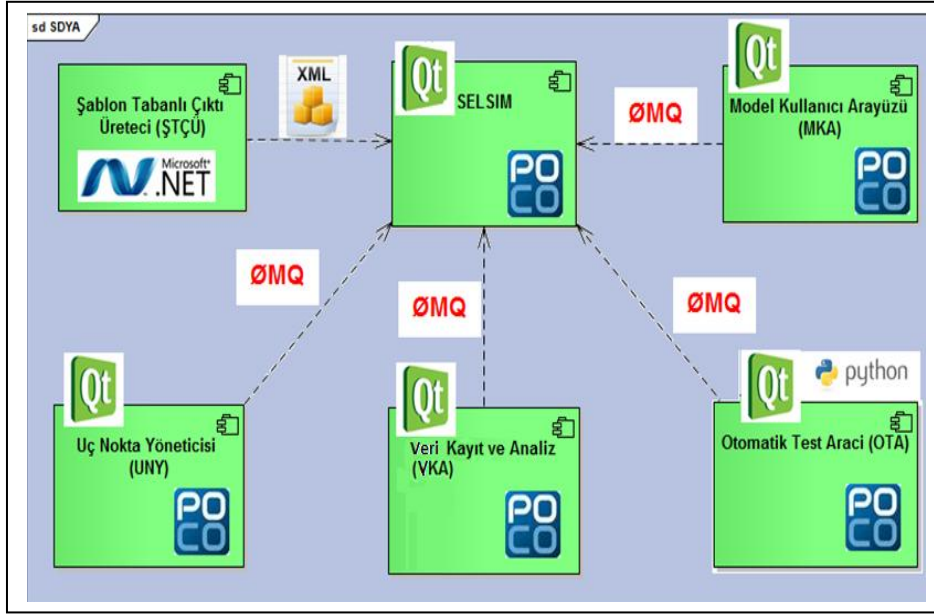
SDYA geliştirme sürecindeki kritik teknolojilerin seçimi CMMI Karar Analiz ve Çözüm (KAÇ) süreciyle gerçekleştirilmiştir. SDYA'nın bileşen tabanlı mimarisi ve SDYA geliştirilmesinde kullanılan teknolojiler Şekil 2'de gösterilmiştir.

SDYA geliştirme sürecinde dağıtık mimari isteklerinin karşılanması ve yazılım modülleri arasındaki veri iletişimini sağlayabilmek için arakatman yazılımı kullanılmasına karar verilmiştir. Uygulanan CMMI KAÇ süreci sonunda bu amaca yönelik olarak açık kaynak kodlu ZeroMQ (ZMQ) [6] yazılımı seçilmiştir. ZeroMQ ile uygulamaların ağ üzerinde az gecikmeyle çalışması ve verilerin yüksek oranda iletilmesi sağlanmaktadır.

ZeroMQ, yeni bir mesajlaşma sisteminin hızlı bir şekilde oluşturulabilmesi için bir bağlantı noktası arayüzü de sunmaktadır. ZeroMQ bağlantı noktaları, işletim sistemine yakın ve alt seviye görünmesine rağmen birçok farklı özellik sağlamaktadır. Örneğin: tek bir ZeroMQ bağlantı noktası, birden çok bağlantı noktasına bağlanarak bunlar arasındaki mesajlaşmayı dengeleyebilmektedir.

Kullanıcı arayüzü geliştirme aracı olarak, dil desteği, çapraz platform yazılımları desteği, performans kriterleri, öğrenim ve kullanım kolaylığı gibi kriterler göz önünde bulundurularak gerçekleştirilen CMMI KAÇ süreci sonucunda Qt[7] seçilmiştir. Qt yazılımının "signal-slot" yapısı, kullanıcı arayüzü bileşenlerinin model ile olan iletişimini kod karmaşıklığı olmadan, kolay anlaşılabilir bir şekilde ve kısa sürede gerçekleştirilmeye olanak sağlamaktadır.

SDYA'nın "RT_preempt Linux" gibi gerçek zamanlı bir işletim sistemi üzerinde çalışabilmesi için Qt Framework ve "POCO" yazılım kütüphanesi gibi çapraz platform yazılım kütüphaneleri kullanılarak ayrık bir bileşen mimarisi ve modüler bir tasarım oluşturulmuştur. Gerçek zamanlı Linux işletim sisteminin yanı sıra yüksek performanslı ZeroMQ arakatman kütüphanesinin entegrasyonu ile yüksek performanslı ve modüler bir ürün ortaya çıkarılmıştır.



Şekil 2. SDYA kullanılan teknolojiler ve bileşen tabanlı mimari

3. SDYA Mimarisi

SDYA Mimarisi oluşturulurken SMP-2 standardı göz önünde bulundurmuş, yaklaşım olarak Katmanlı Mimari yapısı seçilmiştir. SDYA Katmanlı Mimarisi Şekil 3'te gösterilmiştir.

Tekrar kullanılabilirlik, test edilebilirlik, modülerlik, esneklik, ölçeklenebilirlik ve genişletilebilirlik gibi yazılım kalite faktörlerini destekleyebilmesi, seçilen kritik teknolojilerdeki değişimlerin bunları kullanan diğer yazılım bileşenlerini asgari seviyede etkilemesini sağlayabilecek bir seviyede soyutlanması, hata toleransı ve performansı artırma hedefleri nedeniyle, SDYA mimarisinde katmanlı bir mimari yaklaşımı uygulanmıştır.[8] Bu mimariyi üç ana başlıkta inceleyebiliriz.

3.1. Çekirdek Katman

Çekirdek katmanı, mimarinin ana bileşenidir. Ortak Servisler, İşletim Sistemi Soyutlama Katmanı, Mesajlaşma Soyutlama Katmanı ve Donanım Soyutlama Katmanını (DSK) içermektedir.

DSK gerçek donanımları temsil eden sanal bir katmandır. Seri, MIL-STD-1553B, Ethernet gibi aviyonik arayüzlerin sisteme kolaylıkla eklenmesini sağlamaktadır. DSK sayesinde simülasyon modelleri gerçek veri arayüzlerini kullanarak uçuş yazılımları ve/veya gerçek ekipmanlarla mesaj alışverişi yapmaktadır.

3.2. Simülasyon Motoru (SELSIM) Katmanı

SELSIM katmanında Model Yöneticisi, Mesaj Yöneticisi, Veri Yöneticisi, Fonksiyonel Servisler ve I/O Yöneticisi bulunmaktadır.

Mesaj Yöneticisi simülasyon modelindeki mesajların zamanlamasını yönetmektedir. Mesaj yöneticisi sayesinde gönderim zamanı gelmiş olan mesajlar, I/O'dan ilgili aviyonik ekipmanlara/uçuş bilgisayarına gönderilmek üzere Donanım Soyutlama Katmanına iletilmektedir.

Veri Yöneticisi ise ham ve mühendislik veri dönüşümlerini destekler. Veri



Şekil 3. SDYA Katmanlı Mimarisi

yöneticisi, uçuş bilgisayarından veya aviyonik cihazlardan alınan ham verilerin, sistemde tanımlı olan veri tabanında bulunan değerlere göre mühendislik dönüşümünü gerçekleştirmektedir. Simülasyon modelleri tarafından üretilen ve uçuş yazılımı veya aviyonik cihazlara gönderilecek olan mühendislik verilerinin, veri tabanında bulunan tanımlamalara göre ham değerlerine dönüşümünü gerçekleştirmektedir.

3.3. Uygulama Katmanı

Uygulama katmanında Kullanıcı Arayüzü Soyutlama katmanı ve Model Kullanıcı Arayüzü, Veri Kayıt ve Analiz, Otomatik Test Aracı, Uç Nokta Yöneticisi, Şablon Tabanlı Çıktı Üretici uygulamaları bulunmaktadır. Projeye özgü olarak geliştirilen simülasyon modelleri de bu katmanın içinde yer almakta ve SELSIM Katmanı ile arayüzleri bulunmaktadır.

Kullanıcı Arayüzü Soyutlama Katmanı, MVC (Model-View-Controller) mimari desenine göre tasarlanmış bir katmandır.

SDYA gerçek ekipmanla arayüzü olan, dinamik kontrol özelliğine sahip ve veri enjekte edilebilen, gerçek zamanlı bir simülasyon yazılımıdır. Simülasyon modelleri ise her projenin kendine özgü gereksinimleri doğrultusunda, geliştiricilere sağlanan arayüzler kullanılarak oluşturulan ve SELSIM tarafından koşturulan uygulama yazılımlarıdır. Performans ve yeni model ekleme ile ilgili gereksinimleri karşılayabilmek için modellerin bilgilerini farklı uçlara dağıtan bir arakatman yazılımı kullanılmıştır. Böylece farklı işletim sistemlerindeki uygulamalar arası mesajlaşmalarda basit, güvenilir, hızlı ve verimli bir yol izlenmiştir.

Dağıtık sistemler kullanıcıya tek bir sistem gibi görünmelerine rağmen birden fazla bağımsız platformdan oluşabilmektedirler. Dağıtık uçlar arasındaki mesajlaşma farklı platformlarda farklı etkileşimler gerektirmektedir. Yayınla/Abone ol mekanizması platform isterlerinin

çoğunu karşılarken, uydu platformunun “gönderme”, “gönderme-onaylama”,

“istek-cevap”, “arama-onaylama-cevap” ve “işlem-onaylama-güncelleme-cevap” gibi bazı farklı etkileşim ve ileri seviye ihtiyaçlarını karşılamak için yeterli esneklik sağlayamamaktadır. Bu nedenle, SDYA'nın arakatman yazılım çözümü, farklı mesajlaşma kalıplarını desteklerken yeterli esnekliği sağlayan ZeroMQ olmuştur.

Dağıtık mimari ve arakatman yazılımı sayesinde simülasyon modelleri birden fazla bilgisayarda çalıştırılabilmektedir. Bu sayede yoğun işlem gerektiren simülasyon modellerinin, performansı etkilemeden çalıştırılması sağlanmaktadır. Uç Nokta yöneticisi kullanılarak sistemdeki bilgisayarlar üzerindeki sağlık bilgileri ve uygulama çalıştırma/kapatma işlemleri gerçekleştirilebilmektedir. Şekil 4'te SDYA bileşenlerinin birden fazla bilgisayar ve işletim sistemi kullanılarak çalıştırılmasına yönelik örnek bir kurulumu gösterilmektedir.

4. SDYA Modülleri

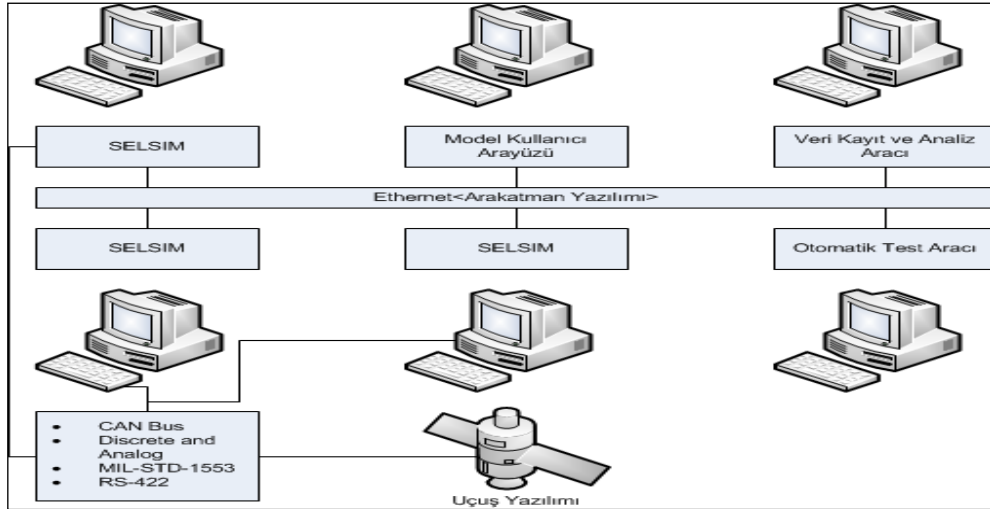
4.1. Simülasyon Motoru (SELSIM)

Aviyonik cihaz ve çevre/ortamlara ait fonksiyonel modellerin yönetimi, veri yönetimi (ham mühendislik veri dönüşümleri), I/O arayüzlerinin yönetimi, zaman ve çerçeve yönetimi,

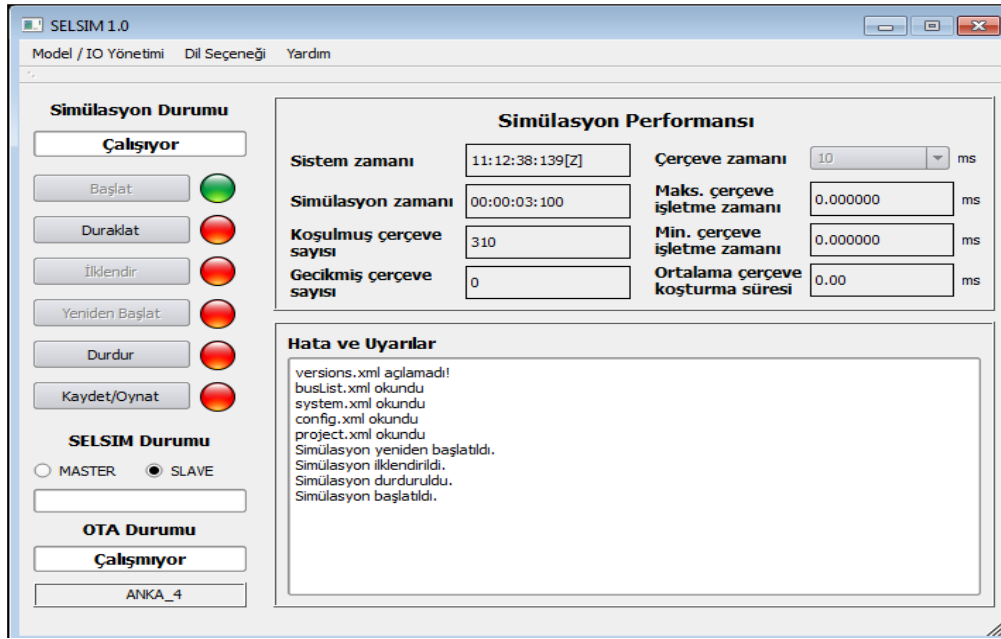
hata durumlarının raporlanması, durum/mod bilgilerinin yönetilmesi ve dağıtık/paylaşımlı ağ üzerinden veri alıp/verme işlevlerini arakatman üzerinden gerçekleştiren yazılımdır.

SELSIM ekranından simülasyonun çalışma durumu ve modu (master/slave) yönetilir ve izlenir. Ayrıca, simülasyon performans verilerini görüntülemeye ve çerçeve zamanını belirlemeye yarayan bir alanı ve hata/uyarı mesajlarının kullanıcıya gösterildiği ekran mevcuttur.

SELSIM ekranından ulaşılan Model Yönetimi ekranında, modellerin frekansları ve çalışma durumları anlık olarak değiştirilebilmekte ve mevcut bilgisayarda çalışmakta olan simülasyon modellerinin çalışma ve gecikme durumları sergilenmektedir. Model Yönetimi bileşeni sayesinde, sistemdeki bütün bilgisayarların ve üzerinde koşan modellerin, paketlerin ve mesajların listelenmesi gerçekleştirilebilmektedir. Simülasyon sisteminin model tasarımının (model ekleme, değiştirme, silme) oluşturulmasını ve IO Yönetimi ekranı ile birlikte mesajların I/O ayarlarının yapılmasını da SELSIM sağlamaktadır. SELSIM Arayüzü Şekil-5'te verilmiştir.



Şekil 4. SDYA örnek kurulum gösterimi



Şekil 5. SELSIM arayüzü

4.1. Model Kullanıcı Arayüzü (MKA)

Kullanıcının, simülasyon veri girişlerini yapmak ve verileri monitör etmek için kullandığı Metin Tabanlı Panel ve Görsel Panel arayüzlerinden oluşan yazılım birimidir.

Metin Tabanlı Panel'de kullanıcı projeye ait bütün mesajlara ve mevcut alanlarına ulaşım, monitör etmek ya da veri girişi yapmak istediklerini seçebilmekte ve

seçilen alanlar tablo yapısında listelenmektedir. Metin Tabanlı Panel Arayüzü Şekil 6'da verilmiştir.

Görsel Panel'de ise kullanıcı, simülasyon çalışır halde değilken tanımlı görsel elemanlar kullanarak paneller oluşturabilmekte ve simülasyon çalışırken bu alanlarla monitör etme ve veri girişi fonksiyonlarını

gerçekleştirebilmektedir. Görsel Panel Kullanıcı Arayüzü Şekil 7’de verilmiştir.

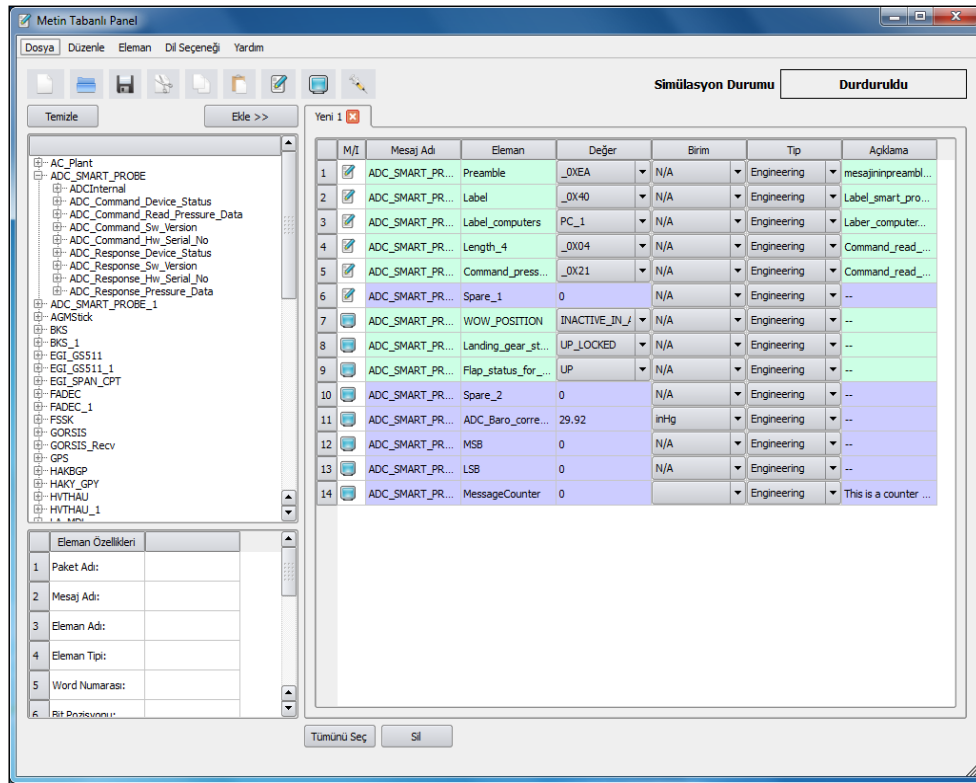
4.1. Veri Kayıt ve Analiz Aracı (VKA)

Simülasyon verilerinin kaydının ve bu kayıtların analizinin yapılmasını sağlayan bir yazılım birimidir. Kayıt özelliği sayesinde Mesajlaşma Soyutlama Katmanı üzerinden alınan model bilgisi ve ilgili mesajların bellekteki bir dosya sistemine kaydedilmesini sağlamaktadır. Analiz yeteneğiyle de istenilen zamanda ilgili mesajların değerlerinin kayıt edilen dosya kullanılarak grafiksel ve tablo olarak yeniden oynatılmasına ve

incelenmesine imkan tanır.

4.4. Otomatik Test Aracı (OTA)

Elemana değer atama ve okuma, simülasyon durumunun değiştirilmesi gibi işlemleri Python betik dili ve C++ ve C# gibi uygulama programlama arayüzü (API) üzerinden sağlayan bir yazılım aracıdır. Bu API sayesinde SDYA



Şekil 6. Metin Tabanlı Panel arayüzü

haricindeki uygulamalarla, simülasyon modelinde kullanılan elemanlara değer atama ve elemanları monitör etme,

simülasyon durumunun değiştirilmesi gibi komutlar gönderilmesine olanak sağlanmaktadır.

4.5. Şablon Tabanlı Çıktı Üretici (ŞTÇÜ)

Simülasyon altyapısında kullanılacak veri yapısını içeren doküman ve kodları üretmeye yarayan yazılım bileşenidir. Veritabanına bağlanarak seçilen proje ve anahattan (baseline) veritabanı bilgilerini çekmeyi sağlamaktadır. Belirlenen format ve ayarda simülasyon altyapısında kullanılacak ortak doküman ve kodları üretirir, bu çıktılar SELSIM, MKA ve OTA tarafından kullanılmaktadır.

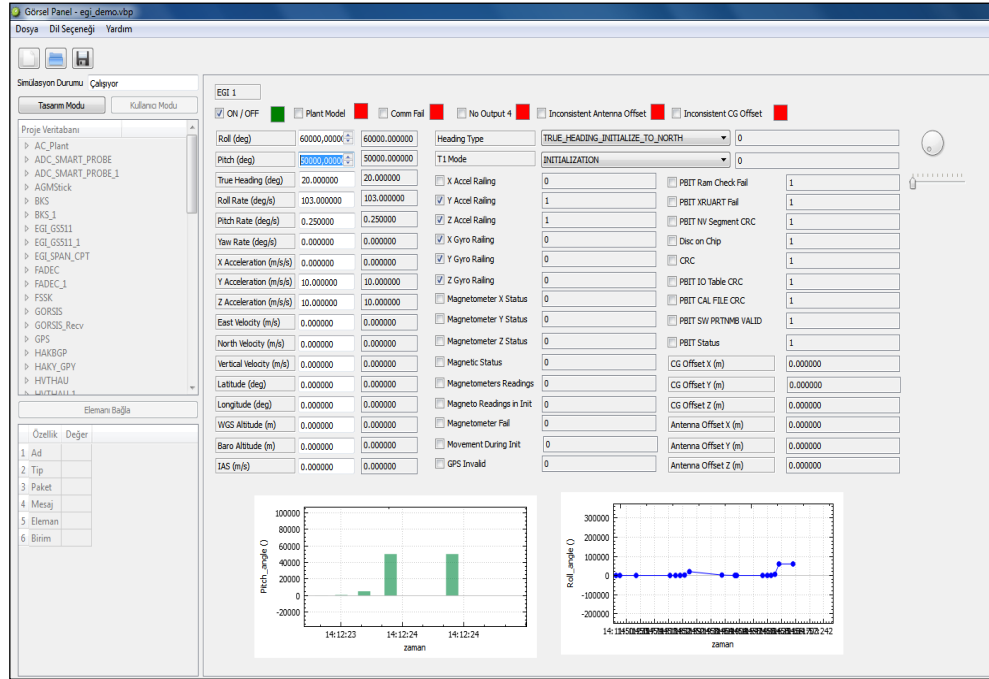
4.6. Uç Nokta Yöneticisi (UNY)

Uç Nokta Yöneticisi, sistemde çalışmakta olan bütün bilgisayarlar ve bu bilgisayarlar üzerinde koşan SELSIM, MKA gibi simülasyon araçlarını ve durumlarını gösteren, kullanıcıya bunları

yönetme imkanı sağlayan yazılım bileşenidir. Sistemdeki bilgisayarların her birinde bulunan bu yazılım, simülasyon sisteminde çalışan tüm uygulamaları, üzerinde buldukları bilgisayarlar, çalışma durumları ve işlemci/bellek kullanımı bilgileriyle birlikte listelemektedir.

5. SDYA'nın Uydu Alanında Kullanımı

SDYA, sağladığı yeniliklere ve değişime açık simülasyon altyapısı, çeşitli sensör modellerinin ve aviyonik veri yollarını destekleyecek şekilde oluşturulmuş simülasyon yazılımı ve gerçek zamana yakın performansı ile uydu alanındaki simülasyon ihtiyaçlarını başarılı bir şekilde karşılamaktadır.



Şekil 7. Görsel Panel arayüzü

Ayrıca mimari seviyede sağladığı ortak servisler ile SMP-2'nin tanımladığı profil tanımlarından "Run-time Infrastructure (Profile 2)" tanımına uygun olarak geliştirilmiştir.

TUSAŞ bünyesinde Uçay Sistemleri Grup Başkanlığı (USGB) Yazılım Ekibi tarafından uydularda kullanılmak üzere geliştirmekte olan Uçuş Yazılımı, Gözlem ve Kontrol Yazılımı, Görev Planlama Yazılımı ve Uçuş Dinamikleri yazılımlarının doğrulanması

aşamalarında SDYA'yı kullanmaktadır. Ayrıca SDYA'nın hedef uydu projelerinde aşağıdaki görevleri üstlenmesi planlanmaktadır:

- Performans ve sağlamlık doğrulaması için doğrulama aracı
- Aviyonik Fonksiyonel Zincir Doğrulaması (FCV) için doğrulama aracı
- Dinamik Uydu Simülatörü (DSS) için geliştirme ve doğrulama aracı
- Uydu montaj, entegrasyon ve test prosedürleri için geliştirme ve doğrulama aracı
- Uydu operasyonel prosedürleri doğrulaması için destek aracı
- Uydu kontrol merkezi doğrulaması için destek aracı
- Uydu operatör eğitimi için destek aracı
- Tespit edilen uçuş anormalliklerinin analizi ve incelenmesi için destek aracı

6. Sonuç

Bu makale, SDYA mimarisini, kullanılan teknolojileri ve altyapının temel öğelerini **Kaynakça**

- [1] ECSS-E-ST-40C Space Engineering: Software, ESA-ESTEC, 6 March 2009, <http://www.ecss.nl/>
- [2] ECSS-Q-ST-80C Space Product Assurance: Software Product Assurance, ESA-ESTEC, 6 March 2009, <http://www.ecss.nl/>
- [3] ECSS-E-TM-40-07 Space Engineering: Simulation Modelling Platform, ESA-ESTEC, 25 January 2011, <http://www.ecss.nl/>
- [4] RTCA/DO-178B, Software Considerations in Airborne Systems and Equipment Certification, 1 December 1992, <http://www.rtca.org/>
- [5] Nargül, A.S., Güçlü, K., Ülgen, H.İ., Kalebek, A., Öztuna, N.E., Sürme, U.M., Çakır, U., "TUSAŞ Sistem Entegrasyon Laboratuvarı

anlatmakta, ayrıca, bu tür altyapıların özgün geliştirilen uçuş yazılımlarının SEL ortamında doğrulanmasındaki önemini ortaya koymaktadır. SDYA'nın hava platformları dışında uydu projelerinin geçerleme sürecinde de kullanılması ve Yazılım Ürün Hattı (Software Product Lines) için temel oluşturması planlanmaktadır. SDYA'nın mevcut kabiliyetleri ve esnek mimarisıyla uydu entegrasyon ve test aktivitelerinin büyük bir kısmını destekleyebileceği öngörülmüştür. SDYA, sistem/yazılım geliştirme ve test ekiplerine, geliştirme ve doğrulama sürecinin risklerini azaltan ve verimliliği arttıran güçlü ve esnek bir yapı sunmaktadır. Uydulara yönelik çok hassas ve güvenilir olması gereken doğrulama süreçleri, otomatik test kullanımının artırılması gibi test teknikleri ve prosedürleriyle daha da iyileştirilebilir. Uydu alanındaki bu zorlu isteklerin karşılanması, karmaşık sistemlerin entegrasyon ve testlerini destekleyebilmek için SDYA gibi alt yapıların kullanımı ve sürekli idamesi daha da kritik bir hale gelmiştir.

Doğrulama Yazılımları Altyapısı (SDYA): Geliştirme, Doğrulama ve Geçerli Kılma Deneyimleri ", Ulusal Savunma Uygulamaları Modelleme ve Simülasyon Konferansı (USMOS), 2015

- [6] ZeroMQ Mesajlaşma katmanı, <http://www.zeromq.org/> Erişim Tarihi: 15.06.2016
- [7] Qt Framework, www.qt.io/ Erişim Tarihi: 15.06.2016
- [8] Sürme, U.M., Öztuna, E., Uğurlu, O., Çakır, U., Nargül, A.S., Güçlü, K., "SDYA: A Real Time and Distributed Software Verification Infrastructure for Validating Flight Software (On-Board Software) at System Integration Laboratory.", ESA Simulation for European Space Programmers (SESP) 2015