2008 - 347.

TÜBİTAK

TÜRKİYE BİLİMSEL VE TEKNOLOJİK ARAŞTIRMA KURUMU

THE SCIENTIFIC AND TECHNOLOGICAL RESEARCH COUNCIL OF TURKEY

Elektrik, Elektronik ve Enformatik Araştırma Grubu

Electrical, Electronical and Informatics Research Group

Anlamsal Birlikte İşlerlik Platformu Tabanlı Akıllı Sağlık Takibi (Intelligent Healthcare Monitoring based on a Semantic Interoperability Platform)

Proje No: 105E133

Prof. Dr. Asuman Doğaç

EYLÜL 2008
ANKARA

Önsöz

Bu proje kablosuz tıbbi sensor verisi ve hastane bilgi sistemleriyle kolay entegrasyonu sağlayarak Akıllı Sağlık İzleme ve Karar Destek Sistemi geliştirmeyi amaçlamaktadır. Projede devamlı hasta gözetimi, ajan teknolojisindeki "ajan davranışı" nın akıllı bir klinik karar destek sistemiyle desteklenmesiyle sağlanmıştır. Bu karar destek sistemi bilgisayarlaştırılmış klinik yönergelere dayalı olup birlikte işlerlik problemini aşmak amacıyla anlamsal olarak zenginleştirilmiş web servisleri kullanılarak ulaşılmaktadır. Ek olarak, klinik yönergenin işleyişini gösteren kullanıcı dostu bir arayüz de geliştirilmiştir.

Bu proje, Tübitak Elektrik, Elektronik ve Enformatik Araştırma Grubu tarafından desteklenmektedir.

1. İçindekiler

1. Şekil Listesi

Proje Başlığı

Anlamsal Birlikte İşlerlik Platformu tabanlı Akıllı Sağlık Takibi
(Intelligent Healthcare Monitoring based on a Semantic Interoperability Platform)

Proje Yürütücüsü	Telefon
Prof. Dr. Asuman DOĞAÇ	**0 312 210 55 98**
Adresi	Faks
İnönü Bulvarı, SRDC, Bilgisayar Müh. Bölümü, ODTÜ	**0 312 210 10 04**
	e-mail
	asuman@srdc.metu.edu.tr

ÖZET

Avrupadaki yaşlı nufus arttıkça her düzeyden sağlık personelinin yükü de daha çok artıyor. Son otuz yıldır Avrupa Birliği ülkelerinde uygulanan sağlık hizmetleri sayesinde bu ülkelerde ölüm oranları büyük miktarda düşmüştür. Ölüm oranlarındaki düşüşte sağlanan bu başarı, beraberinde sağlanan sağlık hizmetlerine olan beklentileri ve talepleri de artırmıştır. Bu etkenlerin yanında bilgi sistemleri, mobil ulaşım araçları, kablosuz sağlık algılayıcıları, ağ ve internet teknolojileri, sağlık personellerine hizmetlerinin uzaktan verilebilmesi için büyük olanaklar sağlar. Uzaktan sağlanabilecek bu sağlık hizmetleri sayesinde, sağlık personelinin yükü hafifletilmiş ve bu hizmetlerin kalitesi daha da artırılmış olacaktır.

Bu proje çerçevesinde akıllı sağlık takip ve karar destek sistemleri geliştirerek, kablosuz medikal algılayıcılardan elde edile bilginin günümüzde kullanılan sağlık bilgi sistemlerine entegrasyonunu sağlamak amaçlanmıştır. Hasta takibi, kullanılacak olan 'akıllı ajan' teknolojisi ve bu ajanların davranışlarını belirlemede kullanılacak klinik uygulama kılavuzlarına dayalı olarak çalışacak akıllı karar destek mekanizmaları sayesinde sağlanacaktır. Yapılacak olan sistemde değişik sağlık bilgi sistemlerinde depolanmış hastalara ait gerekli bilgilere, birlikte işlerliği ve bilgi uyumunu sağlayabilmek için anlamsal olarak zenginleştirelecek web servis teknolojileriyle ulaşılacaktır. Bu sayede medikal algılayıcılardan elde edilen bilgiler ve gözlemler ile birlikte hastanın geçmiş sağlık bilgileri de mantıksal işleme sürecinde kullanılabilecektir.

Anahtar Sözcükler

Akıllı hasta takibi, Karar destek sistemi, kablosuz medikal algılayıcılar, birlikte çalışablirlik platformu, klinik uygulama kılavuzları

Project Title

Intelligent Healthcare Monitoring based on a Semantic Interoperability Platform

Principal Investigator	Telephone
Prof. Dr. Asuman DOĞAÇ	**0 312 210 55 98**
Address	Telefax
İnönü Avenue, SRDC, Computer Eng. Dept., METU	**0 312 210 10 04**
	e-mail asuman@srdc.metu.edu.tr

ABSTRACT

The medical practitioners at all levels are becoming more overloaded as the aging population of Europe increases. The health services of the EU can claim considerable credit for the decline in mortality over the last thirty years. However this success, particularly the fall in mortality rates among older people, has increased the demand for healthcare. Furthermore, there are discrepancies in health status between the old and new member states due to health system failures in the latter. On the other hand Information technology, combined with recent advances in networking, mobile communications and wireless medical sensor technologies offers a great potential to support healthcare professionals and to deliver health care services at a distance hence providing the opportunities to improve healthcare.

This project aims to develop an intelligent healthcare monitoring and decision support system on a platform integrating the wireless medical sensor data with hospital information systems. In this project, the patient monitoring will be achieved by using agent technology where the "agent behavior" will be supported by intelligent decision support systems based on clinical practice guidelines. In our system, patient history stored in medical information systems will be accessed through semantically enriched Web services to tackle the interoperability problem. In this way, not only the observations received from wireless medical sensors but also the patient medical history will be used in the reasoning process.

Keywords

intelligent healthcare monitoring, decision support systems, wireless medical sensors, interoperability platform, clinical practice guidelines

2. Giriş

Sağlık hizmetlerine duyulan talebin artmasıyla beraber tıp uygulamaları giderek karmaşıklaşmıştır. Sağlık örgütü hem tıp uygulamalarında artan bu karmaşıklığa çare olması hemde sağlık hizmetlerini daha da geliştirmek için klinik uygulama kılavuzları geliştirmektedir. Buna örnek olarak, National Guideline Clearinghouse™ belirtilere dayanan klinik uygulama kılavuzları ve ilgili diğer dokümanları içeren bir veri tabanı oluşturmuş ve bu veri tabanını kullanıma açmıştır. Sağlık uzmanları, yaygınlaşmış sağlık standartlarına ve uygulama kılavuzlarına rağmen, bu kılavuzları anlamada ve kullanmada zorluk çekiyorlar. Bu problem beraberinde klinik kılavuzlarında belirtilen method ve bilgileri otomatikleştirmek için bilgisayara bağlı karar destek sistemlerinin gereksinmini artırıyor. Bu sistemlerin uygulanmasında da bazı zorluklar mevcuttur, bunlardan bir tanesi de hastaya ait bilgilerin çok farklı bilgi kaynaklarında ve farklı formatlarda olma olasılığıdır.

Bu projede klinik karar destek mekanizması sistemde ajanların davranışlarını sağlayacak bir mekanizma olarak kullanılacaktır. Bu sistem, hastaların sağlık bilgi sistemlerinde depolanmış medikal bilgilerine anlamsal olarak zenginleştirelecek web servisler aracılığıyla ulaşacak ve böylece birlikte çalışılabirlik problemleri aşılmış olacaktır. Bu yolla bilgilerin mantıksal işlenme sürecinde sadece algılayıcılardan elde edilecek hastanın o andaki fizyolojik belirtileri değil, hastanın geçmiş medikal bilgileri de kullanılacaktır. Bu bilginin gereksinimindeki temel neden ise klinik uygulama kılavuzlarında bir sonraki klinik yolun seçilmesinde fizyolojik belirtiler ve hasta tedavi planıyla birlikte hastaların geçmiş medikal bilgilerininede (örnek olarak teşhis, tedavi listesi, alerji/ilaç etkileşimleri) ihtiyaç duyulmasıdır. Medikal algılayıcılardan gelecek bilgiler ya özel bir formata sahip olacaktır (örneğin, elektro kardiyogram verisi için, Philips' XML ECG veri formatı) yada belli bir standarda uygun şekilde oluşturulacaktır (yine elektro kardiyogram verisi için, mevcut standartlar: SCP-ECP, US Food and Drug Administration FDA/HL7 Annotated ECG, I-Med ve ecgML). Fakat her iki durumda birlikte çalışabilirlik problemine bir çözüm getirmez. Bunun nedeni kullanılabilecek birden çok standardın mevcut olmasıdır. Bu problem algılayıcılardan elde edilecek verinin elektronik hasta kaydı ile entegrasyonunda, sağlık bilgi sistemlerinin de çok farklı standartlara (HL7v2.x, HL7v3 CDA, CEN ENV 13606 EHRExtract, openEHR Archetypes) uymalarından dolayı daha da karmaşıklaşır. Bu problemlere ek olarak farklı medikal algılayıcılardan gelebilecek verilerin birleştirilmesinde doğabilecek birlikte çalışablirlik problemlerine de çözüm getirmek gerekir.

Bütün bu birlikte çalışabilirlik problemleri, hem algılayıcılardan elde edilen verilere ve hem de sağlık bilgi sistemlerinde saklı verilere, işlevsellikleri ve kullandıkları mesaj yapıları ontolojiler yardımı ile belirtilmiş web servisler yardımıyla ulaşılarak çözülecektir. Böylece bilgi alışverişi sırasında oluşan işlevsellik ve anlamsal farklılıklar belirtilen web servisleri ve ontolojileri ile çözümleneceklerdir. Ontolojiler ve bunların arasında tanımlanacak ontoloji eşleşmeleri yardımı ile farklı medikal platformlar arasında anlamsal bütünlük sağlanacaktır. Böyle bir 'birlikte işlerlik' platformunun oluşturulması bize daha sonra sağlık takip sürecinde karar destek sisteminin oluşturulmasına imkan sağlayacaktır. Bütün bunlarla birlikte sistem her hastaya tahsis edilmiş ajanlar yardımıyla hastaları sürekli olarak izleyebilecek ve sağlık uzmanlarınına karar destek mekanizması sunarak yardım edecektir.

Böyle bir bilgi altyapısını kurmak beraberinde hasta bilgilerinin güvenlik ve gizlilik içinde saklanması ve taşınmasını gerektiriyor. Hastann kimliği ve medikal kayıtları gelişi güzel bir şekilde saklanamayacağı gibi çoğu sağlık sitemi sağlayıcısının da bu bilgilere ulaşma hakları farklı olacaktır. Projede tüm bunlara çözüm getirecek kapsamlı güvenlik ve gizlilik mekanizmaları öneriliyor. Bu mekanizmaları sağlarken kişisel bilgilerin nasıl işleneceğini anlatan d 95/46/EC ve 2002/58/EC ve tıbbi verilerin saklanmasında önerilen R(97)5 önerileri göz önüne alınacaktır.

3. Genel Bilgiler

PROJE NO	105E133
PROJE ADI	Anlamsal Birlikte İşlerlik Platformu Tabanlı Akıllı Sağlık Takibi (Intelligent Healthcare Monitoring based on a Semantic Interoperability Platform)
KURULUŞ	Orta Doğu Teknik Üniversitesi
PROJE YÜRÜTÜCÜSÜ	Prof. Dr. Asuman Doğaç
ARAŞTIRMACILAR	Gökçe Banu Laleci Ertürkmen, Yıldıray Kabak, Mehmet Olduz, Özgür Gülderen, Alper Okcan, Tuncay Namlı, Umut Orhan, Mustafa Yüksel, İbrahim Taşyurt, Musa Ataş

PROJE SÜRESİ				
Başlama Tarihi	**Bitiş Tarihi**	**Proje Süresi (Ay)**	**Onaylanan Ek Süre (Ay)**	**Ek Süre Dahil Bitiş Tarihi**
01.09.2006	01.09.2008	24	-	01.09.2008

PROJENİN GENEL BÜTÇE DURUMU					
Fasıllar	**Sözleşmedeki Ödenek**	**Onaylanan Ek Ödenek**	**Toplam Ödenek**	**Toplam Gerçekleşen Harcama**	**Kalan Ödenek**
Proje Teşvik İkramiyesi	24000	-	24000	15903	8097
Yardımcı Personel	-	-	-	-	-
Bursiyer	62400	-	62400	60900	1500
Sarf Malzemesi	1000	-	1000	0	1000
Seyahat	-	-	-	0	0
Hizmet Alımı	-	-	-	0	0
Makine/Teçhizat	-	-	-	0	0
Kurum Hissesi	12680		12680	12680	0
TOPLAM	**100080**	**0**	**100080**	**89483**	**10597**

RAPOR DÖNEMİNDEKİ BÜTÇE DURUMU						
Fasıllar	**Önceki Dönem(ler)den Kalan Ödenek***	**Dönem İçerisinde Transfer Edilen Ödenek**	**Dönem İçerisinde Transfer Edilen Ek Ödenek**	**Toplam Dönem Ödeneği**	**Toplam Gerçekleşen Harcama**	**Kalan Ödenek**
Proje Teşvik İkramiyesi	0	6.000	-	6.000	6.000	0
Yardımcı Personel	-	-	-	-	-	-

Bursiyer	7.500YTL	10.400	-	10.400	16.400	1.500
Sarf Malzemesi	1.000 YTL	-	-	-	-	1.000
Seyahat	-	-	-	-	-	-
Hizmet Alımı	-	-	-	-	-	-
Makine/Teçhizat	-	-	-	-	-	-
Kurum Hissesi	7.100 YTL	5.200	-	5.200	12.300	0
TOPLAM	**15.600YTL**	**21.600YTL**	-	**21600YTL**	**34.700 YTL**	**2500 YTL**

***Bu sütundaki miktarlar proje hesabına ilk üç dönem içinde aktarılıp, ilk üç dönem içerisinde harcanmadan kalan miktarları göstermektedir.**

TÜBİTAK
ARAŞTIRMA PROJESİ
SONUÇ RAPORU
(*Bilimsel Rapor*)

PROJE NO : 105E133
RAPOR NO : 4
RAPOR DÖNEMİ : 01/03/2008 - 01/09/2008
PROJE YÜRÜTÜCÜSÜ : Asuman Doğaç

BİLİMSEL RAPORDA OLMASI GEREKEN BİLGİLER

1. Dönem içinde projeyle ilgili bilimsel ve teknik gelişmeler proje planı ile karşılaştırılarak verilmeli, elde edilen veriler ile varılan ara sonuçlar, varsa materyal, yöntem ve kapsam değişikleri belirtilmeli ve tartışılmalıdır.
2. Dönem içindeki idari gelişmeler (yardımcı araştırıcı ve personel değişikliği, ek süre, yürütücünün kurum değişikliği ve varsa diğer destekleyen kuruluşlarla sürdürülen işbirliği, vb. konularındaki bilgiler) verilmelidir.
3. Proje çalışmaları kabul edilen çalışma takvimine uygun yürümüyorsa gerekçeleri açıklanmalıdır.
4. Bir sonraki dönem içinde yapılması planlanan çalışmalar (öneri formundan farklı bir durum oluşmuş ise) belirtilmelidir.
5. Destekleyen diğer kuruluşlarla ilgili sorunlar var ise ayrıntıları ve çözüm önerileri sunulmalıdır.
6. Dönem içinde proje kapsamında yapılan yayımların ve toplantılarda sunulan bildirilerin birer kopyası eklenmeli ve yapılan yayımlarda TÜBİTAK desteği belirtilmiş olmalıdır.

BİLİMSEL GELİŞME RAPORU EK SAYFASI
(Proje No: 105E133)

1. Dönem İçinde Projeyle İlgili Bilimsel ve Teknik Gelişmeler

Projenin amaç ve hedefleri şu şekilde özetlenebilir. Avrupa ülkelerindeki demografik yapı incelendiğinde, yaşlı kabul edilebilecek 50 yaş üstü vatandaşların nüfusun çoğunluğu oluşturduğu ve gün geçtikçe nüfus paydasındaki yüzdelerinin artığı gözlenmektedir. Bu demografik yapı, her düzeyden sağlık giderlerini ve sağlık personelinin yükünü bir hayli arttırmaktadır. Son otuz yıldır Avrupa Birliği ülkelerinde uygulanan sağlık hizmetleri sayesinde bu ülkelerde ölüm oranları büyük ölçüde düşürülmüştür. Ölüm oranlarındaki düşüşte sağlanan bu başarı, beraberinde sağlanan sağlık hizmetlerine olan beklentileri ve talepleri de arttırmıştır. Bu etkenlerin yanında bilgi sistemleri, mobil ulaşım araçları, kablosuz sağlık algılayıcıları, ağ ve internet teknolojileri, sağlık personellerine hizmetlerinin uzaktan verilebilmesi için büyük fırsatlar yaratmıştır. Uzaktan sağlanabilecek bu sağlık hizmetleri sayesinde, sağlık personelinin yükü hafifletilebilir ve bu hizmetlerin kalitesi daha da arttırılabilir.

Bu proje çerçevesinde akıllı sağlık takip ve karar destek sistemleri geliştirerek, kablosuz medikal algılayıcılardan elde edilen bilginin günümüzde kullanılan sağlık bilgi sistemlerine bütünleştirilmesini sağlamak amaçlanmıştır. Hasta takibinin, kullanılacak olan 'akıllı ajan' teknolojisi ve bu ajanların davranışlarını belirlemede kullanılacak klinik uygulama kılavuzlarına dayalı olarak çalışacak akıllı karar destek mekanizmaları aracılığı ile gerçekleştirilmesi planlanmıştır. Yapılacak olan sistemde değişik sağlık bilgi sistemlerinde depolanmış hastalara ait gerekli bilgilere, birlikte işlerliği ve bilgi uyumunu sağlayabilmek için anlamsal olarak zenginleştirelecek web servis teknolojileri kullanılarak ulaşılacaktır. Bu sayede medikal algılayıcılardan elde edilen bilgiler ve gözlemler ile birlikte hastanın geçmiş sağlık bilgileri de mantıksal işleme sürecinde kullanılabilecektir.

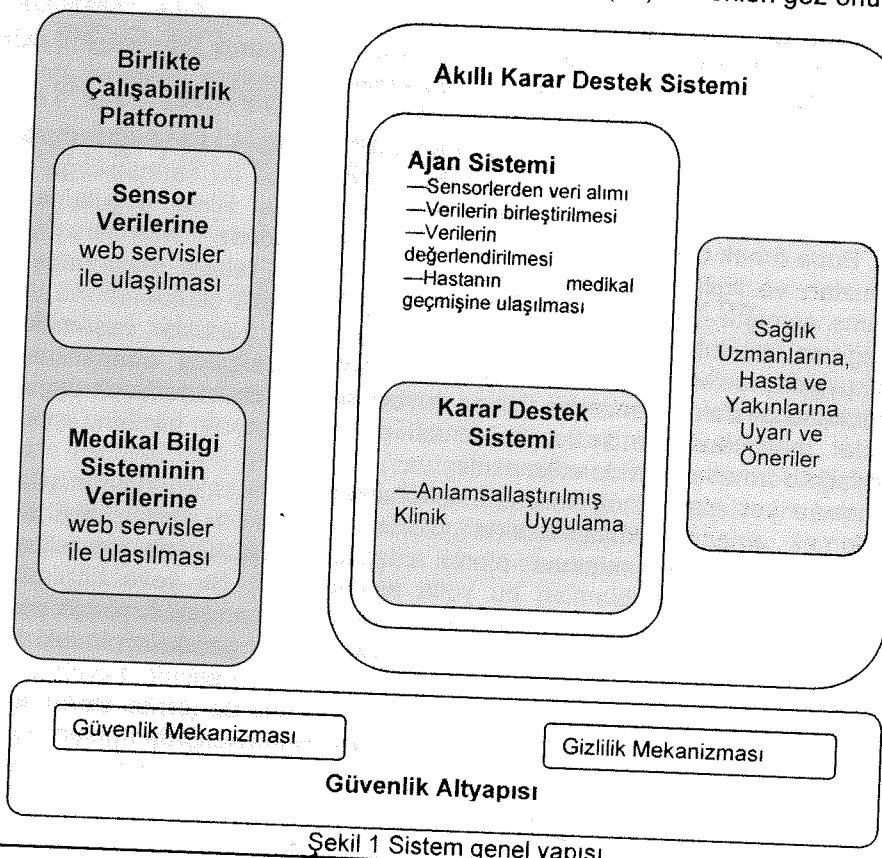
Sağlık hizmetlerine duyulan talebin artmasıyla beraber tıp uygulamaları giderek karmaşıklaşmıştır. Sağlık örgütleri hem tıp uygulamalarında artan bu karmaşıklığa çare olması hem de sağlık hizmetlerini daha da geliştirilmesi amacıyla klinik uygulama kılavuzları geliştirmektedir. Buna örnek olarak, National Guideline Clearinghouse™ belirtilere dayanan klinik uygulama kılavuzları ve ilgili diğer dokümanları içeren bir veri tabanı oluşturmuş ve bu veri tabanını kullanıma açmıştır. Sağlık uzmanları, yaygınlaşmış sağlık standartlarına ve uygulama kılavuzlarına rağmen, bu kılavuzları anlamada ve kullanmada zorluklar yaşamaktadır. Diğer yandan, klinik uygulama kılavuzlarında belirtilen metot ve bilgileri otomatikleştirmek için bilgisayara bağlı karar destek sistemlerine ihtiyaç duyulmaktadır. Bu sistemlerin uygulanmasında da bazı zorluklar mevcuttur, bunlardan bir tanesi de hastaya ait bilgilerin çok farklı bilgi kaynaklarından farklı formatlarda elde edilmesidir.

Bu projede yer alan klinik karar destek mekanizması, ajan davranışlarının sistem içerisinde belirlemek amacıyla kullanılacaktır. Sistem, hastaların sağlık bilgi sistemlerinde depolanmış medikal bilgilerine, anlamsal olarak zenginleştirilecek web servisler aracılığıyla ulaşmakta ve birlikte işlerlik problemleri bu yolla aşılmaktadır. Bu yolla bilgilerin mantıksal işlenme sürecinde sadece algılayıcılardan elde edilen hastanın o andaki fizyolojik belirtileri değil, hastanın geçmiş medikal bilgileri de kullanılmaktadır. Bu bilginin gereksinimindeki temel neden ise klinik uygulama kılavuzlarında bir sonraki klinik yolun seçilmesinde fizyolojik belirtiler ve hasta tedavi planıyla birlikte hastaların geçmiş medikal bilgilerine de (örnek olarak teşhis, tedavi listesi, alerji/ilaç etkileşimleri) ihtiyaç duyulmasıdır. Medikal algılayıcılardan gelen bilgiler ya özel bir formata sahiptir (örneğin, elektrokardiyogram verisi için, Philips' XML ECG veri formatı) ya da belirli bir standarda uygun şekilde oluşturulmaktadır (yine elektrokardiyogram verisi için, mevcut standartlar: SCP-ECP, US Food and Drug Administration FDA/HL7 Annotated ECG, I-Med ve ecgML). Fakat her iki durum da birlikte çalışabilirlik problemine bir çözüm getirmez. Bunun

nedeni kullanılabilecek birden çok standardın mevcut olmasıdır. Bu problem, algılayıcılardan elde edilecek verinin elektronik hasta kaydı ile bütünleştirilmesi ve sağlık bilgi sistemlerinin çok farklı standartlara (HL7v2.x, HL7v3 CDA, CEN ENV 13606 EHRExtract, openEHR Archetypes) uymalarından dolayı daha da karmaşıklaşır. Bu problemlere ek olarak farklı medikal algılayıcılardan gelebilecek verilerin birleştirilmesinde doğabilecek birlikte çalışabilirlik problemlerine de çözüm getirmek gerekir.

Bütün bu birlikte çalışabilirlik problemleri, hem algılayıcılardan elde edilen verilere hem de sağlık bilgi sistemlerinde saklı verilere, işlevsellikleri ve kullandıkları mesaj yapıları ontolojiler yardımı ile belirtilmiş web servisler yardımıyla ulaşılarak çözülmektedir. Böylece bilgi alışverişi sırasında oluşan işlevsel ve anlamsal farklılıklar belirtilen web servisleri ve ontolojileri ile aşılmaktadır. Ontolojiler ve bunların arasında tanımlanan ontoloji eşleşmeleri yardımı ile farklı medikal platformlar arasında anlamsal bütünlük sağlanmaktadır. Böyle bir 'birlikte işlerlik' platformunun oluşturulması bize daha sonra sağlık takip sürecinde karar destek sisteminin oluşturulmasına olanak tanımaktadır. Bütün bunlarla birlikte sistem her hastaya tahsis edilmiş etmenler yardımıyla hastaları sürekli olarak izleyebilmekte ve sağlık uzmanlarına karar destek mekanizması sunarak yardım etmektedir.

Böyle bir bilgi altyapısını kurmak beraberinde hasta bilgilerinin güvenlik ve gizlilik içinde saklanması ve taşınmasını gerektirmektedir. Hastanın kimliği ve medikal kayıtları gelişi güzel bir şekilde saklanamayacağı gibi farklı sağlık sistemi sağlayıcılarının da bu bilgilere ulaşma hakları farklı olacaktır. Projede tüm bunlara çözüm getirecek kapsamlı güvenlik ve gizlilik mekanizmaları önerilmektedir. Bu mekanizmaları sağlarken kişisel bilgilerin nasıl işleneceğini anlatan 95/46/EC ve 2002/58/EC ve tıbbi verilerin saklanmasında önerilen R(97)5 önerileri göz önüne alınmıştır.



Şekil 1 Sistem genel yapısı

Çalışma Takviminde belirtilen ve dördüncü gelişme dönemini kapsayan süre zarfında, Tablo 1'de verilen görevler üzerinde çalışılmışır.

Başlıca Aşamalar	Ayrıntılı Bilgi	Zamanlama
Uyarı Sistemi	Bu sistem PDA, mobil telefonlar ve diğer iletişim cihazlarını kullanarak gerekli kişilere hastanın aktivitelerini ve kritik durumları haber verecektir. Sistem, kullanıcı yönetme alt componenti ile kimin uyarılması gerektiğini, kullanılacak alete göre veri sunum formatının değiştirilmesini, uyarının gönderilme garantisini,	18-20
Tüm sistemin entegrasyonu	Geliştirilen tüm yazılım parçalarının entegrasyonu gerçekleştirilecek.	20-24

Tablo 1

Yukarıdaki çalışma plana göre elde edilen bilimsel ve teknik gelişmeler detaylı olarak ilgili kısımlarda açıklanmıştır.

1. *Uyarı Sistemi*

Proje kapsamında geliştirilmesi tasarlanan Uyarı Sistemi, hastalarla ilgili dikkat gereken uyarıların ya da acil durumların, sağlık personeline otomatik olarak bildirilmesini sağlayan alarm mekanizmalarını içermektedir. Proje kapsamında geliştirilen uyarı sistemi, çoklu etmen platformu ve kural tabanlı karar mekanizmaları üzerine kuruludur. Aşağıdaki bölümde; proje kapsamında geliştirilecek olan uyarı sisteminin gereksinimleri verilmektedir.

Uyarı Sistemi Gereksinimleri

Uyarı Sistemi; proje kapsamında geliştirilen Akıllı Klinik Karar Destek Sistemi'nin bir uzantısı durumundadır. Sistemin genel amacı; özel dikkat ve yardım gerekebilecek durumlarda; ilgili insanları uyaracak eylemleri gerçekleştirebilmektir. Bu doğrultuda, uyarı sistemini oluşturan yazılım bileşenleri şunlardır:

- Web tabanlı kullanıcı arayüzü
- İletişim Katmanı
- Alarm Yöneticisi
- Veri Modeli

Uyarı sisteminin kontrolü, geliştirilen web tabanlı kullanıcı arayüzü sayesinde gerçekleştirilecektir. Bu arayüz ile, sistemdeki kullanıcıların yönetimi sağlanacak ilgili sağlık personelinin hastalara atanması yine bu arayüz üzerinden olacaktır; kullanıcılar kişisel bilgilerini, iletişim bilgilerini ve uyarı mesajları ile ilgili tercihlerini bu arayüz yoluyla sisteme kaydedeceklerdir.

Uyası sisteminin farklı iletişim kanallarını destekleyerek; kullanıcılara alarm mesajlarının alımı için birden çok seçenek sunması; temel bir gereksinimdir. İletilecek olan uyarı mesajının ciddiyet ve aciliyetine göre; mesajların iletişim kanalı değiştirililebilmelidir. Örneğin, mobil telefonların hızlı bir iletişim gereci olduğu göz önüne alınarak; acil mesajların SMS yoluyla iletilirken; aciliyeti daha az olan uyarıların e-posta yoluyla gönderilmesi tercih edilebilir bir seçenektir. Buna ek olarak, kullanıcı tercihleri dikkate alınarak anlık mesajlaşma (Instant Messaging) yöntemleri de makul seçenekler olarak öne çıkmaktadır. Geliştirilen uyarı sisteminin bu üç seçeneği kapsaması öngörülmüştür.

Sistemin bir diğer gereksinimi de gönderilen uyarı mesajlarının gizlilik gereksinimin karşılanmasıdır. Hasta ile sağlık bilgilerinin hassasiyetinin yüksek olması; uyarı sistemdeki güvenlik ve gizlilik gereksinimini ortaya çıkarmaktadır. Sistemde yer alması planlanan SMS ve anlık mesajlaşma protokolleri, kendi içlerinde bazı güvenlik tedbirleri barındırsalar da; kötü niyetli kullanıma daha açık olan e-posta'ların, güvenli bir şekilde iletiminin sağlanması ele alınması gereken önemli bir konudur.

Uyarı mesajlarının, alıcıya vaktinde ve kısa bir zaman diliminde ulaştırılması da sistemde vazgeçilemeyecek bir diğer gereksinimdir; yukarıda da değinildiği gibi sistemde kullanıcıya birden çok mesaj kanalı önerilmekte olup, gönderilen mesajların kullanıcıya ulaşıp ulaşmadığının kontrolü ve gereken durumlarda, alarm mesajının tekrar gönderilmesi veya başka bir sağlık personeli ya da ilgili kişiye yönlendirilebiliyor olması dikkate alınması gereken gereksinimlerdir.

Bütün bunlara ek olarak; sistemin birden fazla uyarı mesajını aynı anda iletebiliyor olması ve oluşan uyarıların arşivlenmesi sistemin diğer gereksinimleridir.

Uyarı Sistemi Bileşenleri

Uyarı sistemi bileşenleri; yukarıda belirtilen gereksinimlerin ışığında analiz edildi ve tasarlandı. Sistemde aşağıdaki aktörlerin yer alması planlanmaktadır.

- **Sistem Yöneticisi:** Sistemdeki yetkili kullanıcıdır. Bu tipte kullanıcılar; web tabanlı arayüz aracılığıyla sistemdeki diğer kullanıcıların yönetimini gerçekleştirirler. Buna ek olarak; hastalara ilgili sağlık personeli atamasını da 'Sistem Yöneticisi' olan kullanıcılar yapar.

- **Kullanıcılar:** Doktor ve hemşirelerden oluşan, sistemin kullanıcısı durumunda olan sağlık personelidir. Sistem yöneticisi tarafından sisteme eklenir ve hastalara atanırlar. Kullancılar, web arayüzü aracılığıyla; uyarı mesajlarını almadaki tercihlerini(iletişim yolu, tekrar sayısı, mesaj alındı onayı, yönlendirme vs.) belirleyebilir; iletişim bilgilerini değiştirebilir ayrıca sistem kayıtlarından önceki uyarı mesajlarını görüntüleyebilirler.

- **Alarm Dağıtım Etmeni:** Akıllı Klinik Karar Destek Sistemi çalıştırılırken otomatik olarak yaratılan ve sistem çalıştığı sürece, aktif olan bir bileşendir. Sistemdeki diğer etmenler gibi JADE platformuna bağlı olarak işlev görmektedir. Alarm Dağıtım Etmeni'nin görevi; kılavuzlar çalışırken kılavuz etmenlerinden gelen alarm mesajlarını kabul ederek; mesajları ilgili hastaya atanmış olan sağlık personeline ulaştırmaktır. Mesajlar ulaştırılırken; alıcının mesaj alımında daha önceden belirtmiş olduğu tercihler göz önüne alınarak, kural tabanlı bir işlemle; mesajın hangi yolla iletileceğine karar verilir; mesajlar

alıcısına ulaşmadığı durumda, gerekiyorsa başka bir sağlık kullanıcısına yönendirilir. Alarm Dağıtım Etmeni aynı zamanda mesajların daha sonra web arayüzünden görüntülenmek üzere veri tabanına kaydedilmesinden de sorumludur.

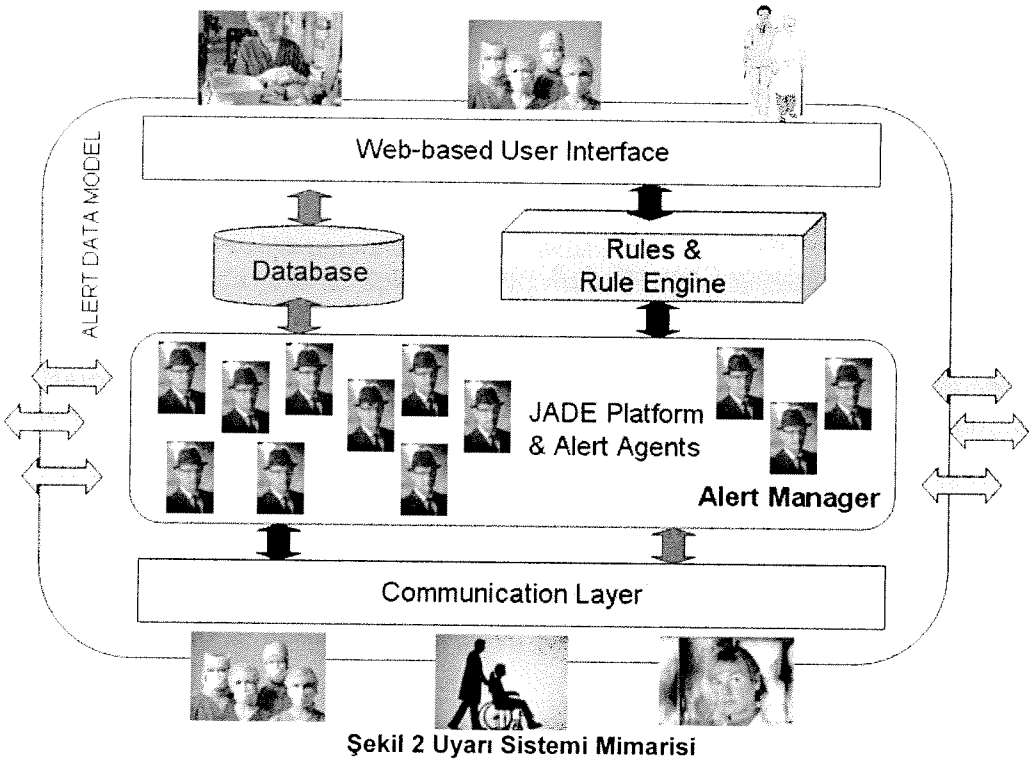
- **Alarm Tetikleme Bileşeni:** Kılavuzlardaki mesaj adımlarından gelmeyip; sistemdeki sensor verilerinin belirtilen eşik değerini aşması sonucunda oluşan alarm mesajlarını kullanıcıya ulaştırmakla yükümlü olan bileşendir. Sensor verileri etmen sistemi yoluyla gelmediğinden bu bileşen bir etmen olarak tasarlanmamıştır.

Veri Modeli, Uyarı Sistemi'ndeki verilerin tutulmasını sağlayan anlamsal meta-data katmanıdır. Sistemdeki genel anlamsallığı ve bilgileri yönetmekle yükümlüdür. Bu sistemin gerçekleştirimi, veri tabanları, kurallar ve kural tabanlı motorlar (Rule Engine) üzerine kuruludur.

Buna ek olarak, mesajların iletilmesinden sorumlu olan **İletişim Katmanı**, hem Alarm Dağıtım Etmeni hem de Alarm Tetikleme Bileşeni tarafınan kullanıldığından; diğer bileşenlerden ayrık bir biçimde tasarlanmıştır. Bu bileşenin SMS, e-posta ve *GoogleTalk* anlık mesaj protokollerini desteklemesi düşünülmüştür.

Web tabanlı kullanıcı arayüzü, kullanıcıların sistemle etkileşimini sağlayan bir ön uç vazifesi görmekte olup sisteme ait tek görsel bileşendir. Kullanıcılarının, her an hastane ortamında olmayabileceği ve sisteme uzaktan erişmeleri gerekebileceği düşünülerek; arayüz web tabanlı olarak düşünülmüştür.

Sistemin genel yapısı aşağıdaki şekilde görülmektedir:



Şekil 2 Uyarı Sistemi Mimarisi

Uyarı Sistemi'nin Gerçekleştirimi

Uyarı sisteminin gereksinimlerinin belirlenmesi ve genel tasarımının yapılması, projenin üçüncü döneminde gerçekleştirilmiş ve uygulamaya başlanmıştır; uygulamanın, proje iş planında da belirtildiği gibi dördüncü gelişme döneminde tamamlanması öngörülmüştür. Bu bölümde, üçüncü gelişme döneminde gerçekleştirimi tamamlanan **Veri Modeli** ile ilgili bilgi verilecektir.

Veri Modeli

Alarm sisteminin veri modeli kural taban, kural motoru ve ilişkisel veritabanları üzerine kurulmuştur. Veritabanı, alarm istemindeki verilerin saklanmasını sağlarken; kural motoru ve kural tabanı kullanıcıya ulaştırılacak mesajların nasıl iletileceğini belirlemekte kullanılır.

Kural Tabanı ve Kural Motoru

Ulaştırılacak alarmların iletim koşulları, kulanıcının daha önce belirttiği tercihlerin kurallar şeklinde saklanması ve bu kuralların kural motoru üzerinde işletilmesiyle belirlenir. Analiz ve tasarım aşamasında pek çok kural tabanlı sistem incelenmiş, JESS[1] kural motorunun kullanılmasında karar kılınmıştır. JADE platformu ile olan uyumu ve kural motorunun olgunlaşmış olması JESS'in seçiminde etkili olmuştur. Aşağıda örnek bir JESS kural şablonu ve bu şablona uyan bir kural görülmektedir.

```
IF
userid = {INTEGER},
assignmentid= {INTEGER}
urgency = { LOW(1),MEDIUM(2),HIGH(3)} THEN
                    mediumid = {GoogleTalk, E-mail, SMS},
                    needack={TRUE,FALSE}
                    numberoftry={INTEGER}
                    waitduration={INTEGER}
                    mustsend={TRUE,FALSE}


(defrule decide-urgency-Contact1_1_3
    (userid 1)(assignmentid 1)(urgencyid 3)=>
                    (store mediumid 3)
                    (store needack true)
                    (store numberoftry 4)
                    (store waitduration 3)
                    (store mustsend true)
)
```

Yukarıdaki kural, 1 numaralı kullanıcıya, 1 numaralı hasta atamasıyla ilgili, aciliyeti 3(yüksek) olan bir uyarı gönderileceği zamanki iletim tercihleri belirtmektedir.

- **Mediumid:** Mesajın hangi kanalla iletileciğini belirtir. 3 numaralı kanal, sistemde GoogleTalk'u işaret etmektedir.
- **Needack:** Mesajın alıcı tarafından onaylanması gerekip gerekmediğini belirtir. Örnek

[1] http://www.jessrules.com/

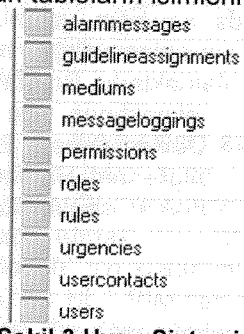
kuraldaki mesajın onaylanması gereklidir.

- **Numberoftry:** Mesajın kaç defa gönderilmesi gerektiğini belirtir.
- **Waitduration:** Mesajın iki gönderimi arasında kaç dakikalık bekleme olacağını belirtir.
- **Mustsend:** Mesajın iletim onayının alınmaması durumunda başka bir kullanıcıya yönlendirilip yönlendirilmeyeceğini belirtir.

Sistemde tanımlı olan kurallar kalıcı bir şekilde, uyarı sistemi veritabanındaki *Rules* tablosunda saklanmaktadır.

Uyarı Sistemi Veritabanı

Uyarı sisteminde kullanılan verilerin ve de oluşturulan kuralların ilişkisel bir veritabanında saklanması düşünülmüş bu çerçevede, MySQL[2] veritabanında, gerekli tablolar oluşturulmuştur. Aşağıdaki şekil, oluşturulan tabloların isimlerini göstermektedir.



Şekil 3 Uyarı Sistemi Veri Modelindeki Tablolar

- **AlarmMessages** tablosu, sisteme gelen uyarı mesajlarını depolar
- **GuidelineAssignments** tablosu, hasta-kılavuz eşleşmelerine atanmış olan doktor ve hemşirelerin id'lerini saklar. Hasta bilgilerine, EHR veritabanından ulaşılır.
- **MessageLoggings** tablosu kullanıcılara gönderilmiş olan mesajları depolar.
- **Mediums** tablosunda, eldeki iletim yolları kayıtlıdır. 1, GoogleTalka; 2, e-posta'ya; 3 ise SMS'e karşılık gelmektedir.
- **Permissions** tablosu, eğer tanımlandıysa; bir kullanıcıya mesaj ulaşmadığında mesajın yönlendirileceği sağlık personeli listesini saklar.
- **Roles** tablosunda kullanıcı tipleri(Yönetici,oktor,Hemşire,Hasta yakını) kayıtlıdır.
- **Rules** tablosunda, JESS ile tanımlanmış kurallar saklanır.
- **Urgencies** tablosu, kaytlı olan aciliyet kodlarının isimleriyle birlikte saklandığı tablodur.
- **Usercontacts** tablosu kullanıcıların tanımladığı cep telefonu, GoogleTalk ve e-posta adres ve numaralarını saklar.
- **Users** tablosu, kullanıcı bilgilerini saklar.

Pilot Uygulama için Güncellemeler

Son kullanıcılardan gelen geribildirimler göz önünde bulundurularak Uyarı Sisteminde aşağıdaki güncellemeler gerçekleştirilmiştir:

- Prototiplerde daha önce SMS uyarılarını göndermekte kullanılan SMS servis sağlayıcısı sadece Türkiye GSM operatörlerine SMS mesajları gönderimini desteklemekteydi. Bu kısıtlamanın giderilmesi için, dünya çapında bir dağıtım ağına sahip bir SMS servis

[2] http://www.mysql.com/

sağlayıcısı (Clickatell) için yeni bir SMS uyarı adaptörü geliştirildi ve sisteme dahil edildi.

- Son kullanıcıların geribildirimlerine paralel olarak, uyarı sisteminin yapılandırılmasında daha basit bir yapılandırma seçeneği sisteme dahil edildi. Bu seçenek özellikle küçük çaplı organizasyonlar için kolaylık sağlayacak.

2. *Tüm sistemin entegrasyonu*

Anlamsal Birlikte-işlerlik

Anlamsal olarak zenginleştirilmiş web servisleri yardımıyla anlamsal birlikte-işlerlik sensor web servislerinin anlamsal mimlenmesiyle sağlanmaktadır. İçerik seviyesindeki birlikte-işlerlik çözümü için, sistemde kullanılan sensor bilgisini ifade ederken sistemimizi IEEE 11073 ("Health informatics, Point of care medical device communication") standard ailesine dayandırdık. Bu standardlar, formatların birbirine çevrimi sırasında kayıpsız verimli bir tıbbi bilgi alışverişi sağlamaktadır. IEEE 11073 standardı sensor bilgisinin anlamsallığını bir Alan Bilgi Modeli (ABM) ve bunu tamamlayıcı bir terminoloji yardımıyla tanımlamaktadır. Bu standardları kullanarak, ortak bir tıbbi araç ifade standardından faydalanmış olmaktayız. Bu ABM ve IEEE 11073 standardlarına ek olarak, ayrıca IHE Patient Care Device (Hasta Bakım Aracı) entegrasyon profilleri IEEE 11073 ABM'de ifade edilen sensor bilgilerinin HL mesajlarına çevrimi için bir mekanizma sunmaktadır. Böylece sensor cihaz bilgisinin sağlık kurumlarında entegrasyonu sağlanmaktadır.

ECG Analizlerinin Bilgi Dönüşümü

ECG Analizleri üçüncü parti bir kurum tarafından gerçekleştirilir. Bu yüzden, sensor web servislerinin ECG bilgisi için anlamsal mimlenmesi kurumlara özel bilgi dönüşümü ile gerçekleştirilir. Bunun doğal bir sonucu olarak, ECG Web servisleri, SpO2 ve BP sensorlerinde olan gerçek analizleri sağlamak yerine, zaten analiz edilmiş gözlemleri sağlamaktadır.
Sistemdeki yönergelerde aşağıdaki parametreler kullanılmaktadır:

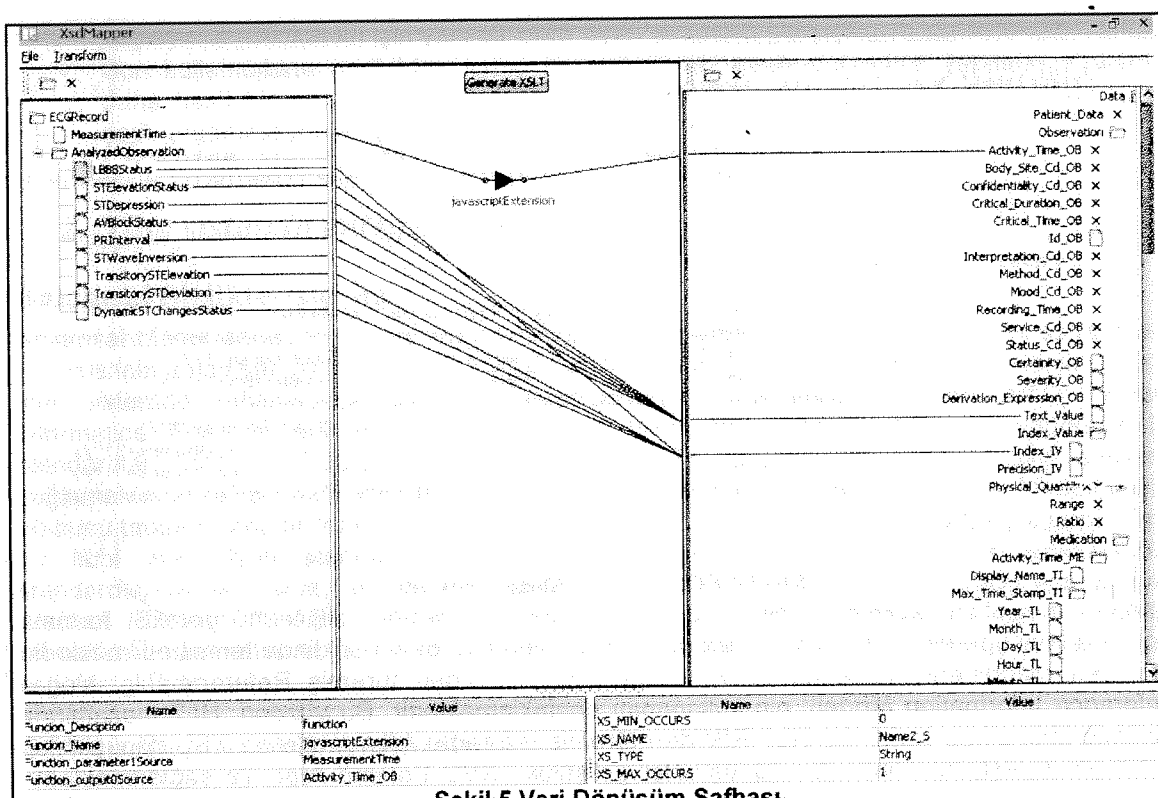
- ECG New LBBB Durumu
- ECG ST Elevation Durumu
- ST Depression Değeri
- ECG AV Block Durumu
- PR Değeri
- ST-wave inversion
- Transitory ST Elevation
- Transitory ST Deviation Durumu
- Dynamic ST Changes Durumu

Bu parametreleri ifade etmek için kullanılan kuruma özel XSD Şekil 4'te gösterilmektedir.

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">
        <xsd:element name="ECGRecord" minOccurs="0" maxOccurs="1">
                <xsd:complexType>
                        <xsd:sequence>
                                <xsd:element name="MeasurementTime" type="xsd:date" maxOccurs="1"/>
                                <xsd:element name="AnalyzedObservation" type="ECGAnalysis" minOccurs="1"/
                        </xsd:sequence>
                </xsd:complexType>
        </xsd:element>
        <xsd:complexType name="ECGAnalysis">
                <xsd:sequence>
                        <xsd:element name="LBBBStatus" type="xsd:string" maxOccurs="1"/>
                        <xsd:element name="STElevationStatus" type="xsd:string" maxOccurs="1"/>
                        <xsd:element name="STDepression" type="xsd:string" maxOccurs="1"/>
                        <xsd:element name="AVBlockStatus" type="xsd:string" maxOccurs="1"/>
                        <xsd:element name="PRInterval" type="xsd:string" maxOccurs="1"/>
                        <xsd:element name="STWaveInversion" type="xsd:string" maxOccurs="1"/>
                        <xsd:element name="TransitorySTElevation" type="xsd:string" maxOccurs="1"/>
                        <xsd:element name="TransitorySTDeviation" type="xsd:string" maxOccurs="1"/>
                        <xsd:element name="DynamicSTChangesStatus" type="xsd:string" maxOccurs="1"/>
                </xsd:sequence>
        </xsd:complexType>
</xsd:schema>
```

Şekil 4 ECG Gözlem Mesaj Formatı

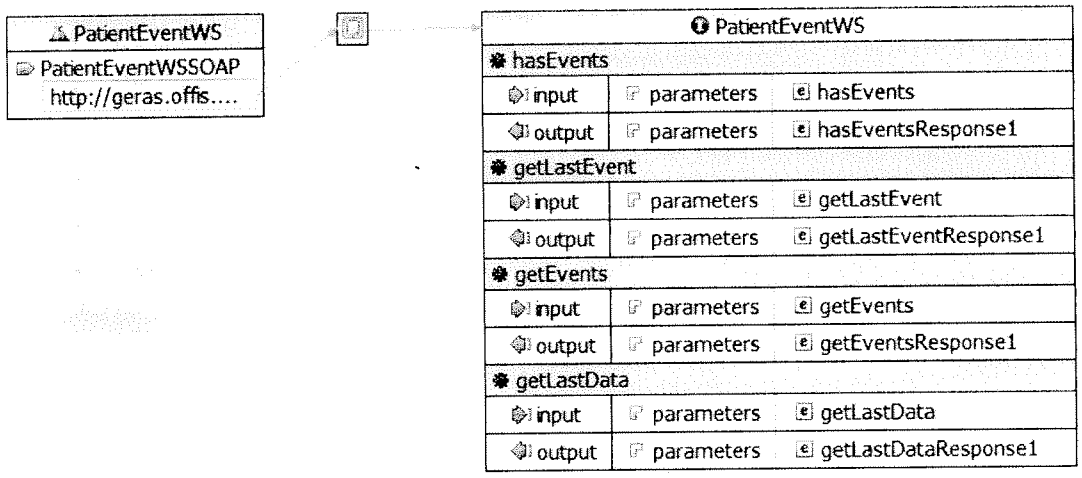
Yönergenin çalıştırılabilmesi için, tıbbi cihazların sağladığı bilgilerin gerekli formata dönüştürülmesi gerekmektedir. GLIF'teki klinik bilgi, veri maddeleri şeklinde temsil edilmektedir. Tıbbi veri maddelerini ifade etmek için, GLIF, HL7 RIM'den türemiş Referans Bilgi Modeli (Reference Information Model)' nin kullanımını desteklemektedir. Bu yüzden, HL7 mesajlarına çevrilmiş tıbbi cihaz verisi, bu HL7 RIM'den türemiş referenas bilgi modeline XSL dönüşümleri yardımıyla eşleştirilmektedir. Bu dönüşümler geliştirilen XSD Eşleme Aracı ile sağlanmaktadır. Bu araç yardımıyla üretilen tanımlar, bir XSL Dönüşüm Motoru yardımıyla verinin bir formattan diğerine dönüştürülmesinde kullanılır. ECG Analiz parametreleri için eşleme tanımlama süreci Şekil 5'te gösterilmiştir.



Şekil 5 Veri Dönüşüm Safhası

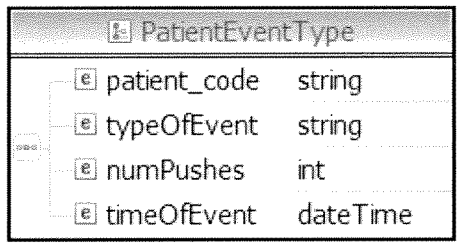
Hasta Vakaları için Web Servisleri

Yönergenin çalışması için, hasta gözetimi kalitesinin yükseltilmesi amacıyla, hastanın vaka düğmesinin tetiklediği vakaların kaydedilmesi gereklidir. Bu veri, web servisleri aracılı ile ulaşılır hale gelmektedir.



Şekil 6 Hasta Vaka Web Servisi Operasyonları

PatientEventType veri formatı Şekil 7'de gösterilmektedir. Projede kullanılan yönergelerin çalışması için sadece "Göğüs Ağrısı" vaka türü önemlidir. Bu vaka türü, gerektiğinde vaka düğmesine bir defa basılarak oluşturulur. Vaka türünün iki defa basılması durumunda 'Dyspnoea', üç defa basılması durumunda ise 'Palpitations' vaka türleri oluşturularak yönergeye iletirilir.



Şekil 7 PatientEventType Veri formatı

Uyarı Sistemi Entegrasyonu

"Uyarı Sistemi" bölümünde anlatıldığı gibi, sistemin uyarı bileşeni, ajan teknolojilerine dayalıdır. Uyarı Dağıtım Ajanı, bu çok ajanlı sistemdeki ajanlardan bir tanesidir. Sistemdeki Uyarı Bileşeni, Yönerge Çalıştırım Bileşeni ve kablosuz tıbbi sensor Birlikte-İşlerlik Platformu ile etkileşim halindedir. Tüm sistemin verimli ve tam anlamıyla çalışabilir olması için Uyarı bileşeni bu diğer bileşenlerle tamamen entegre olmak zorundadır.

Ajanların çevreiyle iletişimi, ajan mesajlaşma imkanları ile gerçekleştirilmektedir. JADE ajanlarının kullanıldığı bu sistemde, Ajan İletişim Dili (Agent Communication Language (ACL)) mesajları ajanlar arasındaki iletişimi sağlar. Yönerge Çalıştırım Bileşeni, Yönerge Ajanı dolayısıyla Sistem Ajan Platformunun bir üyesi olduğu için, Uyarı Dağıtım Ajanı ile aralarındaki iletişim doğrudan ACL mesajları kullanılarak gerçekleştirilir. Ancak, bir ajanlardan harici bileşen olan Birlikte-İşlerlik Platformunun sensor uyarılarının dağıtılması için, Uyarı Dağıtım Ajanına bağlanması gereklidir. Bir sonraki bölümde sensor uyarılarının Uyarı Dağıtım Ajanına nasıl iletildiği anlatılmaktadır.

Sensor Uyarılarının Dağıtımı

Sistem tarafından oluşturulan iki tür uyarı vardır. Bunlardan ilki, Hastane Pilot Uygulamalarına özgü olan klinik yönerge uyarı türüdür. Bu uyarılar klinik yönerge tanımlarında önceden belirlenmiştir ve dinamik olarak değiştirilemezler. Uyarı oluşturacak koşullar yönerge algoritmasının akışı ile belirlenir.

Diğer taraftan, hem Hastane hem de Evde Bakım Pilot Uygulamaları tarafından kullanılan bir diğer uyarı türü vardır: Sensor uyarıları. Sensor uyarıları sadece sensor verisi kaynaklı koşulların uyarılarının bildirimi için kullanılır.

Uyarı Dağıtım Ajanı ve Yönerge Ajanı aynı JADE birimi üzerinde konuşlandırıldığından bu iki bileşen arasındaki iletişim kolaylıkla ACL mesajları yardımıyla gerçekleştirilebilir. Ancak, sensor uyarıları için durum farklıdır. Sensor Uyarıları, sistemin ajan platformundan bir başka platformda konuşlandırılmış Birlikte-İşlerlik Platformu içerisinde üretilir. Bu yüzden, ACL mesajları sensor uyarılarının Uyarı Bileşenine iletilmesi için elverişli değildir. Buna ek olarak, sensor uyarıları sensor veritabanında tutulmaktadır; bunun bir sonucu olarak da, sensor veritabanından sensor uyarılarını toplayabilmek için, ilgili tabloya veri girişiyle tetiklenen bir mekanizma sağlanması gereklidir.

Sensor veritabanı 39 tablodan oluşan bir ilişkisel veritabanıdır. Veritabanı MySQL sunucusunda bulunmaktadır. Tüm tabloların arasında, sensor uyarıları için en önemli görevi üstlenen tablo

"Alarm" tablosudur. Alarm tablosunun yapısı aşağıdaki şekilde gösterildiği gibidir:

Column Name	Datatype
AlarmID	INTEGER
patient_id	INTEGER
AlarmMessage	VARCHAR(2000)
GuidelineID	INTEGER
Date	DATETIME
ConfidenceIndex	FLOAT
SeverityIndex	FLOAT
AlarmThreshold	VARCHAR(1000)
Distributed	BIT(1)
Resolved	TINYINT(3)

Şekil 8 Alarm Tablosunun Yapısı

- **AlarmID** veritabanı tarafından otomatik olarak verilen belirleyicidir.
- **Patient_id** sensor verisetinde hasta kimliğini belirtir.
- **Alarm Message** uyarıcı alıcılarına gönderilen metindir.
- **Guideline ID** Hastane Pilot Uygulamalarında kullanılan uyarı mesajının ilişkili olduğu yönergenin belirleyicisidir. Ev-bakımı Pilot Uygulama içinse varsayılan değeri kullanılır.
- **Date** uyarının oluşturulduğu tarihtir.
- **Confidence Index** sensor verisinin kesinliğini belirtir.
- **Severity Index** uyarı aciliyetinin bir ölçümüdür.
- **Alarm Threshold** eşikdeğerini aşan sensor verisinin ismini gösterir.
- **Distributed** and **Resolved** uyarı mesajlarının gönderim başarısını belirtir.

"Alarm" tablosuna yeni bir giriş durumunda bir tetikleme mekanizması aktif hale gelir. Bu tetikleme mekanizması ilk olarak C dili ile önceden tanımlanmış DLL dosyasının bir fonksiyonunu çağırır. Bu DLL'deki fonksiyonun aktifleşmesi ile birlikte, ortamda bir Java Sanal Makinası oluşturarak bir java uygulaması başlatılır. Bu java uygulaması uyarı mesajını göndermek için öncelikle Alarm Tablosundan aşağıdaki parametreleri çeker:

- Patient ID
- Alarm Message
- Guideline ID
- Date
- Confidence Index
- Severity Index

Bu parametreleri elde ettikten sonra, Java uygulaması, Uyarı Bileşeni ile web servisleri aracılığıyla haberleşir. Bu amaçla, Uyarı Bileşeni tarafında bir web servisi geliştirilmiş ve konuşlandırılmıştır. Uyarıyı oluşturan taraf, sensor veritabanı tarafı, web serivsinin *agentCaller()* operasyonunu yukardaki parametrelere güvenirlik ve eşikdeğeri ekleyerek çağırır. Uyarı

iletiminin geri kalanı sunucu tarafında gerçekleştirilir.

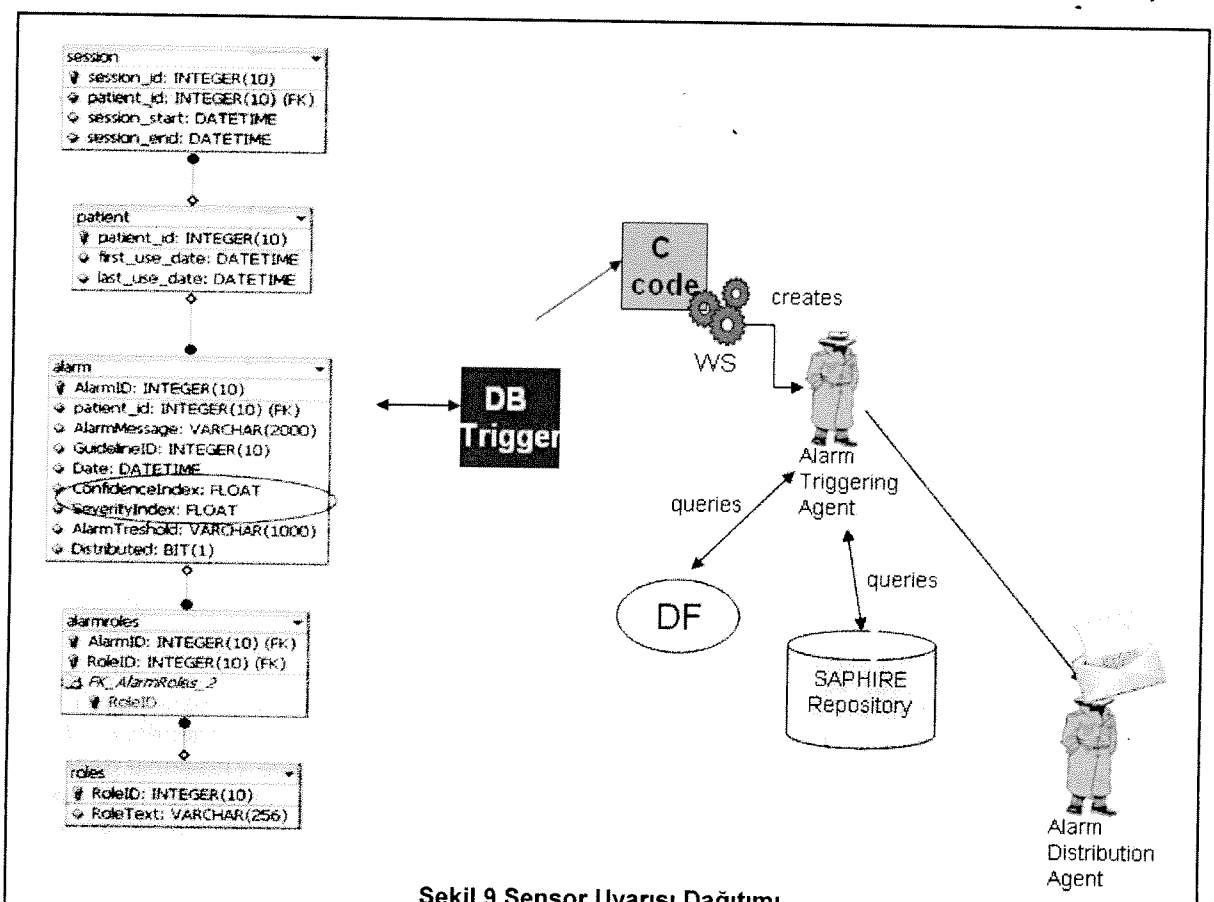
Uyarı Dağıtım bileşeni bir ajan olduğu için, bu bileşenle haberleşmenin de ajanlar tarafından yapılması gerekmektedir. Web Servisinin çalışmaya başlamasıyla birlikte, tek kullanımlık bir ajan, ajan haberleşmesini kolaylaştırmak amacıyla oluşturulur. Uyarı Ontolojisine uygun aşağıdaki yapıyla bir ACL mesajı oluşturulur:

- Alarm Message :
  - Patient Info
    - Assignment ID (STRING)
    - Patient ID (STRING)
    - Guideline ID (STRING)
    - Patient Name (STRING)
  - Message Content
    - Urgency (INTEGER)
    - Role (STRING)
    - Content (STRING)

Mesajın aciliyetine Confidence Index, Severity Index ve Confidence Threshold parametrelerinin beraber yorumlanmasıyla karar verilir. Yukarda bahsedildiği üzere, Confidence index parametresi sensor verisinin güvenirliğini belirtir. Eğer bu değer Confidence Threshold değerine (Sistemde 0,5 olarak belirlendi) eşit veya bu değerden daha fazla ise, mesaj aciliyeti olarak Severity Index kullanılır. Diğer türlü, aciliyet Severity Index değerinin bir eksiği olarak belirlenir. Aşağıdaki tablo bu aciliyet karar listesini göstermektedir.

SI	CI	URGENCY
0	CI<0.5	GREEN
0	CI>=0.5	GREEN
1	CI<0.5	GREEN
1	CI>=0.5	YELLOW
2	CI<0.5	YELLOW
2	CI>=0.5	RED

Ajan tarafından mesaj oluşturulduktan sonra, mesaj Uyarı Dağıtım Ajanına gönderilir. Mesaj gönderildikten sonra, tek kullanımlık ajan yokedilir.
Sensor Uyarılarının dağıtımının genel bakışı Şekil 9'de gösterilmektedir.

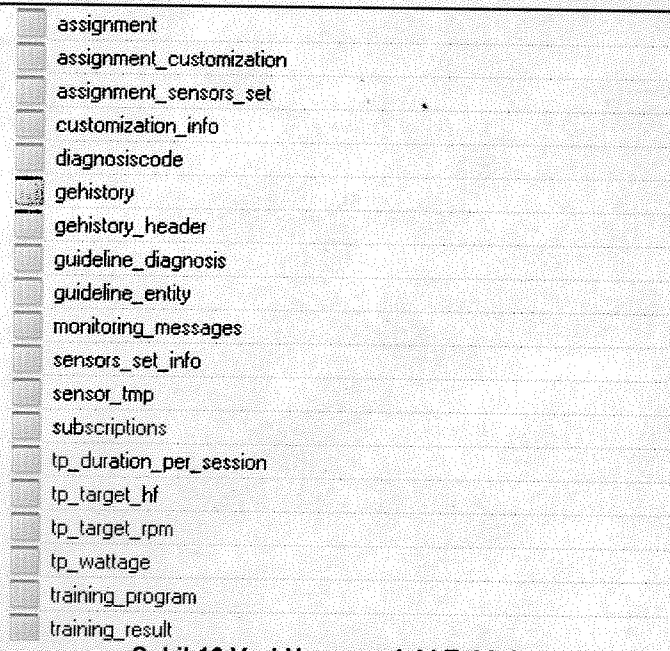


Şekil 9 Sensor Uyarısı Dağıtımı

Veritabanlarının Entegrasyonu

Farklı sunucularda bulunan birkaç veritabanı ve projenin farklı gereksinimleri için konfigurasyonları ile sistem heterojen bir ortamdan oluşmaktadır. Projenin very havuzundaki en başta gelen iki veritabanı, EHR Veritabanı ve Sensor Veritabanıdır.

Sistemin tüm bileşenlerinin aralarında bilgi paylaşmasını sağlayan bir veri havuzu bulunmaktadır. Bu havuz arka tarafta bir İlişkisel Veritabanı Yönetim Sistemi (RDBMS) ve bu veritabanından bilgi çekilmesini ve veritabanına bilgi girişini sağlayan bir java arayüzünden oluşmaktadır. Şekil 10'da bu veri havuzu tabloları gösterilmektedir.

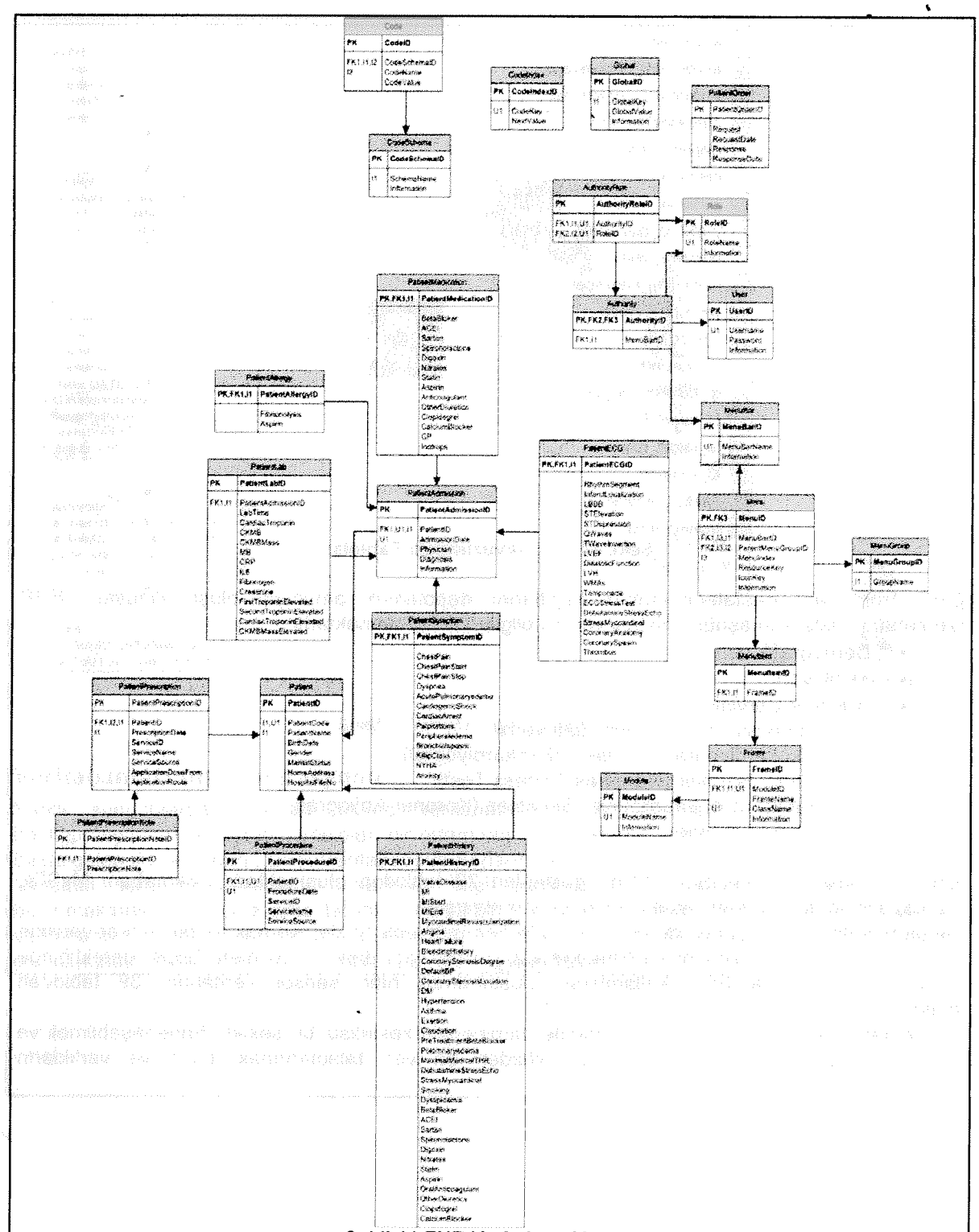


Şekil 10 Veri Havuzundaki Tablolar

EHR Veritabanı hastaların sağlık kayıtlarını depolamak amacıyla oluşturulmuştur. EHR Veritabanı hastaların aşağıda listesi verilen bilgilerini depolamaktadır.

- Demografik veri
- Klinik veri
- Hastane Servisleri
  - Laboratuvar Servisleri (Lab verisi)
  - Ekokardiyografi Servisleri (Ekokardiyografi)
  - Farklı İşlemler (Nonefraktif stress Testi)
  - Kateterizim Laboratuvar Servisleri (Koroner Anjiografi)
  - Efraktif İzleme

EHR Veritabanı, yapısı Şekil 11'de gösterilen 22 tablodan oluşmaktadır. Veritabanı MySQL Veritabanı Yönetim Sistemi kullanılarak oluşturulmuştur.
Sensor veritabanının görevi de tıbbi sensorlerden gelen bilgiyi depolamak ve bu sensor verisine ihtiyaç duyan Uyarı Bileşeni ve Yönerge Çalıştırıcısı gibi diğer bileşenlere bilgiyi gerektiğinde sağlamaktır. Yine MySQL kullanılarak oluşturulmuş olan sensor veritabanı 39 tablodan oluşmaktadır.
Bu veritabanlarından faydalanan bileşenler birbirleriyle kesintisiz bir şekilde haberleşebilmeli ve veri alışverişinde bulunabilmelidir. Bu yüzden bu veri tablolarındaki tablo ve varlıkların senkronizasyonu sağlanmıştır.

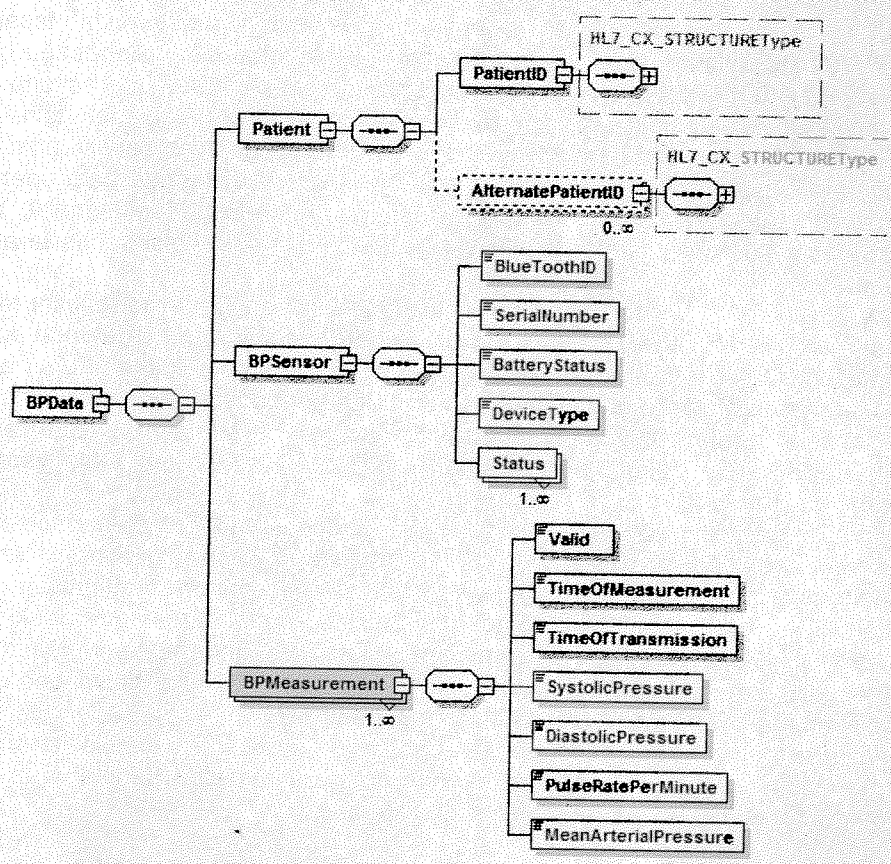


Şekil 11 EHR Veritabanı Yapısı

Sensor Web Servisleri Entegrasyonu

Sistem Yönerge Çalıştırma Bileşeni, EHR ve sensorlerden elde edilen verilere dayanan bir karar destek sistemidir. EHR verisi CDA dosyaları aracılığıyla ulaşılırken, sensor verisi web servislerinden alınmaktadır. Sensor verisi Sistem Birlikte-İşlerlik Platformunda işlenir ve analiz edilir, daha sonra anlamsal olarak detaylandırılmış web servisleri aracılığıyla ulaşılabilir hale gelir.

Sistemde hastanın gözetimi için üç çeşit sensor kullanılmaktadır: Kan Basıncı (Blood Pressure, BP), Nabız Ölçüm Cihazı (SpO2) ve Elektrokardiogram (ECG). Örnek olarak Kan basıncı sensor veri formatı Şekil 12'de verilmiştir.



Şekil 12 Kan Basıncı Veri Formatı

Servisler başlangıçta tek başlarına Axis web servisleri olarak geliştirilmişler ve diğer bileşenlerle entegrasyonları ayrıca ele alınmıştır.

Kablosuz tıbbi sensorlerden alınan veri başlıca Akıllı Klinik Karar Destek Sistemi tarafından kullanılmaktadır. Yönerge Çalıştırma Bileşeni, GLIF ile temsil edilen yönerge algoritmalarını çalıştırır. Bu bileşenin diğerleriyle etkileşimi gönerge algoritması tanımlarına göre yönlendirilir. Anlamsal olarak zenginleştirilmiş web servisleri aracılıyla yapılamakta olan sensor verisinin erişimi, bu etkileşimlerin en belirgin ve önemli parçasını oluşturur.

Web servislerinin, yönerge çalıştırımı ile ilgili entegrasyon, temel olarak karşılanması gereken iki birlikte işlerlik gereksinimini doğurmaktadır:

- Sözdizimsel Birlikteişlerlik
- Anlamsal Birlikteişlerlik

Web servisleri, SOAP mesajları yardımıyla, uzaktaki işlemlere ulaşım için iki standard yol sağlamaktadır. Bu yüzden, sensor verisine ulaşım aracı olarak web servislerinin seçilmesi, sensor verilerinin sözdizimsel entegrasyonu ile ilgili bir çok gereksinimi karşılamaktadır. Ancak, web servisleri kullanılsa dahi, hala çözülmesi gerekli uygulamaya özel problemler vardor.

Bu problemlerden ilki, Yönerge Çalıştırım Bileşeninin kullandığı SOAP istemici ile ilgilidir. SOAP istemcisi, web servislerinin uyandırılması için istemci mesajları oluşturan kapsamlı bir SOAP mesaj oluşturucusudur.

Sunucu, bir web servisinin bir tek operasyon içereceği varsayımıyla tasarlanmış ve geliştirilmiştir. Ancak, sensor web servisleri diğer bileşenlerin de kullanacağı birden fazla operasyona sahip olabilmektedir. Bu yüzden, sunucunun serviste hangi operasyonu seçeceğine yönelik bir belirsizlik bulunmaktaydı. Bu problemin üstesinden gelebilmek için, sunucu bir serviste ismi verilen operasyonu çağırabilecek şekilde genişletildi. Web servislerinin isimlerdi de yönerge çalıştırım bileşenini kapsamlı tutmak amacıyla web servislerinde bazı operasyonaların isimleri değiştirildi. Buna ek olarak, web servislerin parametreleri de birleştirildi. Web servisi operasyonları hasta tanımlayıcısı ile çağrılır ve en güncel sensor bilgisi servis tarafıdan cevap olarak dönülmektedir.

Sözdizimsel birlikteişlerlik sensor verisinin doğru bir biçimde yönerge çalıştırım bileşenine iletilmesini sağlar. Ancak, eğer bu gelen veri, yönerge çalıştırım modülünün karar destek sisteminde kullanılan veri modeliyle (GLIF veri modeli) entegre olmadığı sürece, işlevsiz olacaktır. Yukarda da bahsedildiği gibi, sensor verisi daha önceden belirlenmiş XML şemalarına uygun web servisler tarafından sağlanmaktadır. Bu yüzden bu şemalar ve GLIF arasından bir dönüşüme ihtiyaç var. Bu dönüşümü sağlamak için, GLIF Hasta Veri Obje yapıtaşları XML şemaları olarak oluşturulmuştur.

Bu proje çerçevesinde, BP ve SpO2 sensorleri için sensor veri şemaları öncelikle standard modellere eşlenmiştir. IEEE 11073 standardı, Hasta Bakım Cihazı bilgilerini bir veri modeli ve onun terminolojisiyle tanımlamaktadır. Proje kapsamında geliştirilen dönüşüm aracı, kendisine has formattaki veya IEEE 11073 standardı dışındaki herhangi bir standarddaki sensor veri bilgilerinin IEEE 11073 sözdizimine çevrimini mümkün hale getirmektedir. Daha sonrasında IEEE 11073 sözdizimi XSL dönüşümleriyle HL7'e ve son olarak da GLIF formatına dönüştürülür. ECG için ise dönüşüm direk ecgML'den GLIF formatınadır.

Bu dönüşüm yeteneği yönerge çalıştırım tarafında web servisleri istemcisi tarafından alınan verinin anında GLIF sözdizimine çevrirecek şekilde geliştirilmiştir. Sensor web servislerinin anlamsal entegrasyonu bu sayede gerçekleştirilmektedir.

Sistemin İzleme Aracıyla Entegrasyonu

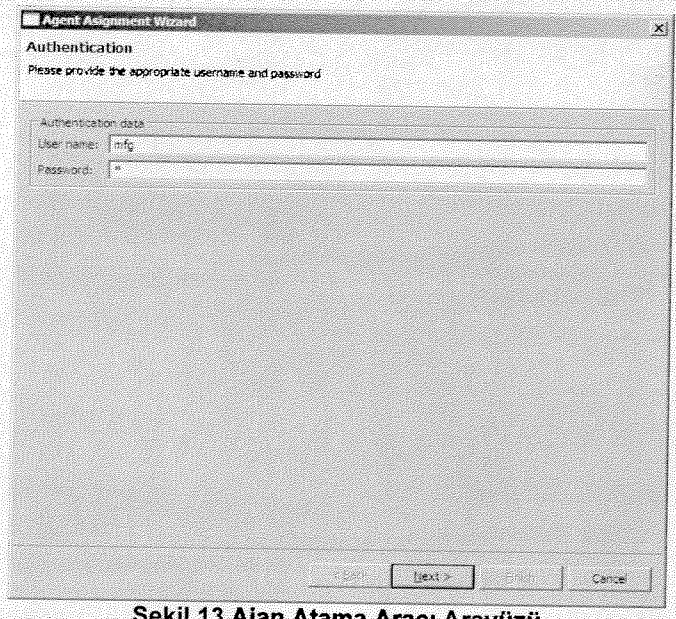
Hekimlere, sensor verisine dayalı ileri derece ilgi gerektiren hayati tehlikedeki koşulları ve durumları tanımlamasına izin verecek kullanıcı dostu ve esnek bir arayüz geliştirilmesi hedeflenmişti. Bu amaçla proje çerçevesinde aşağıdaki iki ana bileşen geliştirildi:

- Ajan Atama Aracı
- Veri Erişimi Eşleme Aracı

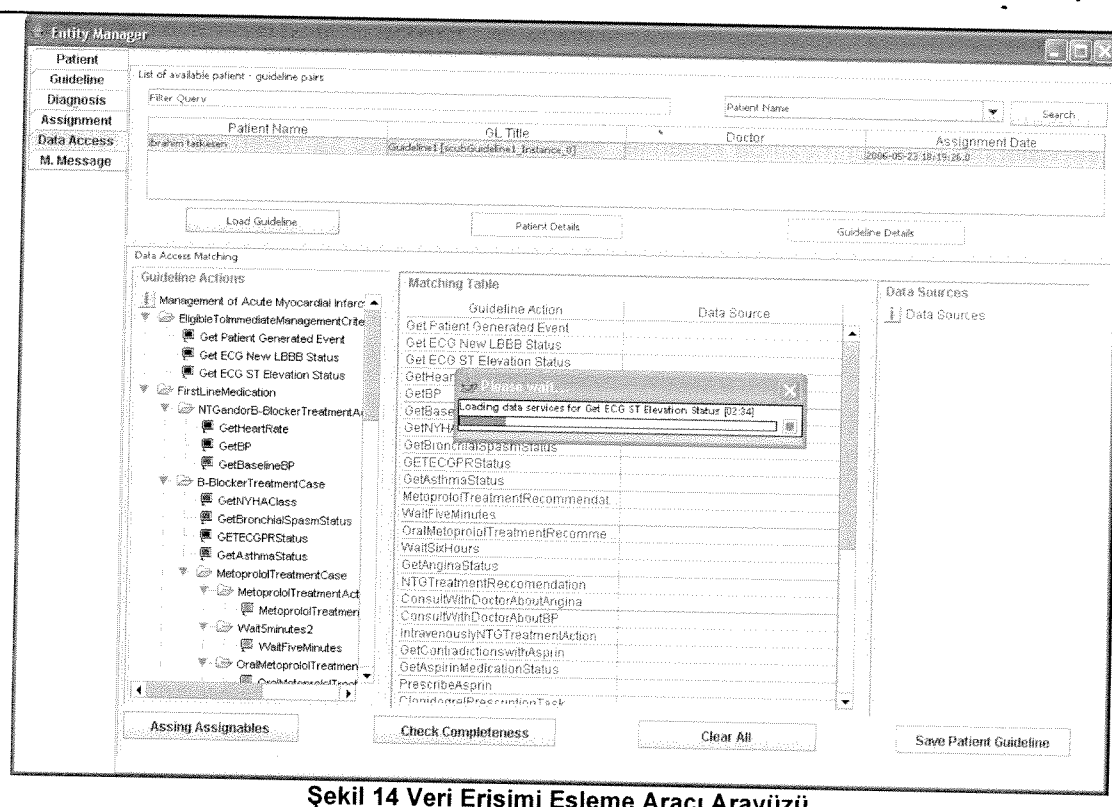
Ajan Atama Aracı yönergeleri hastalara atamada kullanılır. Bu araç, hekimlere bir hasta listesinden hasta seçmelerini ve bir yönerge ve bir sensor setini hastaya atamalarını sağlamaktadır. Buna ek olarak, doktor aynı zamanda uyarı eşik değerlerini ve kurallarını da bu

arayüz yardımıyla kolayca belirleyebilmektedir. Doktor atama işlemiyle ilgili ayarları bitirince, atama sistem veri havuzuna kaydedilir.

Veri Erişimi Eşleme Aracı yönergelerine işletimi sırasında hasta-yönerge çiftinin erişebileceği veri servislerinin tanımlanmasında kullanılır. Sensor ve klinik web servisleri ve EHR referansları bu süreçle ilgilidir. Veri Ulaşım Servisleri hasta-yönerge çiftlerine anlamsallık ve kod şemaları yardımıyla yarı otomatik olarak atanır. Aşağıdaki şekillerde bu bileşenin arayüzler verilmiştir.



Şekil 13 Ajan Atama Aracı Arayüzü



Şekil 14 Veri Erişimi Eşleme Aracı Arayüzü

Bu bölümde adı geçen araçlar ayrı ve kendi başına uygulamalar olarak geliştirildi. Ancak, tüm sistemin klinik süreci tanımlamak için daha uygun bir araç olmasını sağlamak için tüm araçların bir çatı altında toplanması gerekliydi. Bu araçların entegrasyonu için, Ajan Eşleme Aracının, Veri Erişimi Eşleme Aracına bağlanması kararlaştırıldı. Ajan Eşleme Aracı, Veri Erişimi Eşleme Aracı içersine dahil edildi ve kullanıcı arayüzünde yeni bir sekmeyle birbirine bağlandı. Bu sayede, şu anda kullanıcı genel amaçlı sadece bir arayüz başlatarak ilgili işlevlere ulaşabilmekte. Sekmeler arasında gezinerek Ajan Eşleme Aracına ve Veri Erişimi Eşleme Aracına kolaylıkla geçebilmektedir.

2. Dönem İçinde İdari Gelişmeler

Proje, proje önerisinde belirtildiği şekilde ve herhangi bir idari değişiklik olmadan yürütülmüştür.

3. Proje Çalışma Takvimine Uygun Yürümüyorsa Gerekçeleri

Proje planlandığı üzere çalışma takvimine uygun olarak tamamlanmıştır.

4. Bir Sonraki Dönemde Yapılması Planlanan Çalışmalar

-

5. Destekleyen Diğer Kuruluşlarla İlgili Sorunlar Varsa Ayrıntıları ve Çözüm Önerileri

-

6. Dönem İçinde Proje Kapsamında Yapılan Yayımlar ve Toplantılarda Sunulan Bildiriler

- Laleci G. B., Dogac A. A Semantically Enriched Clinical Guideline Model Enabling Deployment in Heterogeneous Healthcare Environments Under revision in IEEE Transactions on Information Technology in Biomedicine. (Science Citation Index Core, Impact Factor: 1.575).
- Laleci G., Dogac A., Olduz M., Tasyurt I., Yuksel M., Okcan A. SAPHIRE: A Multi-Agent System for Remote Healthcare Monitoring through Computerized Clinical Guidelines Whitestein Series in Software Agent Technologies and Autonomic Computing Special Issue on Agent Technology and e-Health, pp. 25-54, 2008.
- Nee O., Hein A., Gorath T., Hulsmann N., Laleci G., Yuksel M., Olduz M., Tasyurt I., Orhan U., Dogac A., Fruntelata A., Ghiorghe S., Ludwig R. SAPHIRE: Intelligent Healthcare Monitoring based on Semantic Interoperability Platform - Pilot Applications IEE Proceedings Communications Special Issue on Telemedicine and e-Health Communication Systems, 2008, 2, (2), pp. 192–201. (Science Citation Index Core)
- Dogac A., Laleci G., Olduz M., Tasyurt I., Yuksel M., Okcan A., SAPHIRE-Intelligent Healthcare Monitoring Based on Semantic Interoperability Platform Proceedings of Med-e-Tel 2008 Conference, April 16-18, 2008, ISSN 1818-9334

5. Bulgular ve Tartışma/Sonuç

Sistem entegre edildikten sonra sistem bir hastaneye pilot uygulama olarak kuruldu. Tüm ayar ve kurumlar hastane içerisinde yapıldı. bilgilendirici katılım belgesini imzalayıp pilot uygulamaya katılan hastalar sistem tarafından gözlemlendi, denetlendi ve sonrasında uyarı ve yönerge tavsiyeleri gerçek zamanlı olarak yine sistem tarafından üretildi. Hastalar ve hastane personeli uygulama sonlandırıldığında geri bildirim anketi doldurdular.

Hastaların EHR verileri, oluşturulan pdf raporları, uyarı ve sensor bilgileri analiz edildi ve raporlandı. Pilot uygulama için hastanede her biri 2 hasta kapasiteli 2 oda ayrıldı. Her iki oda da Bluetooth arayüzleri olan bir geçit bilgisayarı vardı. Bu bilgisayarlar kablosuz sensorlerin bilgilerinin alınmasına ve merkezi bilgisayara iletilmesini sağlamaktaydı. Sunucu bilgisayarsensor verisini depolamakta, veri analizi yapmakta ve sensor verisini web servisleri aracılıyla ulaşabilir hale getirmekteydi. Ayrıca, hasta elektronik sağlık kayıtları da sunucu bilgisayarda idi. Hastane personeli sisteme workstation bilgisyarı aracılıyla ulaştılar. Workstation bilgisayarında Akıllı Klinik Karar Destek Sistemi, Ajan Atama ve Veri Erişimi Eşleme Aracı, İzleme Aracı ve Uyarı Bileşenleri bulunmaktaydı.

Uygulama tıbbi senaryolar kullanıcının gözüyle hastane bilgi sistemini farklı durumlar için kullanılırkenki gereksinim bildirimlere dayanarak oluşturulmuştur. Bu senaryolar aşağıdaki gibidir:

- İzleme başlatımı
- Sensorlerden veri alımı
- Sistem uyarılarının alımı
- EHR ve hastana bilgi sisteminden bilgi alımı
- Yönergelerin çalıştırımı
- Hastane personelinin sistemi kullanması için eğitimi
- Hasta izleminin duraklatılması
- Periyodik sistem kontrolü

Bu periyodda, Hastane Pilot Uygulamasının tasarlanmasının yanında aynı zamanda sistemde de bazı tasfiyeler de gerçekleştirildi. Bu iyileştirmeler genel olarak tıbbi bilginin görselleştirilmesinde ve sistem arayüzlerinde gerçekleştirildi. Proje kapsamında klinik yönerge modelleri tanımlanmış ve daha önceki raporlarda akış grafiği olarak gösterilmişti. Doktorlardan gelen geribildirimlere paralel olarak bu akış grafiklerinde ve yönerge modellerinde de iyileştirilmeler gerçekleştirildi. Pilot uygulamada bu modellerin son hali Akıllı Klinik Karar Destek Sisteminde başarıyla simule edildi.

Proje kapsamında aynı zamanda sistemi kullanacak hastane personeline bu konuda yardım edildi. Çalışan bileşenler için süreçlerin tanımları sağlandı. Bu yardım süresinde e-mail, telefon ve VNC bağlantıları kullanıldı.

Sonuç olarak, Hastane Pilot Uygulaması hastaneye konuşlandırılıp, sistemin ne kadar başarılı olduğu ve hem hastaların hem de hastane personelinin sistem değerlendirmesi alındı. Projede sadece kavramı ispatlayıcı geliştirilen Akıllı Anlamsal Tabanlı Karar Destek Sisteminin kullanıcılarına büyük kolaylıklar sağlayabileceği ortaya çıktı. Henüz daha çok başlangıç seviyesinde olsa da ilerde en azından küçük parçalarının ticarileştirilmesi düşünülebilir.

6. Bir Önceki Dönemin Raporuna Ait Hakem Görüş ve Önerilerine İlişkin Açıklamalar

Etmen-Etmen etkileşimli Güvenlik Altyapısı

Projede sistem içirisindeki içerisindeki güvenlik yapısı sadece web servislerinde gizlilik ve güvenliğin gerçekleştirilmesiyle kalmayıp, aynı zamanda EHR'den bilgi çekerken de bilgi gizliliği, güvenilirliği ve bütünlüğü göz önünde bulundurulmuştur. EHR'den veri alımı (CDA dosyalarına erişim), IHE XDS standardına uygun olarak gerçekleştirilmiştir. Bunun gerçekleştirilmesi stardında Kurumlar arası kullanıcı kimlik denetimi (XUA) ve SAML Web Tek Oturum (SAML Web Single Sign-On) profilleri analiz edilmiş ve kullanılmıştır. Buna ek olarak ajanların mesajlaşması da JADE güvenlik mekanizmasına dönüştürülmüş bu sayede tüm bilgilerin güvenliği ve gizliliği sağlanmış olmaktadır.

Sensor yapısının genişleyebilirliği

Sensor verisi birlikte-işlerlik platformuna iletilmekte ve XML şemalarına sahip web servisleri aracılıyla ulaşılabilir hale getirilmektedir. XML Web Servisleriyle geliştirilen uygulamalar bulundukları yer veya implementasyonlarından bağımsız birbirleriyle haberleşebilmektedir. Projede standard yapıları tercih etmiş olsak da, web servisi mesaj formatı herhangi bir yapıda olabilir. Birlikte-işlerlik platformu bu farklı yapıda olsalar dahi veriyi yönergenin işleyebileceği bir yapıya dönüştürme yeteneğine sahiptir. Tüm bunlar göz önünde bulundurularak, sisteme yeni bir sensor eklenmesi gerektiğinde yapılması gerekenler, sensor bilgisini ulaşılabilir kılan bir web servisi yazmak ve eğer bu sensor bilgisi bir standarda uymuş değilse, yine proje çerçevesinde geliştirilmiş olan XLST dönüşüm aracıyla bu veritipini GLIF formatına eşlemektir. Eğer veri zaten standard bir formattaysa sadece web servisini yazmak yeterli olacaktır. Bu altyapı göz önünde bulundurulduğunda sistemdeki sensor bileşeninin esnek bir yapıda olduğunu söyleyebiliriz.

7. Yararlanılan Kaynaklar

1. Dogac A., Laleci G., Olduz M., Tasyurt I., Yuksel M., Okcan A., SAPHIRE-Intelligent Healthcare Monitoring Based on Semantic Interoperability Platform Proceedings of Med-e-Tel 2008 Conference, April 16-18, 2008, ISSN 1818-9334

2. Laleci G. B., Dogac A., A Semantically Enriched Clinical Guideline Model Enabling Deployment in Heterogeneous Healthcare Environments Under revision in IEEE Transactions on Information Technology in Biomedicine.

3. Laleci G., Dogac A., Olduz M., Tasyurt I., Yuksel M., Okcan A., SAPHIRE: A Multi-Agent System for Remote Healthcare Monitoring through Computerized Clinical Guidelines Whitestein Series in Software Agent Technologies and Autonomic Computing Special Issue on Agent Technology and e-Health, pp. 25-54, 2008.

4. Namli, T., Dogac A., Implementation Experiences On IHE XUA and BPPC Technical Report, December 2006.

5. Nee O., Hein A., Gorath T., Hulsmann N., Laleci G., Yuksel M., Olduz M., Tasyurt I., Orhan U., Dogac A., Fruntelata A., Ghiorghe S., Ludwig R., SAPHIRE: Intelligent Healthcare Monitoring based on Semantic Interoperability Platform - Pilot Applications IEE Proceedings Communications Special Issue on Telemedicine and e-Health Communication Systems, 2008, 2, (2), pp. 192–201.

TÜBİTAK
PROJE ÖZET BİLGİ FORMU

Proje No: 105E133
Proje Başlığı: Anlamsal Birlikte İşlerlik Platformu Tabanlı Akıllı Sağlık Takibi (Intelligent Healthcare Monitoring based on a Semantic Interoperability Platform)
Proje Yürütücüsü ve Araştırmacılar: Prof. Dr. Asuman Doğaç, Gökçe Banu Laleci Ertürkmen, Yıldıray Kabak, Mehmet Olduz, Özgür Gülderen, Alper Okcan, Tuncay Namlı, Umut Orhan, Mustafa Yüksel, İbrahim Taşyurt
Projenin Yürütüldüğü Kuruluş ve Adresi: Orta Doğu Teknik Üniversitesi
Destekleyen Kuruluş(ların) Adı ve Adresi: TÜBİTAK Elektrik Elektronik ve Enformatik Araştırma Grubu Atatürk Bulvarı No:221 06100 Kavaklıdere ANKARA Tel: (+90 312) 468 21 79
Projenin Başlangıç ve Bitiş Tarihleri: 01.09.2006 - 01.09.2008
Öz (en çok 70 kelime) Bu proje kablosuz tıbbi sensor verisi ve hastane bilgi sistemleriyle kolay entegrasyonu sağlayarak Akıllı Sağlık İzleme ve Karar Destek Sistemi geliştirmeyi amaçlamaktadır. Projede devamlı hasta gözetimi, ajan teknolojisindeki "ajan davranışı" nın akıllı bir klinik karar destek sistemiyle desteklenmesiyle sağlanmıştır. Bu karar destek sistemi bilgisayarlaştırılmış klinik yönergelere dayalı olup birlikte işlerlik problemini aşmak amacıyla anlamsal olarak zenginleştirilmiş web servisleri kullanılarak ulaşılmaktadır. Ek olarak, klinik yönergenin işleyişini gösteren kullanıcı dostu bir arayüz de geliştirilmiştir.
Anahtar Kelimeler: Akıllı hasta takibi, Karar destek sistemi, kablosuz medikal algılayıcılar, birlikte çalışablirlik platformu, klinik uygulama kılavuzları

Projeden Yapılan Yayınlar:

- Laleci G. B., Dogac A. A Semantically Enriched Clinical Guideline Model Enabling Deployment in Heterogeneous Healthcare Environments Under revision in IEEE Transactions on Information Technology in Biomedicine. (Science Citation Index Core, Impact Factor: 1.575).
- Laleci G., Dogac A., Olduz M., Tasyurt I., Yuksel M., Okcan A. SAPHIRE: A Multi-Agent System for Remote Healthcare Monitoring through Computerized Clinical Guidelines Whitestein Series in Software Agent Technologies and Autonomic Computing Special Issue on Agent Technology and e-Health, pp. 25-54, 2008.
- Nee O., Hein A., Gorath T., Hulsmann N., Laleci G., Yuksel M., Olduz M., Tasyurt I., Orhan U., Dogac A., Fruntelata A., Ghiorghe S., Ludwig R. SAPHIRE: Intelligent Healthcare Monitoring based on Semantic Interoperability Platform - Pilot Applications IEE Proceedings Communications Special Issue on Telemedicine and e-Health Communication Systems, 2008, 2, (2), pp. 192–201. (Science Citation Index Core)
- Dogac A., Laleci G., Olduz M., Tasyurt I., Yuksel M., Okcan A., SAPHIRE-Intelligent Healthcare Monitoring Based on Semantic Interoperability Platform Proceedings of Med-e-Tel 2008 Conference, April 16-18, 2008, ISSN 1818-9334

SAPHIRE: A Multi-Agent System for Remote Healthcare Monitoring through Computerized Clinical Guidelines

Gokce B. Laleci, Asuman Dogac, Mehmet Olduz, Ibrahim Tasyurt, Mustafa Yuksel and Alper Okcan

Abstract. Due to increasing percentage of graying population and patients with chronic diseases, the world is facing serious problems for serving high quality healthcare services to citizens at a reasonable costs. In this paper, we are providing a Clininical Desicion Support system for remote monitoring of patients at their homes, and at the hospital to decrease the load of medical practitioners and also healthcare costs. As the expert knowledge required to build the clinical decision support system, Clinical Guidelines are exploited. Examining the reasons of failure for adoption of clinical guidelines by healthcare institutes, we have realized that necessary measures should be taken in order to establish a semantic interoperability environment to be able to communicate with various heteregenous clinical systems. In this paper these requirements are detailed and a semantic infrastructure to enable easy deployment and execution of clinical guidelines in heteregenous healtcare enviroments is presented. Due to the nature of the problem which necessitates having many autonomous entities dealing with heteregenous distributed resources, we have built the system as a Multi Agent System. The architecture described in this paper is realized within the scope of IST-27074 SAPHIRE project.

1. Introduction

The World is facing problems to provide high quality healthcare services at a reasonable cost to the citizens due to the increasing percentage of graying population. According to a study performed by United Nations, by 2050, 22 percent of

This work is supported by the European Commission through IST-27074 SAPHIRE project and in part by the Scientific and Technical Research Council of Turkey, Project No: EEEAG 105E133.

the World's population, nearly 2 billion people, will be 60 and older. With the demographic change, the prevalence of chronic conditions such às chronic respiratory and vessel diseases increases: the percentage of elderly at 60s and older having at least one chronic disease is more than 60 [1]. The solution to decrease both the cost of healthcare services and also the load of medical practitioners requires a dramatic change in the way future healthcare services are provided. The expected necessary changes are: moving from reactive to preventive medicine, concentrating on the long term care rather than only acute care, citizen centered care rather than hospital centered care, including remote care delivery mechanisms where the citizen is taking a bigger role in his/her treatment and lifestyle management. All of these necessitate technologies for long term monitoring of the patients both in hospital and home settings.

Enabling underlying infrastructures such as wireless medical sensor devices, wearable medical systems integrating sensors on body-worn platforms like wrist-worn devices or biomedical clothes are offering pervasive solutions for continuous health status monitoring through non-invasive biomedical, biochemical and physical measurements. Remote monitoring systems typically collect these patient readings and then transmit them to a remote server for storage and later examination by healthcare professionals. Once available on the server, the readings can be used in numerous ways by home health agencies, by clinicians, by physicians, and by informal care providers. However remote healthcare monitoring systems will be exploited to their full potential when the analysis is also performed automatically through clinical decision support systems fed by expert knowledge. Clinical practice guidelines constitutes the most suitable source of information for building such clinical decision support systems.

Clinical practice guidelines are the systematically developed statements designed to assist practitioners to make decisions about appropriate medical problems. They aim to reduce inter-practice variations and cost of medical services, improve quality of care and standardize clinical procedures [2]. In order to be able to share clinical guidelines and manage their enforcement through computerized systems, a number of machine processable models of Clinical Guidelines have been proposed such as GLIF [3], ASBRU [4], ARDEN [5] and EON [6]. Based on these machine processable guideline definitions, a number of clinical decision support systems have been built such as GLEE [7], GLARE [8] and DeGel [9].

Despite the benefits of clinical guidelines, and also although we have such machine processable models and clinical decision support systems for execution them, it has been a well accepted fact that wide adoption computerized clinical practice guidelines has yet to be achieved even within a single healthcare institute. This is because of the difficulty of integration of clinical decision support systems with the already existing clinical workflow systems run by healthcare institutes: for this the clinical decision support system needs to communicate with various heterogeneous clinical applications run by the healthcare institute [10, 11]. Especially in the case of long term and remote monitoring of the patients, the clinical decision support systems need to communicate with many different information sources:

medical devices, several electronic healthcare record systems, and the decisions need to affect the processes held at disparate care providers such as homecare, emergency centers, primary and secondary care, and rehabilitation centers. Hence we definitely need robust clinical guideline execution systems that can cope with semantic and technical integration problems with disparate healthcare information systems.

In this paper, the SAPHIRE project will be introduced which provides a Multi-Agent system for the monitoring of chronic diseases both at hospital and also in home environments based on a semantic infrastructure. The system is capable of deploying and executing clinical guidelines in a care environment including many disparate care providers having heterogeneous information systems. In Section 2, the challenges and requirements of deploying and executing a clinical guideline execution infrastructure for remote monitoring of patients in a heterogeneous care environment will be detailed. In Section 3, the SAPHIRE Multi-Agent System that addresses these challenges through an enabling semantic interoperability environment will be introduced. Finally Section 4 will conclude the paper, discussing the current status and future challenges.

2. The requirements for seamless execution of Clinical Guidelines for long term healthcare monitoring

In order to guarantee successful execution of clinical decision support systems for long term monitoring of patients based on clinical practice guidelines, the integration, more importantly interoperability, with the following external interfaces should be assured:

- *Accessing vital signs of the patient:* In order to be able to monitor the patient's current condition, the clinical decision support systems need to access the vital signs of the patient measured by wireless medical sensors and body-worn platforms. Currently there are many biomedical sensors devices available, and active research is going on for body worn platforms initial products of which will be soon in the market. The clinical decision support systems should be able to communicate with heterogeneous medical devices supplied by various different vendors. We have two interoperability problems to access the vital signs measured by these devices: the first one is the technical interoperability problem to access the vital signs physically: there may be different protocols implemented by different medical device vendors. In SAPHIRE architecture we are addressing the technical level interoperability problem by exposing the sensor data through Web Services. The sensor data is gathered through Bluetooth from wireless sensor devices to a gateway computer where they are exposed as Web services. By exposing the sensor data as Web services, a platform independent way of accessing the vital signs measured by sensor devices is achieved. The second interoperability challenge that should be addressed is content level interoperability problem: After accessing the sensor

data through Web services, the content received should be processable and interpretable by the receiving application, the clinical decision support system in our case. However, the data coming from the wireless medical sensors are either in proprietary format (for example, for electrocardiogram data, Philips XML ECG Data Format) or when it conforms to a standard, this still does not solve the interoperability problem since there are very many standards (again for electrocardiogram data, the available standards include: SCP-ECG [12], US Food and Drug Administration FDA/HL7 Annotated ECG [13], I-Med [14] and ecgML[15]). There is also a very important interoperability initiative for the interoperability of the data coming from medical devices: the IEEE 11073 Standards Family[16] which aims to enable functional and semantic ad-hoc interoperability. For this purpose, the IEEE 11073 proposes an Object-oriented modeling of function and application area, the "Domain Information Model" (DIM). Through the DIM it is possible to define and represent devices, functionalities, measurement data, calibrations, alert information and so on. On top of the DIM, it provides standardized codes for naming all information elements in the DIM such as medical devices and device systems, units of measurements through the "Nomenclature" and "Data Dictionary". IEEE 11073 assumes that all device vendors to adopt this DIM to represent sensor data to achieve interoperability. However for the time being the vendors still using proprietary formats or different standards can not be ignored. For this purpose, in our architecture we provide a translation wizard, through which the translation of proprietary XML schemas of sensor data to the IEEE 11073 format can be easily defined graphically enabling the user to define Javascripts taking the pieces of input XSD schema. This translation definition is used to transform the data instances automatically to one another. In this way it is possible to have all the sensor data in IEEE 11073 format in SAPHIRE Gateway computer to be exposed as Web services.

- *Accessing Electronic Healthcare Records of the Patient:* The gathered vital signs of the patient can only be assessed correctly when consolidated with the Electronic Healthcare Records (EHRs) of the patient. The evaluation of the vital signs should be "personalized" for each patient, based on their past illnesses, active problems, family histories, allergies and adverse reactions. In addition to this, the clinical decision support system executing clinical guidelines needs to know the previous medical history of the patient to follow the correct branch for the medication or operation recommendations to be presented to the medical staff: for example the first line medication to be applied to a patient who may be suffering from myocardial infarction varies based on his/her medical history: it is not appropriate to recommend a B-blocker if the patient previously suffered from bronchial spasm or asthma. To be able assess these, the clinical guideline execution environment needs to access the Electronic Healthcare Records of the Patient where ever they are. However there is a challenge to be addressed here: Patient medical records that the

clinical decision support system need to process are usually physically dispersed in disparate medical institutions which usually do not interoperate with each other. First of all the Clinical Decision support system needs to discover these records, and then needs to seamlessly access the records to process them. One of the prominent initiatives for sharing EHRs is the Integrating Healthcare Enterprise (IHE). IHE, through the Cross Enterprise Document Sharing Integration Profile (XDS) [17], enables a number of healthcare delivery organizations to share clinical records. This profile has received considerable attention and appeared in the National eHealth System blueprints of Canada, USA, Italy, Norway and France.

In the IHE XDS Profile, healthcare enterprises that agree to work together for clinical document sharing are called a "Clinical Affinity Domain". Such institutes agree on a common set of policies such as how the patients are identified, the access is controlled, and the common set of coding terms to represent the metadata of the documents.

In each affinity domain there are a number of "Document Repositories"; the healthcare institutes store the medical documents of the patients to these repositories in a transparent, secure, reliable and persistent way. There is a "Document Registry" which is responsible for storing information about those documents so that the documents of interest for the care of a patient may be easily found, selected and retrieved irrespective of the repository where they are actually stored. The document repositories register the documents along with a set of metadata to the Document Registry. Whenever a "DocumentConsumer" wishes to locate a specific document of a patient, the "Query Document" transaction is issued along with the specified query criteria, and as a response a list of document entries that contain metadata found to meet the specified criteria is returned including the locations and identifier of each corresponding document in one or more Document Repositories. Using these document identifiers and the Document Repository URI's, the "Retrieve Document" transaction is issued to get the document content.

The SAPHIRE multi-agent system that facilitates the execution of the clinical decision support system uses this IHE Profile to locate and access the records of the patients which will be detailed in Section 3.

The Electronic Healthcare Records accessed should be machine processable so that the content can be interpreted to retrieve the necessary piece of the EHR required by the clinical guideline definition. For this purpose in SAPHIRE architecture, the EHR documents are represented as the HL7 Clinical Document Architecture (CDA) [18] documents. The HL7 CDA is a document markup standard that specifies the structure and semantics of "clinical documents" for the purpose of exchange. CDA documents are encoded in Extensible Markup Language (XML) and they derive their machine processable meaning from the HL7 Reference Information Model (RIM) [19] and use the HL7 Version 3 Data Types. In the SAPHIRE architecture, both the "Sections" and "Document Entries" are annotated with coded terms of

medical terminologies and ontologies such as LOINC [20], SNOMED [21] and ICD-10 [22] so that the clinical guideline execution environment can process the information contained in the EHR of the patient. However it should be noted that, in the clinical guideline definition the clinical information requested may have been represented through a code in a different medical terminology from the one that has been used in the CDA document, in this case, the "Ontology Agent" of SAPHIRE multi agent system is contacted to handle the mediation between different coding standards.

- *Accessing the Clinical Workflow systems executed at Healthcare Institutes:* While the clinical decision support system is executing the Clinical Guideline Definition, it is needed to interact with several modules of the clinical workflow executed at the healthcare institutions. For example, if the clinical decision support system recommends to prescribe a B-Blocker to a patient, this medication recommendation should be reflected to the underlying clinical workflow, otherwise the clinical decision support system and the clinical workflow run in parallel without any interaction with each other, the activities are not synchronized with each other. This hampers the use of clinical decision support systems to their full potential. For this kind of interactions like medication, procedure or lab orders, there needs to be an interface provided by the underlying hospital information system executing the clinical workflow. However most of the hospital information systems are proprietary, which makes the deployment of clinical decision support systems to healthcare institutes difficult. Integration with each of such hospital information system is costly; there needs to be a mechanism that enables interoperability for accessing these proprietary systems to avoid manual integration efforts. In SAPHIRE, we are proposing to solve this problem by exposing the functionalities provided by Healthcare Institutions as Web Services, and publishing these Web services to Service registries by annotating them with ontologies reflecting their functionality. This will allow us to automatically deploying the clinical decision support systems executing clinical guidelines automatically. Web services have already started to be adopted by the Healthcare Industry as a solution to technical interoperability problem. The Dutch national infrastructure for healthcare messaging is implemented by wrapping HL7v3 messages as Web services [23].

3. The SAPHIRE Multi Agent System

The SAPHIRE Clinical Decision Support System that is responsible for deploying and executing Clinical Guidelines is a multi-agent system composed of a number of collaborating agents. An overview of the subcomponents and their interaction is depicted in Figure 1. The system is implemented as a multiagent system, since as a result of conceptual design phase we have realized that in order to deploy and execute the clinical guidelines in a heterogeneous distributed environment, there

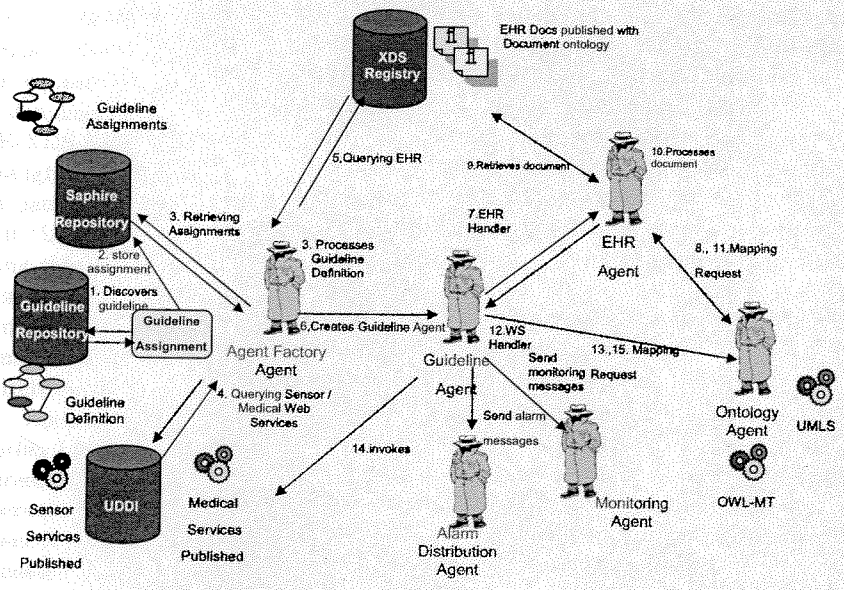


FIGURE 1. The SAPHIRE Multi Agent System

should be a number of autonomous components that should be communicating with each other in a reactive manner, and some of these components should be instantiated and eliminated dynamically based on the demand.

The roles of each SAPHIRE Agent can be introduced as follows:

- *Agent Factory Agent*: The Agent Factory Agent is mainly responsible for specializing the Guideline definition to a patient, and creating the Guideline Agent which will execute the clinical guideline. It discovers the real implementations of the medical services exposing hospital information system functionalities and sensor services and the document identifiers of the EHR documents of the patients, so that the guideline definition becomes ready to be executed.
- *EHR Agent*: In the SAPHIRE architecture the EHR agent functions as the gateway to access and extract clinical data from the Electronic Healthcare records of the patient. EHR Agent is modelled as a separate agent, to abstract the access to EHR from other agents. Currently in the SAPHIRE architecture the main mechanism for sharing EHR documents is IHE XDS Registry/Repository architecture. The EHR agent is capable of communicating with IHE XDS Registry/Repository to retrieve the EHR documents.
- *Ontology Agent*: The SAPHIRE architecture is capable of reconciliation of semantic interoperability problems while accessing the resources of healthcare

institutes. In the SAPHIRE in the guideline definition, patient data references are modelled in a reference information model based on HL7 RIM. It is possible that the medical Web services, the sensor data, and the EHR documents use different reference information models, and clinical terminologies. Through Ontology Agent this semantic interoperability problem is solved.

- *Guideline Agent*: The guideline agent is the main entity which executes the Clinical practice guidelines. The Guideline agent processes the guideline definition specialized to a patient and executes the activities specified in the guideline definition. It can be thought as the enactment engine for the clinical guideline. The guideline agent exploits several modular handlers to achieve this responsibility.
- *Monitoring Agent*: While the guideline is executed, the current status of the guideline execution is sent to a specific agent which we call Monitoring Agent. Monitoring Agent provides an interface to the Clinical Practitioners to visualize the execution of the guideline.
- *Alarm Distribution Agent*: While the guideline is executed, several alarms, notifications, reminders may need to be issued to medical practitioners, and when necessary to the patient relatives. In such cases the alarm message and the role to whom the message should be delivered is informed to an agent, the Alarm Distribution Agent, which is specifically designated to distribute these messages to the necessary recipients in the most efficient and reliable way.

For implementing the SAPHIRE Multi Agent system we have utilized the JADE [24] agent development platform. In the following sections the functionalities of the SAPHIRE Agents will be detailed.

3.1. EHR Agent

As presented in section 2, accessing the Electronic Healthcare Records of the patient is an indispensable requirement for automatic remote monitoring of the patient. However the EHR's of a patient may be stored separately in each healthcare institute s/he has been previously hospitalized. In SAPHIRE Architecture, the healthcare institutes that cooperate for the care of a patient are grouped as Clinical Affinity domains. These clinical affinity domains may have agreed on different platforms for sharing the EHRs of the patient that are not interoperable with each other. This is in fact a real life situation: in U.K as the national health infrastructure, a central architecture called SPINE [25] will be used for sharing medical summaries of patients, while in Canada, an IHE-XDS based infrastructure is being built for the same purpose [26]. To abstract the access to the EHR from the Clinical Guideline Execution Environment, we have created a dedicated agent, the EHR agent for each such affinity domain. EHR agent can be thought as a gateway for locating and accessing EHRs of the patients. Each EHR agent is specialized in the platform agreed in that affinity domain for sharing EHRs. When a request for discovering and requesting an EHR document is received by an EHR agent, the EHR agent both tries to locate the EHR document within its

affinity domain, through the methodology agreed by the clinical affinity domain such as IHE-XDS, and also forwards the request to the EHR agents of the other clinical affinity domains. In this way, the EHR documents will be available to the requesting entity, although heterogeneous systems are used by different affinity domains. In our architecture, we have implemented EHR agents accessing the IHE-XDS EHR Registry/Repositories: When a specific EHR of a specific patient is saught, an EHR Discovery message is sent to the EHR Agent. In this message, the patient identifier is presented and the document type metadata is specified with "LOINC Document Type Codes" such as "11450-4" for "Active Problems". Using this metadata, and the patient identifier, a "QueryDocument" transaction is issued to the XDS Registry, and as a response a set of Document Identifiers pointing to document stored in EHR Repositories is presented. These document identifiers are used to access the document content from the Repositories by issuing a "RetrieveDocument" transaction.

Apart from locating and retrieving EHR documents, EHR agents also serve another important feature: retrieving a specific piece of information from the EHR content. The EHR content standard agreed by each clinical affinity domain may be different, however the EHR agent of that domain, is capable of processing the document format agreed and extract the requested piece of information in the format requested by the Clinical guideline execution environment. As presented in section 2, in our architecture, we are using HL7 CDA documents as EHR documents, and in our implementation, we have implemented an EHR agent that is capable of processing the CDA document, locate the requested piece of information among the CDA Entries, and present it to the requesting entity.

In the EHR access request sent to the EHR Agent, the semantics of the piece of information requested is also specified with coded terms. For example, the Clinical Guideline Execution Environment is in need of discovering whether the patient has previously experienced "asthma". In the request sent to the EHR agent, besides the document type code for "Past illnesses", the coded term representing "asthma" is also specified for example as "C0004096" in UMLS medical terminology. In the CDA document all the entries are also annotated with coded terms, however another code from a different terminology may have been used for identifying the same entry in the CDA document which could be the "J45" term from ICD-10 terminology. To solve this interoperability problem, the EHR agent consults to the Ontology agent, and receives an answer to its translation request. In this way although different medical terminologies may have been used, the requested part of the EHR can be extracted from the whole EHR document.

3.2. Ontology Agent

The Ontology Agent in SAPHIRE Architecture is responsible for handling the semantic mediation of the clinical content used in SAPHIRE Architecture. It is used for the following purposes as presented in Figure 2:

- *Mapping the parameters of Medical Web Services*: In the SAPHIRE Architecture, the guideline execution environment uses a reference information model

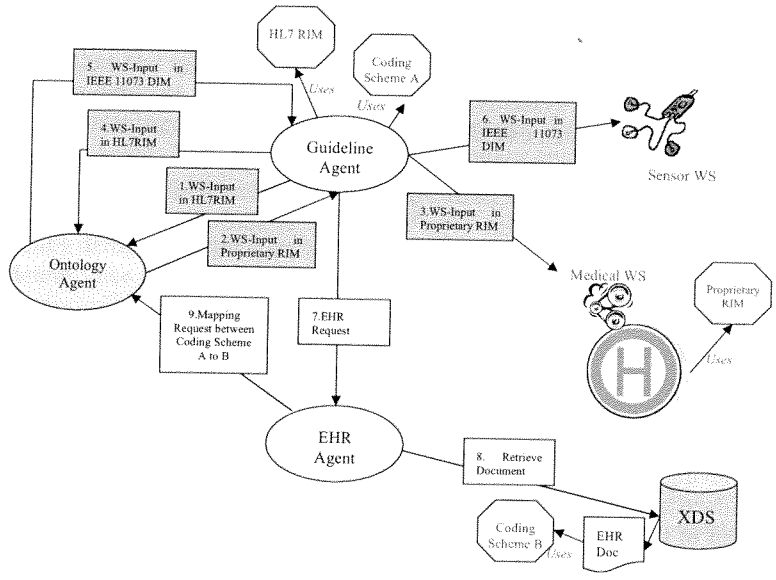


FIGURE 2. The SAPHIRE Ontology Agent

based on HL7 RIM subset to represent the clinical information. However, it is a fact that several other standards or even propriety formats may be used by the healthcare institutes to represent clinical information. The guideline execution environment needs to communicate with the hospital information systems to reflect the results of guideline execution. For example, the guideline execution can result with a proposal of prescription of a medication to the patient; in this case this information may need to be stored to the hospital information system to affect the clinical workflow. In SAPHIRE architecture, these kinds of interactions are handled through the Web services exposed by the healthcare institutes. However it is natural that the parameters of these Web services are conforming to the messaging and content standards used within the hospital, not to the one used in the guideline execution environment. Whenever the Guideline Agent needs to invoke a Medical Web Service, it consults with the Ontology Agent and the input parameters are automatically mediated to the messaging and content standards used by the hospital. The same mechanism is used for mapping the output parameters.

- *Mapping the parameters of Sensor Web Services*: In the SAPHIRE Architecture, the guideline execution environment represents the sensor data to be used in guideline execution in the same reference information model based on HL7 RIM. Currently in our architecture the sensor data will be exposed

An example translation request	*An example response to a translation request*
(request :sender (agent-identifier :name client-agent@foo.com :addresses (sequence iiop://foo.com/acc)) :receiver (set (agent-identifier :name ontology-agent@foo.com :addresses (sequence iiop://foo.com/acc))) :protocol FIPA-Request :language FIPA-SL2 :ontology FIPA-Ontol-Service-Ontology :content (action (agent-identifier :name ontology-agent@foo.co :addresses (sequence iiop://foo.com/acc)) (translate (**C0262926**)) (translation-description :from UMLSDocTypeOntology :to LOINCDocTypeOntology))) :reply-with translation-query-1123234)	(inform :sender (agent-identifier :name ontology-agent@foo.com :addresses (sequence iiop://foo.com/acc)) :receiver (set (agent-identifier :name client-agent@foo.com :addresses (sequence iiop://foo.com/acc))) :language FIPA-SL2 :ontology (set FIPA-Ontol-Service-Ontology) :content (= (iota ?i (result (action (agent-identifier :name ontology-agent@foo.com :addresses (sequence iiop://foo.com/acc)) (translation-description :from UMLSDocTypeOntology :to LOINCDocTypeOntology))) ?i)) (**11348-0**)) :in-reply-to translation-query-1123234)

FIGURE 3. An example translation request and response

as Web services which represent the data in IEEE 11073 DIM. Whenever a data is received form a Sensor Web Service, the Guideline Agent consults with the Ontology Agent to mediate the sensor data to the reference information model used in the guideline execution environment.

- *Mapping the content of the Electronic Healthcare Records of the Patient*: In the SAPHIRE architecture the Electronic Healthcare Records of the patients are represented as HL7 CDA documents. In HL7 CDA, the document sections and entities can be coded with coded terms from different coding schemes. In SAPHIRE, in the guideline definition model the EHR data can also be annotated with concepts from ontologies or coding schemes. Whenever different coding scheme standards are used, the Ontology Agent is consulted for mediation. Since the Guideline Agent cooperates with the EHR Agent whenever an EHR content is necessary, the mediation request to Ontology Agent is sent by the EHR Agent.

The Ontology Agent is compliant with the FIPA Ontology Service Specifications [27]. According to FIPA Specification an Ontology Agent is an agent that provides access to one or more ontology servers and which provide ontology services to an agent community. The Ontology Agent (OA) is responsible for the one or some of these services:

- maintain (for example, register with the DF, upload, download, and modify) a set of public ontologies,
- translate expressions between different ontologies and/or different content languages,
- respond to query for relationships between terms or between ontologies,

The FIPA Specification deals with a standard way to serve the ontology services; it does not mandate any mechanism on how to map the ontologies to one another. As well as all the other agents, the OA registers its service with the Directory Facilitator (DF) and it also registers the list of maintained ontologies and their translation capabilities in order to allow agents to query the DF for the specific OA that manages a specific ontology. Being compliant with the FIPA Ontology Service Specification necessitates the Ontology Agent to be able to accept and respond to the ontology service requests in FIPA-Ontol-Service-Ontology ontology. An example translation request and response is presented in Figure 3.

As presented the FIPA Ontology Service Specification does not deal with how the mapping is facilitated. In the SAPHIRE Architecture, the mapping is facilitated through three different mediation mechanisms (Figure 2):

- *Mapping the parameters of Medical Web Services*: In one of our previous projects, Artemis [28], we have developed an OWL Ontology Mapping Tool, the OWLmt [29], to mediate the input and output parameters of medical Web services between different standards. The SAPHIRE Ontology Agent handles such mapping requests through the OWLmt tool. The OWLmt tool provides a graphical interface to define the mapping patterns between OWL ontologies in different structures but with an overlapping content. This mapping definition is used to automatically translate ontology instances to one another. In SAPHIRE, the schemas of Web service messages, and the schema of the Reference Information Model used by the clinical guideline execution environment are lifted to metamodel level and and represented as OWL ontologies. Then through the OWLmt GUI, the mapping relationships between them is defined graphically once, which will be used by the OWLmt Mapping engine to mediate the Web service parameters to the reference information model understood by the clinical guideline execution environment. For the details of the OWLmt tool, please refer to [29], where detailed examples of mapping definitions from medical domain are presented.
- *Mapping the terminologies used in Clinical Document Content*: The SAPHIRE Ontology Agent handles such requests through a Web service exposing the functionalities of the UMLS Knowledge Source Server [30] . The UMLS Metathesaurus contains information about biomedical concepts and terms

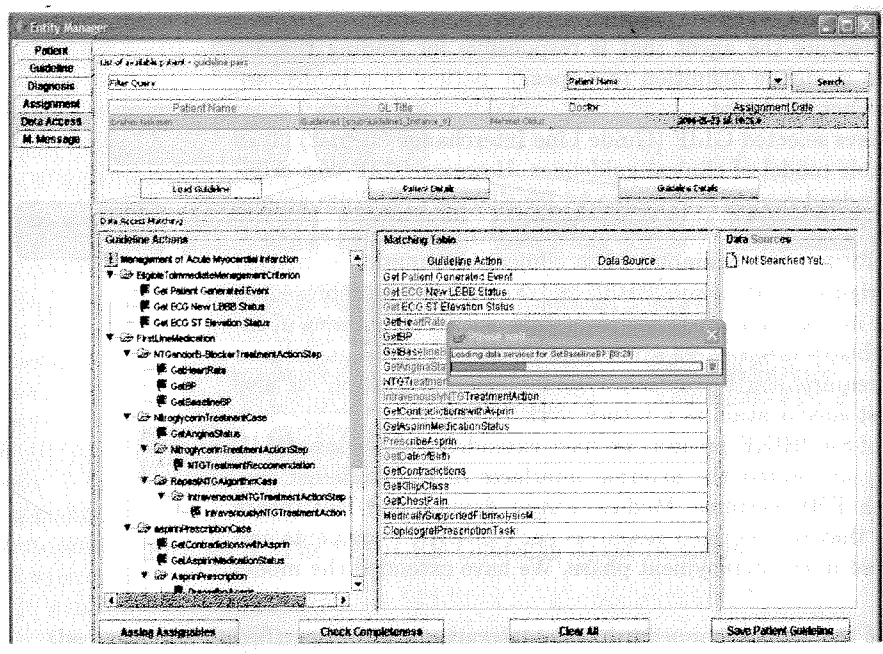


FIGURE 4. The SAPHIRE Agent Factory GUI

from many controlled vocabularies and classifications used in patient records, administrative health data, bibliographic and full-text databases, and expert system. These are referred to as the "source vocabularies" of the Metathesaurus. The Metathesaurus reflects and preserves the meanings, concept names, and relationships from its source vocabularies. The UMLS Knowledge Sources are also downloadable as databases in UMLS Site. In SAPHIRE architecture, we have implemented a Web service that queries the local UMLS database, for finding the synonyms of clinical terms. synonym terms in ICD10, LOINC and SNOMED CT if there are any.

- *Mapping the parameters of Sensor Web Services*: As presented in Section 2 the Sensor data is exposed as Web services in IEEE 11073 DIM. However this information in DIM, should be translated to HL7 RIM which is used by the clinical guideline execution environment. The IEEE 11073 Standards family names this level as "Observation Reporting Interface", and provides guidelines to map the IEEE 11073 DIM to the HL7 observation reporting messages, segments, and fields. The SAPHIRE Ontology agent implements these guidelines to handle this mediation.

3.3. Agent Factory Agent

In the SAPHIRE Architecture the agent that is responsible for leading the deploying a generic clinical guideline definition to a specific patient in a healthcare institution is the Agent Factory Agent.

In SAPHIRE, we have selected GLIF (Guide Line Interchange Format) [3] as the computer interpretable model of clinical guidelines. However GLIF was originally developed as a standard representation model for sharing guidelines among different healthcare institutes, rather than automatically deploying clinical guidelines to a healthcare institute. For example, when clinical information is needed to be retrieved, in the original GLIF, only "EHR" or "Doctor" can be represented as the source of clinical information. It is apparent that with this amount of information it is not possible to use it as an executable model of clinical guidelines. This necessity as the "requirement for an implementable representation" is also specified in GLIF's latest specification as a future work.

Within the scope of SAPHIRE project, we have extended the original GLIF model, and semantically annotated the external interfaces of the guideline execution environment with EHR systems, Medical sensor devices and Healthcare Information Systems so that the required resources such as EHR documents can be dynamically discovered in the deployment phase. We have extended the model so that:

- the functionality of the medical procedures to be interacted can be specified through ontologies.
- both the type of the EHR document sought, and also the type of the piece of information looked for in the EHR document can be specified through ontologies or medical terminologies.
- the kind of vital signs can be specified through a coded term in reference to a terminology identifying medical measurements such as IEEE 11073 Nomenclature.

The details of this extension can be found in [31, 32].

The Agent Factory Agent processes the clinical guideline definitions represented in our extended model, and based on the semantic annotations of the external resources, discovers the instances of the specified resources that are relevant for our specific patient. This process can be summarized as follows:

- In SAPHIRE architecture, the medical Web services exposing functionalities of healthcare information systems, and also the sensor Web services exposing the sensor data retrieved from wireless medical sensor devices are published to a UDDI registry by annotating them with their functionality semantics. Whenever the Agent Factory encounters a reference to a medical procedure, it locates the medical procedures from UDDI service registries by their functionality which has been specified in the extended GLIF model.
- Whenever the Agent Factory encounters a reference to a clinical data of patient to be retrieved from an EHR document, it sends a message to the EHR agent presenting the Document type, and Entry type semantics presented in

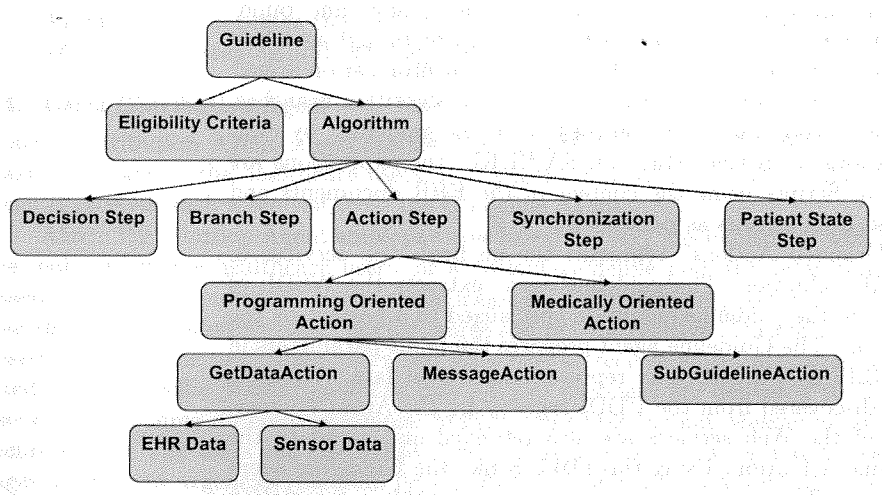


FIGURE 5. The SAPHIRE Guideline Agent Handlers

the extended GLIF model. As a response a set of document identifiers are received pointing to relevant EHR documents.

In addition to that, in the extended GLIF model, we have also reserved slots for storing the pointers to the discovered resources, for example, document identifiers in EHR repositories, the WSDL and OWL-S files of Web services. As a result of the deployment phase briefly presented, the agent factory specializes the generic guideline definition to a patient by filling in these slots.

Whenever the clinical guideline is wished to be executed for remote monitoring of a specific patient, the Agent Factory Agent instantiates a dedicated Guideline Agent for a specific guideline patient pair. In addition to this, the Agent Factory Agent informs the Monitoring Agent, about this instantiation, so that the execution of the remote monitoring process can be traced by clinical practitioners.

3.4. Guideline Agent

Guideline agent is the leading agent that coordinates the execution of the clinical guideline definition for remote monitoring of the patients. SAPHIRE Guideline agent is capable of processing any guideline definition represented in the extended GLIF model, and execute the guideline in cooperation with the other entities of SAPHIRE Multi Agent System. As presented in Figure 5, the guideline definition is composed of a number of building blocks, for each building block we have implemented modular handlers. The SAPHIRE Guideline Agent behavior is implemented to process the extended guideline definition and instantiate these modular handlers as follows:

- The main body of a clinical guideline is represented in the "Algorithm" building block. "Patient State steps" are not executable, can be though as labels for current situation of the patient. The "Branch and Synchronization Steps" coordinate the execution of serial or parallel execution of algorithm branches. The "Decision Steps" coordinate the control flow of the guideline, by evaluating the expressions on patient state. In SAPHIRE, the expressions are represented as Java Scripts using the content of the EHR documents and vital signs received from sensors as parameters.
- The "Medically Oriented Actions" represent the medical Web services in the extended GLIF definition. The guideline agent extracts the WSDL of the Web service from the guideline definition specialized to a patient by the Agent Factory Agent. The Guideline agent prepares the input parameters in HL7 RIM, since GLIF uses this RIM for representing clinical data. While the Web services are discovered from the UDDI registry by the Agent Factory, the OWL-S files of the Web services are also retrieved and saved to the specialized guideline definition. Using this OWL-S file, the Guideline agent checks the semantics of the input/out parameters, and sends a translation request to the Ontology Agent to translate the input messages from the HL7 RIM to the message schema specified in the OWL-S file. The same procedure is repeated when the output is received from the Web service.
- The "Get Data Actions" can be used to represent either references to EHR document or to vital signs of the patient to be retrieved from wireless medical sensor devices through Sensor Web services. The Sensor Web services are also invoked as the Medical Web service, by contacting with the Ontology Agent to mediate the input and output parameters.

  Whenever a reference to a clinical information presented in an EHR document is encountered by the Guideline Agent in the guideline definition, the Guideline Agent sends a request to the EHR agent, with the document identifiers previously filled by the Agent Factory Agent, and also with the semantic annotation of the clinical data to be extracted from the EHR document. As presented, the EHR agent parses the document, consults to Ontology agent when necessary to reconciliate the coded terms one another, and as a response sends the requested content in HL7 RIM to the Guideline Agent. The Guideline Agent stores all of these clinical data, sensor data to a global variable pool, so that other handlers such as "Decision Step Handler" can make use of them when necessary.
- The "Message Actions" are used to generate alarm messages within the clinical guideline execution. When the Guideline Agent encounters a "Message Action" during clinical guideline execution; it immediately constructs an alarm message by combining information coming through guideline definition and agent properties. Alarm message, healthcare role id to whom the message is to be delivered and alarm urgency parameters are retrieved from guideline definition whereas patient and guideline ids are retrieved from agent properties. The constructed alarm messages are transmitted to "Alarm Distribution

Agent", which actuates the delivery. The transmission is performed through JADE [24] messaging and ontology facilities.

3.5. Alarm Distribution Agent

Alarm Distribution Agent is responsible from accurate and punctual delivery of alarm messages to the healthcare users. It triggers the distribution of the alarms when it receives such a request from the Guideline Agent.

Alarm Distribution Agent employs a role based delivery mechanism, in which the real responsible healthcare users for a patient-guideline pair are determined based to the role id indicated by the alarm message. There are four pre-determined role ids which are administrator, doctor, nurse and patient relative. Through a web based interface, the healthcare users can subscribe to receive alarm messages related with a specific patient guideline pair. Alarm messages are delivered to the users through three different mediums: SMS, GoogleTalk Instant Messaging and secure e-mail. The users can customize their preferences for receiving alarm messages in different urgencies (medium type, number of deliveries, acknowledgement requirement, routing option etc.) through a web based user interface. User preferences are stored as JESS [33] rules. These rules are executed in delivery time and the delivery terms are determined.

Acknowledgement facility is a confirmation mechanism in order to ensure reliable delivery of the alarm messages. With this option, users are required to confirm that they have received the alarm messages. For e-mail and Instant Messaging, the acknowledgment method is simply replying to the message; SMS acknowledgment is realized through delivery confirmation message. In case that the message is not acknowledged, it is re-sent to the user for a number of times determined based on user preferences; if the message is still unacknowledged; it is routed to another healthcare user which is specified by the healthcare user.

3.6. Monitoring Agent

Monitoring Agent presents a graphical user interface to the healthcare users for clinical guidelines. Through the Monitoring Agent Interface, healthcare users can start/stop and monitor the execution of clinical guidelines by interacting with the guideline agent. In addition to these, guideline agent can consult to the healthcare professionals' decisions through this component.

Guideline execution is monitored on a user friendly interface which is composed of three parts. The main part of the interface depicts the flowchart of the clinical guideline model, whereas the others are for the message sequence and legend of the flowchart. Guideline execution can be traced on the flowchart model. The status of the guideline steps (committed/ongoing/ not visited) are identified with different colors. User can click on the steps to get detailed information about the step. In the detailed information screen, user can view the tasks, retrieved patient data (sensor, EHR etc.) and the invoked medical services within these tasks. In case that, the medical experts decision is needed, Monitoring Agent displays a pop-up window for consulting. In this way, input is provided for Guideline Agent.

The communication between Monitoring Agent and Guideline Agent is realized via JADE[24] messages. The messages are implemented in JADE ontologies in order to structure a well defined message format for monitoring and consulting. The communication between agents is based on a publish-subscribe mechanism in which multiple monitoring agents can be subscribed to one single Guideline Agent.

Apart from these, an important outcome of the Monitoring Agent is the visual model that it provides for clinical guidelines. This visual flow-chart model can be utilized as an educative medium in training healthcare professionals.

4. Conclusion

The architecture described in this paper is realized within the scope of IST-27074 SAPHIRE project. The prototype implementation is achieved using JADE Agent Platform.

The SAPHIRE has two pilot applications: in the hospital pilot we address the bedside monitoring of subacute phase of the patients suffering from myocardial infarction; in the homecare scenario we address the homecare monitoring of the rehabilitation of the cardiovascular patients undergone a revascularization therapy. A more detailed discussion of SAPHIRE pilot applications can be found in [34]. Through these pilot applications, the system aims to increase adherence to the guidelines, hence provide standardization to care processes, to reduce costs of care with optimal benefit for the patient and doctor, to reduce human error in hospital events/complications and finally to provide a feedback system for medical staff in training.

References

[1] "Demographic change & ageing: The possible role of ICT, Lutz Kubitscke," http://ec.europa.eu/information_society/events/phs_2007/docs/slides/phs2007-kubitschke-s2b.pdf.

[2] M. J. Field and K. N. Lohr, *Guidelines for Clinical Practice: From development to use.* Washington DC: Institute of Medicine, National Academy Press, 1992.

[3] "Guideline Interchange Format (GLIF) 3," InterMed Collaboratory, Tech. Rep., 2004.

[4] A. Seyfang, S. Miksch, and M. Marcos, " Combining Diagnosis and Treatment using Asbru," *International Journal of Medical Informatics*, vol. 68 (1-3), pp. 49–57, 2002.

[5] M. Peleg, O. Ogunyemi, and S. Tu, "Using features of Arden Syntax with object-oriented medical data models for guideline modeling," in *Proceedings of AMIA Symposium*, 2001, pp. 523–527.

[6] S. Tu and M. Musen, "Modeling Data and Knowledge in the EON Guideline Architecture," in *Proceedings of MedInfo 2001*, London, UK, 2001, pp. 280–284.

[7] D. Wang, M. Peleg, S. Tu, A. Boxwala, O. Ogunyemi, Q. Zeng, A. Greenes, V. Patel, and E. Shortliffe, "Design and Implementation of GLIF3 guideline execution engine," *Journal of Biomedical Informatics,*, vol. 37, pp. 305–318, 2004.

[8] P. Terenziani, G. Molino, and M. Torchio, "A modular approach for representing and executing clinical guidelines," *Artificial Intelligence in Medicine,*, vol. 23(3), pp. 249–276, 2001.

[9] Y. Shahar, O. Young, E. Shalom, M. Galperin, A. Mayaffit, R. Moskovitch, and A. Hessing, " A Framework for a Distributed, Hybrid, Multiple-Ontology Clinical-Guideline Library and Automated Guideline-Support Tools," *Journal of Biomedical Informatics,*, vol. 37, no. 5, pp. 325–344, 2004.

[10] M. Fieschi *et al.*, "Medical decision support systems: Old dilemmas and new paradigms?" *Methods Inf Med.,*, vol. 42(3), pp. 190–198, 2003.

[11] M. Entwiste and R. N. Shiffman, "Turning Guidelines into Practice: Making It Happen With Standards - Part 1," *Healthcare and Informatics Review Online, March 2005*, 2005.

[12] "Open ECG Portal," http://www.openecg.net/.

[13] "HL7 XML Standards and FDA," http://www.fda.gov/cder/regulatory/ersr/2003_06_17_XML/index.htm.

[14] "IMED - International Medical Equipment Developing Co. Ltd." http://www.imed.hu/index2.php.

[15] "ecgML," http://ijsr32.infj.ulst.ac.uk/~e10110731/ecgML/.

[16] "ISO/IEEE 11073-10101:2004:Health informatics, Point-of-care medical device communication, Part 10101: Nomenclature," http://www.iso.org/iso/en/ CatalogueDetailPage.CatalogueDetail?CSNUMBER=37890.

[17] "XDS - Cross-Enterprise Document Sharing," http://hcxw2k1.nist.gov/wiki/index.php/XDS_-_Cross-Enterprise_Document_Sharing.

[18] HL7 Clinical Document Architecture, Release 2.0. http://hl7.org/library/ Committees/structure/CDA.ReleaseTwo.CommitteeBallot03.Aug.2004.zip.

[19] HL7 Reference Information Model. http://www.hl7.org/v3ballot8/html/foundationdocuments/welcome/index.htm.

[20] "Logical Observation Identifiers Names and Codes (LOINC)," http://www. regenstrief.org/loinc/.

[21] "SNOMED Clinical Terms," http://www.snomed.org/.

[22] WHO, "International Statistical Classification of Diseases and Related Health Problems, 10th Revision (ICD-10), Second Edition," World Health Organization, Geneva, Switzerland, Tech. Rep., 2005, http://www.who.int/whosis/icd10/.

[23] "Nationaal ICT Instituut in de Zorg," http://www.nictiz.nl.

[24] "JAVA Agent Development Platform, JADE," http://jade.tilab.com/.

[25] "Spine- NHS Connecting for Health," http://www.connectingforhealth.nhs.uk/systems andservices/spine.

[26] "Canada Health Infoway," http://www.infoway-inforoute.ca/en/home/home.aspx.

[27] "FIPA Ontology Service Specification," http://www.tipa.org/specs/fipa00086/XC00086C.html.

[28] A. Dogac, G. Laleci, S. Kirbas, Y. Kabak, S. Sinir, A. Yildiz, and Y. Gurcan, "Artemis: Deploying Semantically Enriched Web Services in the Healthcare Domain," *Information Systems Journal, special issue on Semantic Web and Web Services*, vol. 31(4-5), pp. 321–339, 2006.

[29] V. Bicer, G. Laleci, A. Dogac, and Y. Kabak, "Artemis Message Exchange Framework: Semantic Interoperability of Exchanged Messages in the Healthcare Domain," *ACM Sigmod Record,*, vol. 34, no. 3, September 2005.

[30] "Unified Medical Language System (UMLS)," http://www.nlm.nih.gov/ research/umls/.

[31] "SAPHIRE Project Deliverable 5.2.1:Clinical Guideline models to be used in Clinical Decision Processes ," http://www.srdc.metu.edu.tr/webpage/projects/saphire/deliverables/SAPHIRE D5.2.1Final.doc.

[32] G. Laleci and A. Dogac, "SAPHIRE: A Multi-agent System for Deploying Semantically Enriched Clinical Guidelines in Heterogeneous Healthcare Environments," *Submitted to Elsevier Information Systems Journal.*

[33] "Jess, the Rule Engine for the Java Platform," http://herzberg.ca.sandia.gov/jess/.

[34] O. Nee, A. Hein, T. Gorath, N. H. G. B. Laleci, M. Yuksel, M. Olduz, I. Tasyurt, U. Orhan, A. Dogac, A. Fruntelata, S. Ghiorghe, and R.Ludwig, "SAPHIRE: Intelligent Healthcare Monitoring based on Semantic Interoperability Platform - Pilot Applications," *Accepted forIEEE Proceedings Communications-Special Issue on Telemedicine and a-Health Communication Systems.*

Gokce B. Laleci
Middle East Technical University
Inonu Bulvari, METU Computer Engineering SRDC 06531 Ankara / TURKEY
e-mail: banu@srdc.metu.edu.tr

Asuman Dogac
Middle East Technical University
Inonu Bulvari, METU Computer Engineering SRDC 06531 Ankara / TURKEY
e-mail: asuman@srdc.metu.edu.tr

Mehmet Olduz
Middle East Technical University
Inonu Bulvari, METU Computer Engineering SRDC 06531 Ankara / TURKEY
e-mail: mehmet@srdc.metu.edu.tr

Ibrahim Tasyurt
Middle East Technical University
Inonu Bulvari, METU Computer Engineering SRDC 06531 Ankara / TURKEY
e-mail: tasyurt@srdc.metu.edu.tr

Mustafa Yuksel
Middle East Technical University
Inonu Bulvari, METU Computer Engineering SRDC 06531 Ankara / TURKEY
e-mail: mustafa@srdc.metu.edu.tr

Alper Okcan
Middle East Technical University
Inonu Bulvari, METU Computer Engineering SRDC 06531 Ankara / TURKEY
e-mail: alper@srdc.metu.edu.tr

A Semantically Enriched Clinical Guideline Model Enabling Deployment in Heterogeneous Healthcare Environments

Gokce B. Laleci, Asuman Dogac, *Member, IEEE*

Abstract—
Clinical guidelines are developed to assist healthcare practitioners to make decisions on a patient's medical problems and as such they communicate with external applications to retrieve patient data, to initiate medical actions through clinical workflows and to transmit information to alert/reminder systems. The interoperability problems in the healthcare IT domain prevent wider deployment of clinical guidelines because each deployment requires a tedious custom adaptation phase.

In this article, we provide machine processable mechanisms that express the semantics of clinical guideline interfaces so that automated processes can be used to access the clinical resources for guideline deployment and execution. To be able to deploy the semantically extended guidelines to healthcare settings semi-automatically, the underlying application's semantics must also be available. We describe how this can be achieved based on two prominent implementation technologies in use in the eHealth domain: Integrating Healthcare Enterprise (IHE) Cross Enterprise Document Sharing Integration Profile (XDS) for discovering and exchanging Electronic Healthcare Records (EHRs) and Web service technology for interacting with the clinical workflows and wireless medical sensor devices. The system described in this article is realized within the scope of the SAPHIRE Project.

Index Terms—**Semantically Enriched clinical guidelines, Semi-automatic deployment and execution of guidelines, Semantic mediation**

I. Introduction

Clinical guidelines are systematically developed statements to assist general practitioners in making clinical decisions and managing medical actions more effectively [1]. They usually include plans for treatment and are used in developing the Clinical Decision Support systems.

Clinical guidelines aim to reduce inter-practice variations and the cost of the medical services, improve the quality of care and standardize clinical procedures [2]. A variety of government and professional organizations are producing and disseminating clinical guidelines [3], [4], [5]. However despite the benefits they provide, it is a well accepted fact that wider adoption of computerized clinical practice guidelines is yet to be realized. The reasons can be summarized as "the failure of integration of guideline implementations with clinical workflows" [6], [7] and "the complexity of fully integrated decision support systems due to the nature of heterogeneous set of clinical applications need to be involved in the decision process" [8].

Several computer interpretable models of Clinical Guidelines have been proposed such as GLIF [9], ASBRU [10], ARDEN [11], GUIDE [12] and EON [13]. Additionally, there are several guideline execution engines processing these models, such as GLEE [14], GLARE [15] and DeGel [16] demonstrating that the guideline definitions can be executed to automate the decision making process. However, these execution engines often address the automation in a single homogeneous healthcare organization and custom adaptation phases are required to communicate with clinical applications such as for accessing the patient records or invoking medical services. This lack of integration support in the guideline representation languages and guideline execution environments is stated in GLIF specifications [9] as follows: "There is a need for an implementable specification that can be incorporated into an institutional system where the actions specified must be mapped to institutional procedures and the patient data references must be mapped to the electronic medical records of the underlying system".

In this paper, we address this interoperability problem by developing a semantically enriched guideline model that enables the specification of enough level of semantics to facilitate semi-automatic deployment of guidelines. Furthermore a semantic infrastructure based on widely accepted healthcare standards is described for the semi-automatic deployment and automated execution of guidelines using the semantics encoded in the guideline model proposed. We choose to semantically enrich the GuideLine Interchange Format (GLIF) since GLIF model is formally expressed as an ontology. We extend this ontology to describe the semantics of GLIF execution steps based on the domain knowledge exposed by the related prominent healthcare IT standards as follows:

- *Accessing the Content of Electronic Healthcare Records (EHRs)*: The most prominent EHR standards are the Health Level 7 (HL7) Clinical Document Architecture (CDA) [17] and the European Committee for Standardization (CEN) EN 13606-1 EHRcom [18]. Investigating these standards reveals that to locate an EHR document and to extract the requested patient clinical information from the EHR document, the semantics needs to be explicated at two levels: at the EHR document semantics level to discover the related EHR and at the entry semantics level to extract the clinical statement requested by the guideline.

Gokce B. Laleci and Asuman Dogac are with the Middle East Technical University, Ankara, Turkey

This work is supported by the European Commission through IST-27074 SAPHIRE project and in part by the Scientific and Technical Research Council of Turkey, Project No: EEEAG 105E133

- *Accessing the medical services:* The success of a clinical guideline execution system widely depends on how well it is integrated with the clinical workflow running in healthcare organizations. Previous efforts either did not attempt to define the semantics of clinical workflow interfaces of guidelines, or identified a number of fixed event types that should be manually bound to the events supported by the underlying clinical workflow. We choose to define the operational semantics of medical services through ontologies, and to use this semantics to semi-automatically discover and bind the services of the clinical workflows to guideline execution environment. To define this ontology, we use well-established standards in the healthcare domain: for representing the semantics of clinical workflow services we exploited HL7 [19], which has categorized the events in healthcare domain by considering service functionality.
  Furthermore during the execution of a guideline, there are decision points where the vital signs of a patient are needed. Currently wireless medical sensor devices are widely used both in hospital and in homecare settings. Representing the operational semantics of services through an ontology enables us to extend the types of events that can be described in a guideline definition: in our functionality ontology we define the semantics of sensor services through the nomenclature codes defined by the IEEE 11073-10101 standard [20] for medical devices.

Providing the semantics of guideline execution steps solves only one part of the problem: when deploying the extended guideline model over existing clinical applications, the underlying applications' semantics must also be available so that matches can be discovered. Therefore the extended guideline model is complemented with a deployment and execution architecture by exposing the semantics of underlying clinical applications. Our aim is to enable the semi-automatic deployment and automatic execution of clinical guidelines in real life settings and this can be successful only if the implementation layer is based on open standards enabling interoperability [21]. Hence we based our deployment and execution architecture on widely accepted healthcare and IT standards supporting interoperability: Integrating Healthcare Enterprise (IHE) Cross Enterprise Document Sharing Integration Profile (XDS) [22] for discovering and exchanging Electronic Healthcare Records (EHRs) and the Web service technology for interacting with the clinical workflows. In this layer, we show that the semantics exposed by the prominent healthcare IT standards can be used to match the semantics of the guideline actions defined in the extended model for deploying the guidelines semi-automatically and for executing them automatically.

The system described in this paper is realized within the scope of the SAPHIRE Project [23] and is deployed through two pilot applications, one in a hospital environment and another for homecare [24].

The paper is organized as follows: In Section II, the GLIF guideline representation formalism is introduced briefly. In Section III, the semantic extensions we propose to the GLIF model are described and the deployment and the execution layers exploiting the extended guideline model are presented. Related work is described in Section IV. Finally Section V concludes the paper.

II. GLIF

The extended guideline model we describe in this paper is based on the GLIF (GuideLine Interchange Format) which is proposed as a standard representation model for sharing guidelines among different healthcare organizations [9]. In the GLIF model, clinical guidelines are represented as instances of a formal model called *guideline*. A part of the GLIF model is given in Figure 1.

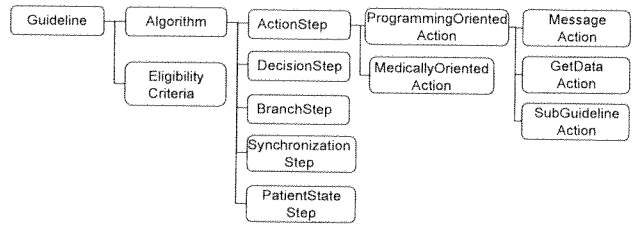


Fig. 1. A Part of the GLIF Model

The clinical process is represented through an *algorithm*, which is a flowchart of guideline steps, including:

- An *Action step* is used for modeling actions to be performed which may include two types of tasks: medically oriented actions such as recommendation for a particular course of treatment, or programming-oriented actions such as retrieving data from an electronic patient record.
- A *Decision step* represents decision points in the guideline defined in terms of formal expressions.
- The *Branch and Synchronization steps* allow modeling multiple simultaneous paths through the guideline.
- A *Patient state step* allows labeling patient states.

The GLIF Model represents the guidelines in three layers. The formal flowchart model described above (the Core GLIF model) constitutes the "Level A" or the conceptual level of the representation. GLIF proposes that in the second level of abstraction, that is, in "Level B", the medical concepts, the clinical data definitions, and the logical criteria in terms of the clinical data definitions should be defined formally.

Clinical data in GLIF is represented as *data items*. For representing clinical data items, GLIF supports the use of a Reference Information Model derived from HL7 RIM (USAM) [25]. This enables the formal definition of high level *Patient data items* such as *Medication, Observation, or Procedure*.

The *medical concepts* are used to annotate the *data items* to relate them with the well known medical terms. The medical concepts are defined through the tuple <conceptName, conceptID, conceptSource>. For example, the "chronic cough" concept can be represented through the following tuple <chronic cough, C0010201, UMLS> in reference to Unified Medical Language System (UMLS) semantic network [26].

GLIF uses these two levels of abstractions to represent a guideline definition as follows: Assume an overly simplified guideline that decides whether to prescribe "Aspirin" or

"Clopidogrel" as a firstline medication to a patient who is suffering from myocardial infarction. The first guideline step in the guideline's "algorithm" can be an "action step" where it is necessary to gather data from patient's Electronic Healthcare Record (EHR). The tasks included in this "action step" are two "get data actions". The first "get data action" instance as shown in Figure 2 is to discover whether the patient is currently using the medication "Aspirin". This action states that the information should be retrieved from the "EHR" (also termed as Electronic Medical Record, EMR). The second "get data action" is for accessing the contraindications of the patient to the medication "Aspirin". After executing these actions, through a "Decision Step", it is checked whether it is advisable to prescribe Aspirin to the patient as presented in Figure 2, in the "OrderAspirin" medically oriented action.

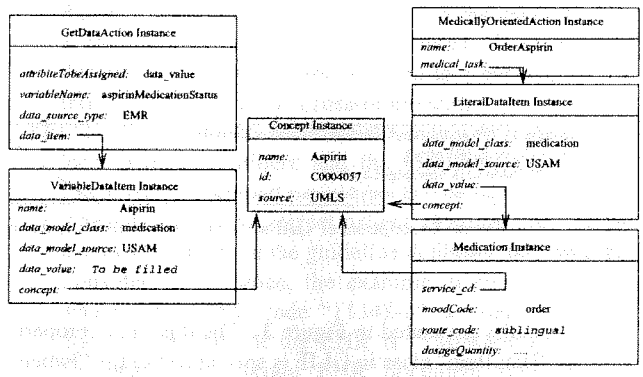


Fig. 2. An Example Get Data Action retrieving Aspirin Medication Status and an Example Medically Oriented Action Ordering Aspirin Prescription

As presented in this example, through GLIF Levels "A" and "B", a guideline is defined only at the conceptual level, as a sharable guideline model between medical practitioners. In other words, how this model can be executed at a specific healthcare organization, how the EHR of the patient can be accessed, how the clinical procedures can be discovered and invoked are not specified. GLIF recognizes this requirement by defining a third level of abstraction, "Level C" or "Medical Knowledge Layer", which aims to provide an implementable specification that can be incorporated into an institutional system. For this third level of abstraction, the following needs have been identified:

- Interfaces to clinical repositories to retrieve Electronic Healthcare Records of a patient,
- Interfaces for interacting with applications such as clinical workflows and alert systems.

In the latest GLIF specification, these requirements have been noted but not addressed.

III. Semantically Enriched Clinical Guideline Model and The Corresponding Deployment Architecture

The essence of extending GLIF is to provide a machine processable mechanism that can express the structure and the semantics of its "Medical Knowledge Layer" so that automated processes can be used to access the clinical resources for guideline deployment and execution. In our extended model, we describe the semantics needed at each guideline deployment and execution step in a machine processable way. The semantics of the proposed extensions are based on the standards in use in the healthcare IT domain.

GetDataAction ≡ (ProgrammingOrientedActionSpecification ∩
(∀ dataSourceType.({User} ∪ {EMR})))
MedicallyOrientedAction ≡ (ActionSpecification ∩ (∃ medicalTask.LiteralDataItem))
MessageAction ≡ (ActionSpecification ∩ (∃ message.xsd:String))
Concept ≡ (KnowledgeItem ∩ (∃ conceptID.xsd:String) ∩ (∃ conceptName.xsd:String)
∩ (∃ conceptSource.xsd:String))
DataItem ≡ (BasicDataType ∩ (∃ dataValue.DataItemValue))
PatientData ≡ (DataItemValue ∩ (∃ serviceCD.Concept))
Observation ⊆ (PatientData)

Fig. 3. A Part of the Formal GLIF Model

As already mentioned, the formal GLIF model is defined as an ontology for representing guidelines, medical data and concepts [27] as shown in Figure 3. In the following sections we present our semantic extensions iteratively on this model and also present how they can be utilized for deploying the clinical guidelines semi-automatically and for automatically executing them in heterogeneous healthcare settings.

A. Semantically Extending GLIF to Facilitate Accessing the Electronic Healthcare Record Systems

At various steps in executing a guideline, there is a need for a specific patient clinical data. For example, the guideline may need information on whether the patient has previously suffered from any "Corticoadrenal Insufficiency". To be able to retrieve this information, first various EHR systems serving patient clinical data must be discovered from a number of different healthcare organizations in which the patient has been previously treated and from these EHR Systems, the specific clinical information, such as "Corticoadrenal Insufficiency", must be extracted.

The original GLIF model uses a "GetDataAction" class to represent data to be retrieved from EHR Systems. However, since in its current form GLIF aims to create a sharable guideline definition, rather than a deployable and directly executable one, the *dataSourceType* property of this object is either the String "EMR" or the String "User" as presented in Figure 3, i.e., it does not have a mechanism to represent how to access the underlying EHR system.

Currently, the most prominent EHR standards are the Health Level 7 (HL7) Clinical Document Architecture (CDA) [17] and the European Committee for Standardization (CEN) EN 13606-1 EHRcom [18]. Investigating these standards reveals that to locate the EHR document and to extract the piece of information from the EHR document, the semantics needs to be explicated at two levels: the EHR document semantics to discover the document and the entry semantics to extract the clinical statement requested by the guideline.

In HL7 CDA documents, the type of the EHR document is presented in the document heading in the *code* attribute such as "Discharge Summary Note". In the document itself the clinical statements represented as "Entries" that are grouped under "Sections", such as "Past Medical History", "Medications" and "History of Present Illnesses". In order to locate an EHR

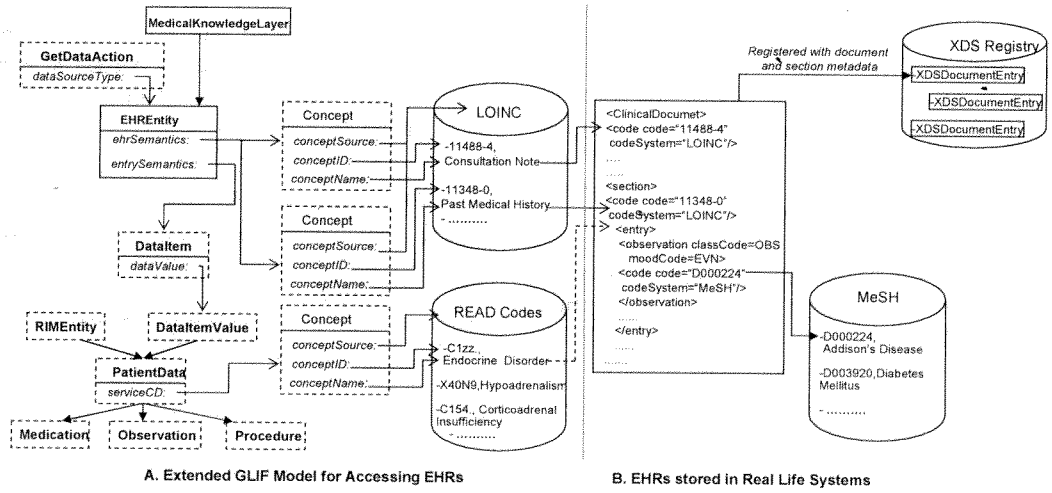


Fig. 4. Extending GLIF with further Semantics for Accessing EHRs

document that contains the requested clinical statement, both the document type semantics and the semantics of the relevant sections are needed.

Similarly in EHRCom, the "Compositions" represent a single EHR document, and the semantics of "Composition" presents the document type such as "Consultation Note". The "Sections" in EHRCom corresponds to the "Sections" in CDA, that is, the clinical statements are grouped under "Sections" such as "Adverse Reactions" and "Family Clinical History" based on their semantics. As in the CDA, in the EHRCom the "Composition" semantics together with the "Section" semantics represent the EHR semantics. The entry semantics, on the other hand, is readily provided by both of these standards.

In order to embed this semantics into GLIF "GetDataAction" class, first the range of the *dataSourceType* property is made to point to a newly defined "EHREntity" class in our extended model as shown in Figure 4 (The dashed boxes are the classes from the original GLIF Model). Then the following properties are defined for the "EHREntity" class:

- *ehrSemantics* property is used for annotating an EHR document with nodes from clinical terminologies or ontologies. As the range of this property, the "Concept" class (Figure 3) defined in GLIF is selected. In this way, it becomes possible to specify the EHR document and section semantics with formally defined terms from terminology systems. For example, the Document Type Code of LOINC can be used to specify this semantics with the following instance of the "Concept" class: <Past Medical History, 11348-0, LOINC>. This property can have multiple values, that is, in an "EHREntity" instance definition, the document type semantics and the semantics of the sections it contains can be specified through multiple instances of *ehrSemantics* property as presented in Figure 4.
- The *entrySemantics* property is introduced for annotating the EHR entry semantics. The range of this property is selected as the "DataItem" class as defined in GLIF (formally presented in Figure 3). The *dataValue* property of "DataItem" class in GLIF is specialized to the "PatientData" class which is further specialized to "Observation", "Medication", and "Procedure" classes. "PatientData" class has a property called *service_cd* inherited from HL7 RIM, whose range is the "Concept" class as shown in Figure 4. Through the instances of the "Concept" class, we represent the semantics of EHR entries, such as <Corticoadrenal Insufficiency, C154., READ>.

A formal description of these extensions is given in 5.

GetDataAction $\equiv$ (ProgrammingOrientedActionSpecification $\cap$
($\exists$ dataSourceType.EHREntity))
EHREntity $\equiv$ (MedicalKnowledgeLayer $\cap$
($\exists$ ehrSemantics.Concept) $\cap$ ($\exists$ entrySemantics.DataItem))

Fig. 5. The Formal Description of GetDataAction and EHREntity in the Extended GLIF Model

B. Exploiting the EHREntity Semantics to locate and access Electronic Healthcare Records

Through the extended GLIF Model, we are able express the detailed semantics of what the guideline deployment and execution engine expects from the underlying EHR systems. In this section, we describe how this semantics can be exploited to locate and process machine processable Electronic Healthcare Records of the patient. We base the underlying semantic deployment architecture on a prominent initiative for sharing EHRs which is the Integrating Healthcare Enterprise (IHE) Cross Enterprise Document Sharing Integration Profile (XDS) [22]. This profile has received considerable attention and appeared in the National eHealth System blueprints of Canada, Italy, Norway and France.

In the IHE XDS Profile, healthcare enterprises that agree to work together for clinical document sharing are called a "Clinical Affinity Domain". In each affinity domain, the healthcare organizations store the medical documents of the patients to "Document Repositories" of their choice. There

is a "Document Registry" which is responsible for storing metadata about those documents so that the documents of interest for the care of a patient can be easily discovered, selected and retrieved irrespective of the repository where they are actually stored. The document repositories register the documents along with a set of metadata to the Document Registry.

The extended GLIF Model is deployed on the IHE XDS layer as follows:

- The first step is to devise mechanisms to locate the relevant Electronic Healthcare Records of a patient. The use of patient identifiers is the accepted practice for locating patient EHRs. However, each organization may (and typically will) have a different patient identifier domain. Additionally, in the IHE XDS profile, the Document Registry also assigns a unique ID to the patient when a document of the patient is registered. To resolve the different patient identifiers assigned to a patient, we use IHE "Patient Identity Cross-referencing" (PIX) Profile [22]. PIX Managers facilitate the mappings between the local patient identifiers and the identifiers used in the DocumentRegistry.
  After identfying the correct patient identifier, the "ehrSemantics" property in the guideline definition is processed. Consider, for instance, the example given in Figure 4, where the LOINC code "11348-0" is used to indicate that the metadata of the EHR is "PastMedicalHistory". To locate the relevant EHR document references, a "queryRegistry" transaction is issued to IHE Document Registry. As a response, the list of document references is returned. In this way these documents can be retrieved from the Repository through the "retrieveDocument" IHE transaction, while the guideline is executed.
- After the required EHR is located through the XDS Registries, in order to execute the clinical guidelines, there is a need to extract a specific clinical statement from the EHR of the patient. For example, the guideline may need information on whether the patient has experienced any "corticoadrenal insufficiency" previously. From the retrieved EHR, this specific clinical information must be extracted. It is clear that to achieve this, the EHR must be available in a machine processable content standard such as CEN EHRcom [18] or HL7 CDA [17]. In addition to this, since different EHR document formats and coding schemes can be used to represent the EHR content, there should be a semantic mediation mechanism available to use the data retrieved from the documents.

In the following, we describe how clinical statement data can be extracted from a machine processable EHR by using semantic mechanisms proposed in this work. First we give an insight to the problem through an example based on HL7 CDA [17]. In Figure 4, on the right hand side, a part of a sample HL7 CDA document is given. In this document, the semantics of a section is annotated with the "11348-0" code from LOINC to describe that this section is about the "Past Medical History". Additionally each entry is an instance of HL7 RIM classes such as the "Observation" class, and their content is also annotated with coded terms. For example, in one of the "Observation" class instances, the MeSH code "D000224" is used to indicate that the patient has suffered from "Addison Disease".

On the other hand, as presented in Figure 4 in the extended GLIF "GetDataAction" class, the type of the data to be extracted from the EHR document is specified through the "ehrSemantics" and "entrySemantics" attributes.

```
if(glifCS!= DocCS) then
  targetSectionSemanticCategory= map(glifSectionSemanticCategory, glifCS,DocCS);
else targetSectionSemanticCategory=glifSectionSemanticCategory;

Section targetSection = sectionImport(EHRDoc,targetSectionSemanticCategory);
Entry targetEntry = discreteDataImport(targetSection,glifEntrySemanticCategory,glifCS);

function discreteDataImport(targetSection,glifEntrySemanticCategory,glifCS)
  forall Entry e in targetSection
        {entryCS=(e.getCode()).getCS();
        if(entryCS!= glifCS) then
          targetSemanticCategory= map(entrySemanticCategory,entryCS,glifCS);
        else targetSemanticCategory=entrySemanticCategory;
        if(targetSemanticCategory == glifEntrySemanticCategory)
        then return e;
        }
```

Fig. 6. The Discrete Data Import Algorithm in SAPHIRE

The algorithm we use in extracting the Clinical Statement requested in a guideline specification using the entry semantics is presented in Figure 6.

First it is necessary to locate the relevant Section of the EHR document. Then for retrieving the particular Clinical statement from this Section, the "discreteDataImport" function in Figure 6 is used. The coded terms used to annotate the entries in the CDA document, and the entry semantics in guideline definition may be different. For example, in the example CDA document presented in Figure 4, the entry is annotated with a MesH code while a READ code is used to annotate the entry semantics in the extended guideline definition.

To find the correspondences among different coding schemes, it is possible to use a terminology server. A medical terminology server serves a controlled vocabulary for medical information systems by modeling medical concepts

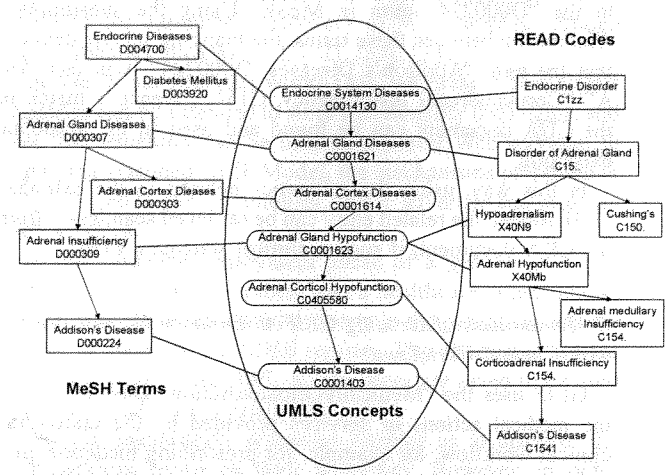


Fig. 7. Term mappings

in a descriptive manner. Among other functionalities, medical terminology servers handle the mapping of the medical concepts to Medical classification schemes; through this facility they also provide translation facilities between the terms of different classification schemes. GALEN [28] is one of the first implementations of such a medical terminology server which defines medical concepts in descriptive logics supporting subsumption. Another such terminology server is UMLS: the UMLS Knowledge Source Server Metathesaurus [26] comprises over one million biomedical concepts and five million concept names, all of which are from over a hundred controlled vocabularies and classification systems such as ICD-9, SNOMED and LOINC, providing a mapping structure between them. In our architecture a mapping functionality supporting subsumption is used based on the concepts and mappings provided by UMLS.

In Figure 7, a part of the UMLS Concept hierarchy and mappings of these concepts to different terminology systems is presented as defined by UMLS Knowledge Metathesaurus. Once such mapping definitions are stored in an ontology server supporting subsumption, it becomes possible to deduce implicit relationships and mapping between the terms of different coding schemes. Our mapping service queries the Ontology Server bootstrapped with the UMLS concepts and term mappings to different terminology systems, in order to discover the respective equivalent terms of the coded terms.

As an example, in Figure 4-A, the READ Code "C154." is specified as the "entrySemantics" in the GLIF definition, to be able to check whether the patient has previously suffered from "Corticoadrenal Insufficiency". However in the example CDA document in Figure 4-B, the MeSH codes are used to represent this semantics.

As seen in the Figure 7, there is a direct mapping between the term "C154." and UMLS Concept "Adrenal Corticol Hypofunction, C0405580", however this term does not have a direct correspondent in MeSH codes. On the other hand, from the UMLS Concept hierarchy it is clear that the concept "Addison's Disease, C0001403" is a subclass of the UMLS Concept "Adrenal Corticol Hypofunction, C0405580", and the concept "Addison's Disease, C0001403" has a direct mapping to the "D000224" term in MeSH. Using the subsumption relationship between these terms, the mapping service deduces that the term "Addison's Diseases, D000224" is in fact "IS-A" "Corticoadrenal Insufficiency, C154.". Hence a match in the CDA document can be found and used in the guideline execution.

In this way, using the semantics defined in the extended GLIF model, the related entry can be retrieved seamlessly from the CDA document, if such a clinical statement exists.

C. Semantically Extending GLIF to Enhance Communication with Clinical Workflows

GLIF uses the "MedicallyOrientedAction" class to specify the medical actions or services provided by the underlying clinical workflow, for example, for prescribing medicine, giving lab orders or making referrals. Currently these tasks are specified with the *medicalTask* property whose range is the *LiteralDataItem* class as presented in Figure 3. As already mentioned in the example in Section II, the *LiteralDataItem* is not capable of describing the semantics of a specific medical action. In order to be able to define an implementable guideline specification, it is necessary to propose a solution that can address the technical interoperability problem capable of accessing various different clinical workflow systems. Web service technology is already being used by the Healthcare Industry as a solution to technical interoperability problem.

When the medical information systems expose their medical applications as Web services for interacting with the clinical workflows such as placing lab orders; the endpoints of these Web services (references to their Web Service Description Language (WSDL) definitions) can be used in clinical guideline definitions. However, in a generic guideline definition that has not been specialized for a specific patient and a healthcare organization, it is not feasible to directly refer to the end point of a Web service. Therefore, we propose to specify the semantic definitions of the Web services in the guideline definition rather than the references to their concrete implementations. This also enables us to find alternative resources when exceptions are raised in accessing the specified Web Service in the execution phase.

MedicallyOrientedAction $\equiv$ (ActionSpecification $\cap$ ($\exists$ medicalTask.MedicalActionEntity))
MedicalActionEntity $\equiv$ (MedicalKnowledgeLayer $\cap$ ($\exists$ functionality.Concept) $\cap$
($\exists$ input.DataItem) $\cap$ ($\exists$ output.DataItem))

Fig. 8. The Formal Description of MedicallyOrientedAction and MedicalActionEntity in the Extended GLIF Model

To be able to define the semantics of the medical action to be interacted during the guideline execution, the GLIF model is extended as follows: First, the "MedicalActionEntity" class is created as a subclass of "MedicalKnowledgeLayer". The range of *medicalTask* property of the "MedicallyOrientedAction" class is set to be the "MedicalActionEntity" (Formally defined in Figure 8 and graphically presented in Figure 9). The "MedicalActionEntity" class is used to specify the semantics of the guideline execution steps which correspond to clinical services.

To describe this semantics, we define a *functionality* property for the "MedicalActionEntity" class whose range is selected as the "Concept" class defined in the GLIF Model. In this way, it becomes possible to annotate the clinical services with a node from a clinical terminology or ontology.

To be able to define the semantics of Healthcare services, we define a functionality ontology based on the UMLS Semantic Network [26], HL7 event definitions [19], and IEEE 11073 Nomenclature codes [20]. We use this ontology to annotate the operational semantics of the clinical services. For example, Figure 9 shows how the "MedicalActionEntity" class can be annotated through a service Functionality Ontology. The advantages of annotating the operational semantics of Medical Actions through ontologies rather than binding them to some pre-defined action types in the guideline model are as follows:

- The ontology is easily extensible allowing us to widen the scope of medical actions that the guideline needs to invoke in clinical workflows without the need to update

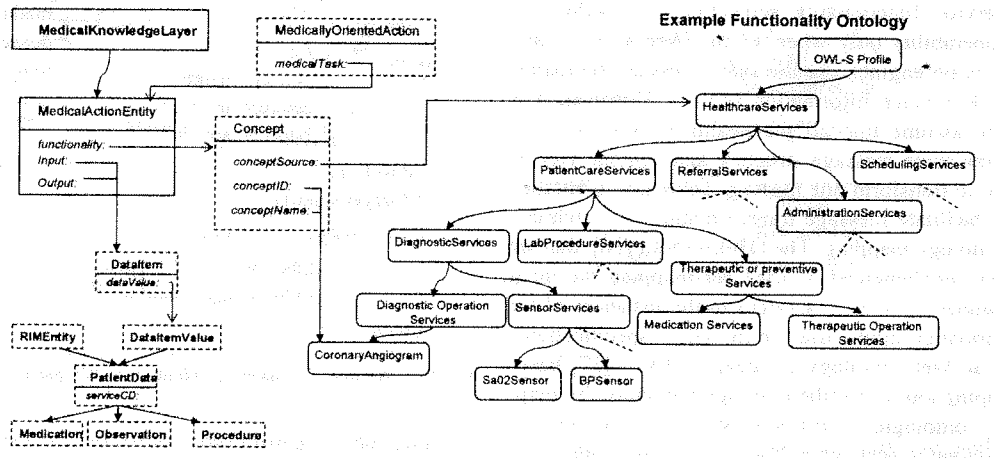


Fig. 9. Annotating "MedicalActionEntity" class with the Service Functionality Ontology

the whole guideline model. For example, currently wireless medical sensor devices are widely used in clinical environments especially in homecare settings. In our architecture, the semantics of services exposing the data produced by wireless medical sensor devices are defined in the functionality ontology, under the node of "Diagnostic Services", where the "IEEE 11073 Nomencalature codes" are exploited to create a hierarchy of sensor device services. This capability moves the semantic support beyond what is currently available in guideline models; it becomes possible to easily extend the type of actions that the guideline deployment/execution engine can interact with.

- Medical Information systems can expose the services of the clinical workflows by annotating their semantics with any ontology of their choice. When the semantics of the actions in the guideline definition are also defined through ontologies, it becomes possible to use "ontology mapping/alignment" mechanisms for service matching. In the deployment phase, although different ontologies may be used to annotate the services of the clinical workflows and the actions in guideline definition, it becomes possible to locate the corresponding clinical services of the guideline actions through semantic mediation.

This semantics is used during the guideline deployment to describe the functionality of the service needed. Symmetrically, there is a need to expose the semantics of the underlying technical interoperability layer. How to map this layer to technical interoperability layer is detailed in Section III-D.

D. Exploiting the MedicalActionEntity Semantics to locate and access Clinical Workflow Services

In deploying the extended GLIF Model over a Web service based technical interoperability layer, the Service Functionality Ontology (SFO) is expressed through the "ServiceProfile" class of OWL-S [29]. For this purpose, as presented in Figure 9 and in Figure 10, the top-most class of the SFO is created as a subclass of OWL-S ServiceProfile class. To express the service

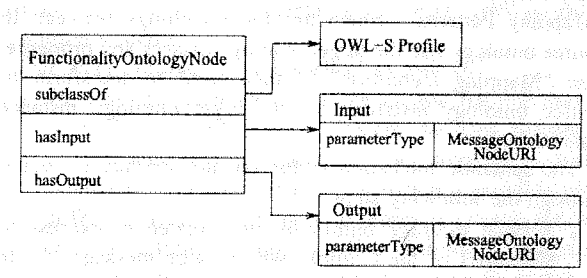


Fig. 10. Relating Service Functionality and Service Message Ontologies with OWL-S

functionality, the ServiceProfile instance of a specific Web service is represented as an instance of one of the Functionality ontology nodes. In this way the semantic annotation of Web Services through one of the SFO nodes is achieved. Once the OWL-S descriptions are defined, the Web Services can be automatically published and discovered through an UDDI Registry [30] by using the functionality semantics expressed in OWL-S. In the literature several methodologies have been proposed for automated discovery of Web services through their semantics such as [31], [32], [33], [34].

In one of our previous works [35], we present how UDDI registries can be extended to store functionality ontologies, and how this semantics can be used to discover Web services. In deploying extended GLIF Model, the mechanisms presented in [35] are used to store ontologies, and to advertise and discover Web services.

In the guideline deployment phase of a generic guideline definition for a specific patient and a healthcare organization , the range of the functionality property in the guideline instance is used in querying the UDDI registry. The discovered service instances' WSDL and OWL-S references are saved to be used while the guideline is executed.

Service functionality semantics enables us to discover the Web services based on their semantics. However, in order to invoke the discovered Web services while the guideline is executing, message level interoperability should also be

addressed. Service functionality semantics may suffice to achieve interoperability only when all the Web services and guideline execution engines use the same message standards, and the same Reference Information Models. However, it is not realistic to assume that all the healthcare organizations comply with the same message structure and content. Hence, there is a need to transform one message content into another.

In order to facilitate message transformation, our architecture utilizes ontology mapping. The OWLmt tool [36], that we have previously implemented, is used for mapping the input and output parameters of Web services to the instances of the reference information model used in the GLIF specification.

OWLmt is an Web Ontology Language (OWL) [37] based ontology mapping tool to handle ontology mediation by mapping the OWL ontologies in different structures and with an overlapping content to each other. It aims to define a document called the "Mapping Definition" describing how the source ontology and the target ontology classes and properties relate. This document includes the units of information called the "Mapping Patterns", which are the matchings between the source ontology and the target ontology classes and properties. The "Mapping Definition" is then used to transform the source ontology instances to the target ontology instances automatically.

The semantic mediation in the our architecture is enabled through the following steps:

- Service message ontologies are created to express the semantics of the Clinical Web service messages. In the healthcare domain, the Web services usually exchange XML messages. Through a normalization tool [38], we create the OWL ontology of the Web service messages from the XML Schema (XSD) [39] definitions of the service messages.
  It is clear that when an OWL ontology is created from an XML Schema (XSD), it is not possible to extract some of the OWL specific semantics such as class expressions or various types of properties. Yet, it is still possible to obtain the class hierarchies and the properties of classes and this information proves useful in ontology mapping.
- Based on the Service Functionality and Service Message Ontologies, Web services are annotated in our architecture through OWL-S [29] as depicted in Figure 10. The OWL-S Profile class has properties called *hasInput* and *hasOutput* whose ranges are "Input" and "Output" classes. These classes, in return have a property, namely, the *parameterType*. The value of this property is set to a node in a local message ontology. In this way, the service's input and output parameters are annotated with the service message ontologies.
- In the GLIF model, the data exchange with the external resources is realized through the instances of Procedure, Medication and Observation RIM classes (Figure 9). We have created an XML schema (XSD) of these classes. Then, the OWL ontology from the XSD file is created automatically through the Normalization Tool.
- The next step is to create the mapping definition between the GLIF RIM ontology and Clinical Web service Message Ontologies through the graphical interface of the OWLmt tool. An example mapping is illustrated in Fig 11. On the left hand side, the "Medication" concept GLIF RIM ontology is presented, the parts of which mapped to the "Medication" instance expected by Web Service. Apart from the copy and the concatenati functionalities, the OWLmt tool also allows the user define more advanced transformation functions in ter of Java scripts.

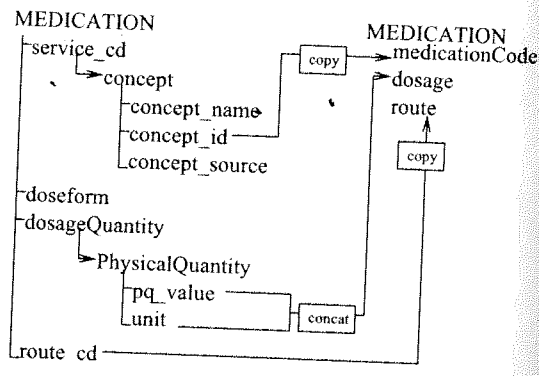


Fig. 11. An example mapping definition of WS parameters

- Once such a "Mapping Definition" is created graphical by the OWLmt tool, the mediation of message instanc are achieved as presented in Figure 12:
  - While the guideline is being executed, the Web se vice parameters are provided as the XML instanc of the Reference Information Model used in GLIF
  - These XML instances are normalized to OWL i stances of the Reference Information Model used GLIF.
  - The OWL instances are automatically transformed t the instances of the Clinical Web Service messag ontologies through the OWLmt tool mapping engin using the "Mapping Definitions" previously created
  - The instances of the Clinical Web Service messag ontologies are normalized to XML instances whic are the messages the Clinical Web services ar expecting to receive.
- The same procedure is followed when a response is received from the Web Service.

In this way, the guideline execution environment always processes the data encoded as instances of the GLIF RIM, as proposed in the GLIF specification; the Clinical Web Services always process the input, output parameters as the instances of the Reference Information Model used internally in the respective healthcare organizations.

IV. Related Work

The difficulty of deploying guideline implementations to healthcare organizations has been addressed in the literature where achieving interoperability with the underlying systems is highlighted as the key challenge.

Some clinical guideline execution engines such as GLEE [14] expect the local healthcare organizations to store their

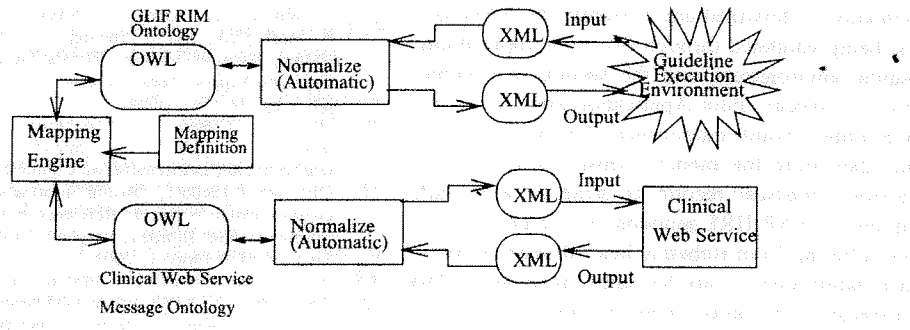


Fig. 12. Semantic Mediation of Clinical Web Service Messages

data in a centralized repository conforming to the guideline data model.

In some other guideline models such as PROforma [40] and GUIDE [12], the relational database tables are used to store the mappings between the guideline model entities and the "columns of the database tables" where the patient data is actually stored [41].

The extended GLIF Model that we propose aims to facilitate guideline deployment through semantic mediation rather than hard coding the underlying systems' reference information models to the guideline, or assuming a central repository that provides all the data in the format needed by the guideline. We describe both the semantics of the guideline steps accessing external resources and the semantics of the interfaces of the underlying applications. Our architecture proposes a guideline deployment and execution environment where the resources are semi-automatically discovered and bound to the guideline definition and semantic mediation is exploited when necessary.

In EON [13], ProDIGY [42] and SAGE [43], the "Virtual Medical Record" (vMR) [44] concept is used which provides a structured data model for representing information related to individual patients where it is assumed that mapping of local EHR to the vMR is facilitated. For example, in SAGE [43] which addresses the integration of guideline-based decision support systems with the workflow of care process, a guideline is encoded using the Virtual Medical Record (vMR). The vMR is based on the HL7 RIM and gives a simplified view of a patients medical record data. SAGE proposes a methodology for representing clinical content in a standard way through vMR's supported with "Clinical Expression Models (CEMs) which place constraints on the attributes of vMR classes. A compositional method is proposed to express new complex concepts.

For technical interoperability with the underlying clinical information systems, SAGE proposes a set of fixed Action Types for abstracting Clinical Information System's (CIS) functions (such as Notify, Inquire, Recommend.OrderSet) inside the guideline model [45]. SAGE engine supports communication with clinical workflows through events based on these action types. In SAGE methodology, before a formalized guideline can be installed and used in a local organization, its medical content must be reviewed and revised (localization process) and its data models, terminologies, and organization assumptions (roles, events, and resources) must be mapped to those of the local organization by the medical staff (the binding process) manually. SAGE assumes a set of standard vMR/Action Service Interfaces will be set by standard bodies [46], and clinical information systems will be using these standard messages, vMR interfaces to interact with guideline execution systems.

Our work complements this effort as follows:

- We describe the semantics of the actions that will interact with Clinical Information System services through a service functionality ontology. This ontology is easily extensible, and also semantic mediation is possible when different ontologies are used to represent clinical services and the actions in the guideline definitions. Additionally we facilitate a semi-automatic semantic matching, discovering and binding process based on this machine processable semantics of the guideline interfaces and the underlying clinical applications.
- When different message schemas are used by clinical information systems and guideline execution systems, we support a semantic mediation mechanism.

Ontologies have been successfully used for representing and supporting implementation of clinical guidelines by a number of efforts [47], [48]. In [47], a medical application ontology that is inspired from UMLS Semantic Metathesaurus is used by a multi-agent system for coordinating their activities in the enactment of clinical guidelines. In [48], a DAML+OIL based Context-Task-Ontology is proposed in order to model the knowledge required for the implementation of guidelines. This ontology is again based on UMLS. The Service Functionality Ontology we propose is inspired by this work. Finally in [12], organizational and medical ontologies are shared among Clinical workflow systems and Guideline management systems. These studies used ontologies for modeling and supporting the execution of clinical guidelines as a common understandable semantic framework. Our approach complements these efforts by exploiting ontologies as a semantic framework for semi-automatic deployment of clinical guidelines in healthcare settings by annotating and discovering the necessary interfaces of clinical guidelines and enabling semantic mediation when necessary.

V. Conclusion

The system architecture described in this paper is realized within the scope of IST-27074 SAPHIRE project. The proto-

type implementation is finalized and currently the SAPHIRE architecture is being validated through two pilot applications: one in a hospital environment, and the second in a homecare application. Homecare Pilot Application aims to test the intelligent intermittent home monitoring of patients through integrating the data from the medical sensors with the data from the electronic medical records to generate alerts and recommendations via SAPHIRE guideline execution environment. Patients suffering from ischemic heart disease followed by a revascularization therapy are the target population. The SAPHIRE Homecare Application details are provided in [49], [24].

The purpose of the hospital application is to demonstrate that the SAPHIRE system developed can provide bedside intelligent healthcare monitoring through wireless sensors and can also provide patient-specific computer-generated clinical decision in accordance with the latest European Cardiology Guidelines [24]. The end users of the project envision that the system will not only reduce the workload in the hospital and out-patient departments, but also will diminish probability of human errors and reduce medical costs by cutting down the stay in intensive care units.

REFERENCES

[1] Institute of Medicine, *Crossing the Quality Chasm: A New Health System for the 21st Century*. Washington DC: National Academy Press, 2001.

[2] M. J. Field and K. N. Lohr, *Guidelines for Clinical Practice: From development to use*. Washington DC: Institute of Medicine, National Academy Press, 1992.

[3] "US National Guideline Clearinghouse," http://www.guideline.gov/.

[4] "Danish Secretariat for Clinical Guidelines," http://www.sst.dk/Planlaegning_og_behandling/SfR.aspx?lang=en.

[5] "National Institute for Clinical Excellence-England/Wales (NICE) Published Guidelines," http://www.nice.org.uk/page.aspx?o=guidelines.completed.

[6] M. Fieschi *et al.*, "Medical decision support systems: Old dilemmas and new paradigms?" *Methods Inf Med.*, vol. 42(3), pp. 190–198, 2003.

[7] R. N. Shiffman, Y. Liaw, C. A. Brandt, and G. J. Corb., "Computer-based Guideline Implementation Systems: A Systematic Review of Functionality and Effectiveness," *Journal of American Medical Informatics Assoc.*, vol. 6(2), pp. 104–114, Mar-Apr 1999.

[8] M. Entwistle and R. N. Shiffman, "Turning Guidelines into Practice: Making It Happen With Standards - Part 1," *Healthcare and Informatics Review Online, March 2005*, 2005.

[9] "Guideline Interchange Format (GLIF) 3," InterMed Collaboratory, Tech. Rep., 2004.

[10] A. Seyfang, S. Miksch, and M. Marcos, " Combining Diagnosis and Treatment using Asbru," *International Journal of Medical Informatics*, vol. 68 (1-3), pp. 49–57, 2002.

[11] M. Peleg, O. Ogunyemi, and S. Tu, "Using features of Arden Syntax with object-oriented medical data models for guideline modeling," in *Proceedings of AMIA Symposium*, 2001, pp. 523–527.

[12] P. Ciccarese, E. Caffi, L. Boiocchi, S. Quaglini, and M. Stefanelli, "A Guideline Management System," in *Proceedings of MedInfo 2004*, 2004, pp. 28–32.

[13] S. Tu and M. Musen, "Modeling Data and Knowledge in the EON Guideline Architecture," in *Proceedings of MedInfo 2001*, London, UK, 2001, pp. 280–284.

[14] D. Wang, M. Peleg, S. Tu, A. Boxwala, O. Ogunyemi, Q. Zeng, A. Greenes, V. Patel, and E. Shortliffe, "Design and Implementation of GLIF3 guideline execution engine," *Journal of Biomedical Informatics.*, vol. 37, pp. 305–318, 2004.

[15] P. Terenziani, S. Montani, A. Bottrighi, and M. T. ans G. Molino, "Managing Clinical Guidelines Contextualization in the GLARE System," *Lecture Notes in Artificial Intelligence*, vol. 3673, pp. 454–465, 2005.

[16] Y. Shahar, O. Young, E. Shalom, M. Galperin, A. Mayaffit, R. Moskovitch, and A. Hessing, " A Framework for a Distributed, Hybrid, Multiple-Ontology Clinical-Guideline Library and Automated Guideline-Support Tools," *Journal of Biomedical Informatics.*, vol. 37, no. 5, pp. 325–344, 2004.

[17] HL7 Clinical Document Architecture, Release 2.0. http://hl7.org/library/Committees/structure/CDA.ReleaseTwo.CommitteeBallot03.Aug.2004.zip.

[18] CEN, prEN 13606-1, "Health informatics – Electronic health record communication – Part 1: Reference model," European Committee for Standardization, Brussels, Belgium, Draft European Standard for CEN Enquiry prEN 13606-1, 2004.

[19] Health Level Seven (HL7). http://hl7.org.

[20] "ISO/IEEE 11073-10101:2004:Health informatics, Point-of-care medical device communication, Part 10101: Nomenclature," http://www.iso.org/iso/en/CatalogueDetailPage.CatalogueDetail?CSNUMBER=37890.

[21] T. Leong, K. Kaiser, and S. Miksch., "Free and Open Source Enabling Technologies for Patient-Centric, Guideline-Based Clinical Decision Support: A Survey," *IMIA Yearbook of Medical Informatics, Methods of Information in Medicine*, vol. 46-1, pp. 74–86, 2007.

[22] "IHE IT Infrastructure Technical Framework," http://www.ihe.net/Technical_Framework/upload/ihe_iti_tf_2.0_vol1_FT_2005-08-15.pdf.

[23] "Intelligent Healthcare Monitoring based on Semantic Interoperability Platform," http://www.srdc.metu.edu.tr/webpage/projects/saphire/.

[24] O. Nee, A. Hein, T. Gorath, N. Hulsmann, G. Laleci, M. Yuksel, M. Olduz, I. Tasyurt, U. Orhan, A. Dogac, A. Fruntelata, S. Ghiorghe, and R. Ludwig, " SAPHIRE: Intelligent Healthcare Monitoring based on Semantic Interoperability Platform - Pilot Applications," *IEE Proceedings Communications-Special Issue on Telemedicine and e-Health Communication System*, vol. June, 2007, 2007.

[25] HL7 Reference Information Model. http://www.hl7.org/v3ballot8/html/foundationdocuments/welcome/index.htm.

[26] "Unified Medical Language System (UMLS)," http://www.nlm.nih.gov/research/umls/.

[27] "The GLIF Ontology," http://www.smi.stanford.edu/projects/intermedweb/guidelines/Protege_Ontology.htm.

[28] S. K. Bechhofer, C. A. Goble, A. L. Rector, and W. D. Solomon, "Terminologies and terminology servers for information environments," in *Proceedings of the 8th International Workshop on Software Technology and Engineering Practice*, 1997.

[29] OWL-S 1.0 Release. http://www.daml.org/services/owl-s/1.0/.

[30] "Universal Description, Discovery and Integration (UDDI)," http://www.uddi.org/.

[31] K. Verma, K. Sivashanmugam, A. Sheth, A. Patil, S. Oundhakar, and J. Miller, "METEOR-S WSDI: A Scalable Infrastructure of Registries for Semantic Publication and Discovery of Web Services," *Journal of Information Technology and Management, Special Issue on Universal Global Integration*, vol. 6, no. 1, pp. 17–3, 2005.

[32] M. Paolucci, T. Kawamura, T. Payne, and K. Sycara, "Matching of Web Service Capabilities," in *Proceedings of ISWC 2002*, pp. 333–347.

[33] "Web Service Modeling Ontology (WSMO), WSMO Working Draft 20 September 2004," http://www.wsmo.org/2004/d2/v1.0/.

[34] L. Li and I. Horrocks, "A Software Framework for Matchmaking Based on Semantic Web Technology," in *Proceedings of WWW'03*, Budapest, Hungary, 2003.

[35] A. Dogac, I. Cingil, G. Laleci, and Y. Kabak, "Improving the Functionality of UDDI Registries through Web Service Semantics," in *Proceedings of 3rd VLDB Workshop on Technologies for E-Services (TES-02)*, August 23-24, 2002.

[36] V. Bicer, G. Laleci, A. Dogac, and Y. Kabak, "Artemis Message Exchange Framework: Semantic Interoperability of Exchanged Messages in the Healthcare Domain," *ACM Sigmod Record.*, vol. 34, no. 3, September 2005.

[37] "OWL Web Ontology Language Overview," http://www.w3.org/TR/owl-features/.

[38] "Harmonise, IST-200029329, Tourism Harmonisation Network, Deliverable 3.2 Semantic mapping and Reconciliation Engine subsystems."

[39] "XML Scheme Definition (XSD)," http://www.w3.org/XML/Schema.

[40] D. Sutton and J. Fox, "The Syntax and Semantics of the PROforma guideline modelling language," *J Am Med Inform Assoc.*, vol. Sep-Oct;10, no. 5, pp. 433–443, 2003.

[41] M. Peleg, S. Tu, J. Bury, P. Ciccarese, J. Fox, R. Greenes, R. Hall, P. Johnson, N. Jones, A. Kumar, S. Miksch, S. Quaglini, A. Seyfang, E. Shortliffe, and M. Stefanelli, "Comparing computer-interpretable guideline models: a case-study approach," *J Am Med Inform Assoc.*, vol. Jan-Feb;10, no. 1, pp. 52–68, 2003.

[42] P. Johnson, S. Tu, and N. Jones, "Achieving Reuse of Computable Guideline Systems," in *Proceedings of MedInfo 2001*, London, UK, 2001, pp. 99–103.

[43] S. Tu, M. Musen, and R. Shankar, "Modeling Guidelines for Integration into Clinical Workflow," in *Proceedings of MedInfo 2004*, 2004, pp. 174–178.

[44] P. Johnson, S. Tu, M. Musen, and I. Purves, "A Virtual Medical Record for Guideline-based Decision Support," in *Proceedings of AMIA Annual Symposium*, Washington, DC: Hanley and Belfus, 2001.

[45] "Sage guideline model specification document," http://sage.wherever.org/references/docs/SAGEGuidelineModelSpec.pdf.

[46] P. Ram, D. Berg, S. Tu, G. Mansfield, Q. Ye, R. Abarbanel, and N. Beard, "Executing clinical practice guidelines using the SAGE execution engine," in *MedInfo*, 2004, pp. 251–255.

[47] D. Isern, D. Sanchez, and A. Moreno, "Agents and clinical guidelines: Filling the semantic gap," in *Frontiers in Artificial Intelligence and Applicationst*. IOS Press, 2007, pp. 67–76.

[48] K. Anand, C. Paolo, S. Barry, and P. Matteo, "Context-Based Task Ontologies for Clinical Guidelines," in *Ontologies in Medicine: Proceedings of the Workshop on Medical Ontologies*. IOS Press, 2003.

[49] A. Hein, O. Nee, D. Willemsen, T. Scheffold, A. Dogac, and G. Laleci, "SAPHIRE - Intelligent Healthcare Monitoring based on Semantic Interoperability Platform - The Homecare Scenario ," in *Proceedings of the 1st European Conference on eHealth (ECEH)*, Geneva, Switzerland, 2006.

SAPHIRE: intelligent healthcare monitoring based on semantic interoperability platform: pilot applications

O. Nee, A. Hein, T. Gorath, N. Hülsmann, G.B. Laleci, M. Yuksel, M. Olduz, I. Tasyurt, U. Orhan, A. Dogac, A. Fruntelata, S. Ghiorghe and R. Ludwig

Abstract: As a response to the challenge of providing high-quality healthcare services with reasonable costs while the elderly population and the associated chronic diseases increase, SAPHIRE architecture provides an intelligent healthcare monitoring architecture. The monitoring of patients is achieved through a clinical decision support system based on clinical guidelines. SAPHIRE provides the necessary interoperability layers to access the patient's vital signs from wireless medical sensors and the electronic healthcare records of the patient in order to exploit them in the decision process seamlessly. This architecture is presented through two pilot applications: one for the bedside monitoring of cardiac patients at hospitals, and the other for homecare monitoring of the cardiac patients rehabilitated after a revascularisation therapy.

1 Introduction

The World is facing the challenge of delivering high-quality healthcare at affordable cost while the greying population continues to grow at an increasing pace. According to the recent studies, the proportion of the population over 65 is expected to almost double from 16.4% in 2004 to 29.9% in 2050 in Europe [1]. Owing to aging population, chronic diseases and their management costs are also on the rise. The current healthcare delivery model is far from ideal to address the challenges ahead [2]. On the other hand, Information Technology combined with recent advances in networking, mobile communications and wireless medical sensor technologies offers great potential to support healthcare professionals and to deliver remote healthcare services, hence providing the opportunities to improve efficiency and quality and better access to care at the point of need.

In this paper, we will present SAPHIRE architecture that provides an intelligent healthcare monitoring platform. To be able to assist medical practitioners efficiently, the healthcare monitoring platform is enriched with a clinical decision support system (CDSS) based on computerised clinical guidelines. Through the SAPHIRE system, the intelligent monitoring architecture is able to access seamlessly the medical history of a patient stored in medical information systems as well as the vital signs of the patients through wireless medical sensors. In this way, not only the observations received from wireless medical sensors but also the patient medical history is used in the reasoning process.

doi:10.1049/iet-com:20060699

Paper first received 26th December 2006 and in revised form 23rd April 2007

O. Nee, A. Hein, T. Gorath and N. Hülsmann are with OFFIS, Institute for Information Technology, Oldenburg, Germany

G.B. Laleci, M. Yuksel, M. Olduz, I. Tasyurt, U. Orhan and A. Dogac are with Software Research and Development Center, Middle East Technical University (METU), Ankara 06531, Turkey

A. Fruntelata and S. Ghiorghe are with The Internal Medicine and Cardiology Department of the Emergency Hospital of Bucharest, Bucharest, Romania

Ralf Ludwig is with Schüchtermann Klinik, Bad Rothenfelde, Germany

E-mail: andreas.hein@offis.de

In the following we will present the basic features of the SAPHIRE architecture by emphasising how it is extending the state-of-the-art healthcare monitoring research:

- The healthcare monitoring system needs to access the vital signs of the patients through wireless medical sensor devices. The SAPHIRE architecture provides a Sensor and Data Point abstraction in order to present seamlessly the sensor data to the applications exploiting it. Fig. 1 shows the data layers used in the SAPHIRE system. The bottom layer represents the actual sensor hardware. Above that, the networking layer is comprised of a Bluetooth stack and a TCP/IP stacks. The sensor driver layer implements the communication protocol that determines the sensor's data structure and how it is transmitted through the network layer. If a new sensor is introduced to the system, the data point abstraction layer ensures that only the proprietary sensor driver needs to be adapted. A virtual device, as it can be seen in the layer above the datapoint abstraction, can – but does not necessarily have to – correspond to a physical device. A virtual device can also receive its data from an algorithm that derives data from other (possibly also virtual) devices. In the SAPHIRE context, an example for a virtual device without a hardware representation is the virtual device for the respiratory rate, where the respiratory rate is derived from the signals of a multi-lead ECG device [3]. Data from the virtual devices are stored in the database (where they are exposed as semantically enriched web service so that the CDSS can use it and where the alarm system can access it), exported as file (for the sensor data analysis software such as Cardionics). Also, the data can be published using the publish/subscribe mechanism of the Java message system (JMS) [4]. The real-time viewer (RTViewer) for Sensor data subscribes to the sensor data topics through JMS to display the data on the physicians' display.
- The healthcare monitoring system needs to access the electronic health records of the patient so that the vital signs of the patient can be put into context while giving recommendations/decisions. However patients' healthcare records are usually physically dispersed in disparate

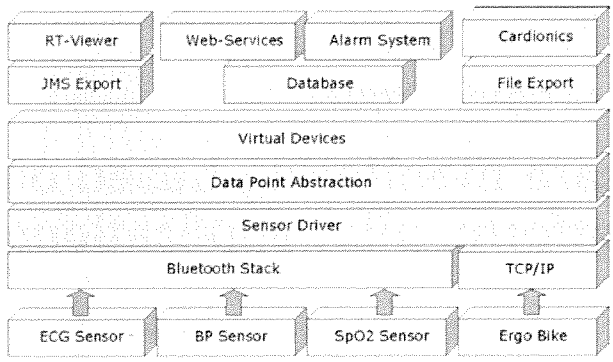


Fig. 1 *Sensors and data point abstraction*

medical institutions which usually do not interoperate with each other. SAPHIRE architecture uses the IHE cross-enterprise document sharing (XDS) [5] architecture to tackle this interoperability problem. In this architecture, the EHR documents are stored in local EHR repositories; however, they are also registered to an XDS registry along with a set of metadata so that the documents then can be discovered and accessed wherever they are actually stored. Through this architecture, we have solved the discovery of and access to relevant EHR documents; however, in order the CDSS to exploit the data stored in these EHR documents, they need to be represented in a machine processable format. In SAPHIRE, we use HL7 CDA level three documents [6] where both the entries and the sections are annotated through coding schemes.

- In SAPHIRE, the behaviour of the CDSS is defined through clinical guidelines. Clinical guidelines are the definitions of medical plans for the study of medical problems and regimens for therapy. As the clinical guideline representation formalism we use the GLIF language [7]. Computerised clinical guidelines have been previously used for implementing CDSSs such as GLEE [8]; however, it has been a well accepted fact that wide adaptation of computerised clinical practice guidelines has yet to be achieved even in a single healthcare institution [9]. One of the reasons for this is complexity of fully integrated decision support systems because of the nature of heterogeneous set of clinical applications need to be involved in the decision process and the lack of commonly agreed electronic healthcare standards and set of interpretable interfaces to proprietary medical information system. The previous approaches towards automating guidelines have addressed this problem through local adaptation where the interfaces of the guidelines to EHRs and medical applications are usually manually bound [8, 10]. To tackle this problem we have extended the GLIF model semantically as detailed in [11]. In this extension, we have annotated the external interfaces of the guideline definition such as references to EHR, vital signs or clinical applications. In addition to this, as discussed in Section 2, we have provided a semi-automatic deployment architecture to process this extended guideline definition and to find dynamically the resources needed in the guideline execution. In this way, the manual deployment effort needed to create interfaces with the underlying EHR systems and clinical workflows is avoided.

A more detailed discussion of SAPHIRE architecture can be found in [11]. The SAPHIRE has two pilot applications: in the hospital pilot, we address the bedside monitoring of subacute phase of the patients suffering from myocardial infarction; in the homecare scenario, we address the homecare monitoring of the rehabilitation of the cardiovascular patients undergone a revascularisation therapy. In the following sections, we will present how SAPHIRE architecture is exploited to model and automate these pilot applications.

2 SAPHIRE hospital pilot application

One of the pilot applications of SAPHIRE project is the hospital pilot application which is being developed for the Emergency Hospital of Bucharest (SCUB). This pilot application aims at demonstrating that the SAPHIRE system can provide bedside intelligent monitoring of patients with subacute coronary syndromes in a wireless fashion to provide computer-generated clinical decision in accordance with the latest European Cardiology Guidelines. In this pilot application, the guideline execution environment will provide continuous feedback to the physicians that will be patient-specific and guideline-oriented, to provide optimized medical care in accordance with medical standards.

In the following sections we will introduce the steps of the guideline used in our pilot, and explain how its semantics have been modelled so that it can be deployed and executed automatically through the SAPHIRE architecture.

2.1 Guideline definition and requirements

In our pilot application, we are using the 'Management of Acute Myocardial Infarction' guideline defined by the European Society of Cardiology [12]. It should be noted that the patients with acute myocardial infarction on admission is not our target population. We are addressing patients who are in subacute phase who can still have acute Myocardial Infarction during hospital stay. For these patients, the use of the system is safe and rapid enough.

We have modelled the guideline in the extended GLIF model we proposed in SAPHIRE [13] using the Protégé Tool [14]. While the guideline is represented in Computer Interpretable Language, the physicians and the medical informatics professionals should collaborate. Although Protégé tool is quite user-friendly being developed by Stanford Medical Informatics department, especially in the definition of expression scripts, the expertise of medical informatics professionals is required.

The guideline execution is triggered by a sensor alarm indicating persistent ST elevation and/or by a new LBBB (left bundle branch block). AST-segment elevation is usually associated with a looming infarction but can also be due to pericarditis or variant angina, whereas the LBBB usually indicates widespread cardiac disease; the alert can be derived directly from the ECG. When the left bundle is blocked, activation of the left ventricle proceeds through the muscle tissue, resulting in a wide (0.12 msec) QRS complex.)' coming from the ECG sensor. This initiates the 'First Line Medication' step, where depending on the current medication and medical history coming from electronic healthcare records (EHR), and also the vital signs coming from sensor devices, medications and therapies are proposed by the guideline. After the first-line medication (such as giving aspirin, applying nitroglycerin therapy), based on the vital signs coming from sensors and patient's medical history, either another guideline for acute heart failure is followed or a fibrinolysis therapy (Fibrinolytic drugs are given after a heart attack to dissolve the thrombus blocking the coronary artery, experimentally in stroke to reperfuse the affected part of the brain, and in massive pulmonary embolism.) or an appropriate 'invasive reperfusion therapy' is applied to the patient. A more

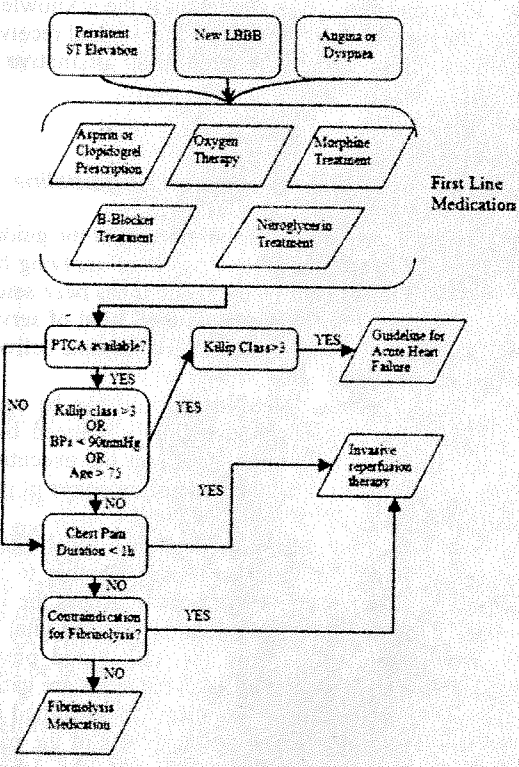


Fig. 2 *Management of acute myocardial infarction guideline*

detailed flow of the guideline is presented in Fig. 2. The parallelograms indicate sub-flows that are not detailed in the figure.

To model the guideline definition in the extended GLIF, we have identified the external resources that we need to interact with while the guideline is being executed. These requirements and how they have been realised can be exemplified as follows:

• The guideline execution needs the vital signs of the patient: for example the oxygen saturation of patient's blood is needed to decide whether to apply oxygen therapy or not. As shown in Fig. 3*a*, the sensor data coming from wireless sensors to the gateway computer are exposed as Web services. These Web Services have been published to the UDDI registry [15] together with their functionality semantics. For semantic annotation, we use the nomenclature codes defined by the IEEE 11073-10101 standard [16]. In the guideline definition, whenever a vital sign of the patient is needed, the nomenclature code from IEEE 11073-10101 is assigned as the semantic category of the data needed. When the guideline is to be deployed, first of all the relevant Web services are discovered from the UDDI registry through their semantics, and saved to the guideline definition to be used in the guideline execution.

• The guideline execution requires data usually stored in the EHR of the patient, such as a possible contraindication to fibrinolysis. In SAPHIRE, the EHRs of the patient are represented as HL7 CDA level three documents [6]. An example CDA document used in SAPHIRE is presented in [17]. As shown in Fig. 4, these EHR documents are shared through IHE XDS registry/repository architecture [5], where the EHR documents are registered/queried to/from the XDS registries through their metadata. In the guideline definition whenever an EHR document is needed, the document metadata is specified through LOINC document type codes [18]. When the guideline is to be deployed, the references to the related EHR documents of the patient stored in XDS repositories are discovered from the XDS registry through this metadata.

• While guideline is being executed, it needs to interact with the underlying clinical workflow. For example, in our guideline, the guideline execution orders a coronary angiography to derive functional parameters of the patient's heart and blood supply, and needs to receive the results to continue its operation. Such interactions are facilitated through the medical web services exposed by the hospital information systems (HIS). In our pilot study, the coronary angiography order is implemented as an asynchronous Web service since gathering the result may take some time but the execution can go on with the rest of the guideline, while operations such as saving the medication recommendations back to HIS are implemented as synchronous Web services. Such Web services are also published to the UDDI registries through their functionality semantics. We are using an HL7-based service functionality ontology [19]; such service requirements are represented through the nodes of this ontology in the guideline definition as presented in Fig. 3*b*. When the guideline is to be deployed, first of all these Web services are discovered from the UDDI registry through this metadata, and started to be used in the guideline execution.

• During the execution of guideline, several alarms and reminders may need to be issued to medical practitioners,

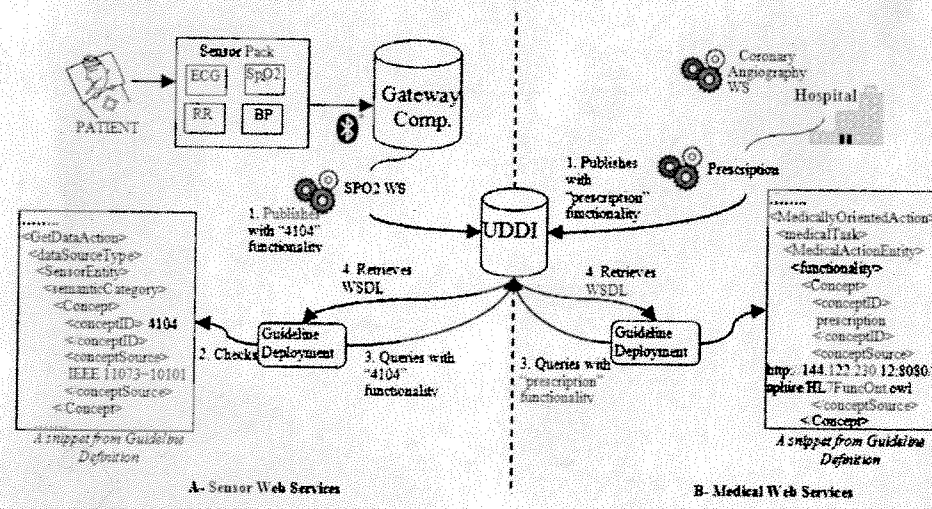


Fig. 3 *Sensor and medical web services architecture*

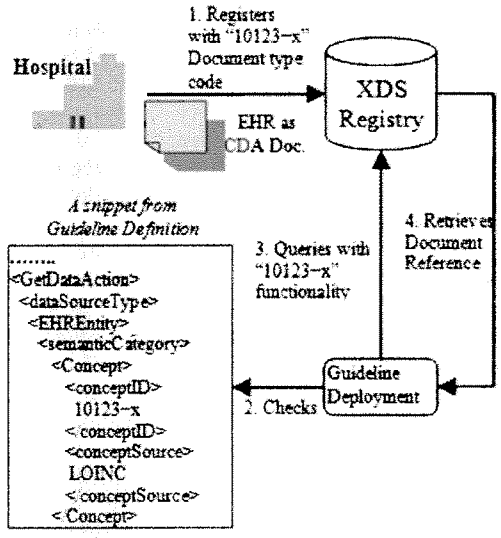


Fig. 4 *SAPHIRE EHR architecture*

and when necessary to the patient's relatives. In such cases, the alarm message and the role to whom the message should be delivered are sent to an agent, the Alarm Distribution Agent, which is specifically designated to distribute these messages to the necessary recipients in the most efficient and reliable way. According to the urgency of the message to be delivered, one or more of the supported messaging means namely email, SMS and instant messaging (MSN, GTalk) are selected by the agent. The alarm distribution mechanism is reliable, that is the acknowledgements of the messages are tracked, if the original receiver is not available, the message is routed to an alternative user specified previously.

2.2 Execution of a small portion of guideline

In this section, execution of a small portion of the guideline is explained. This portion is responsible of applying nitroglycerin (NTG) treatment to patient and it has been selected since it is rich enough to demonstrate all kind of services, namely the EHR, sensor, medical services and delivering alarm to the clinician.

The flowchart of this guideline portion can be seen in Fig. 5. This flowchart is automatically generated by the SAPHIRE guideline monitoring tool and the execution is monitored on it by changing the underlying colors of guideline steps.

The first step is First Line Medication Branch Step. In order the guideline execution to decide whether to apply NTG or B-blocker treatment, it needs to access the EHRs (Baseline Bloodpressure, AnginaStatus) and the vital signs of the patient through sensors (systolic blood pressure, heart rate). In step 2, first of all the vital signs are gathered through invoking the Sensor Web Services identified in the deployment phase. For accessing the EHRs of the patient, the EHR documents are retrieved through the document references located from the XDS registry in the deployment phase. The necessary pieces are extracted from the retrieved HL7 CDA documents to be used in the guideline execution.

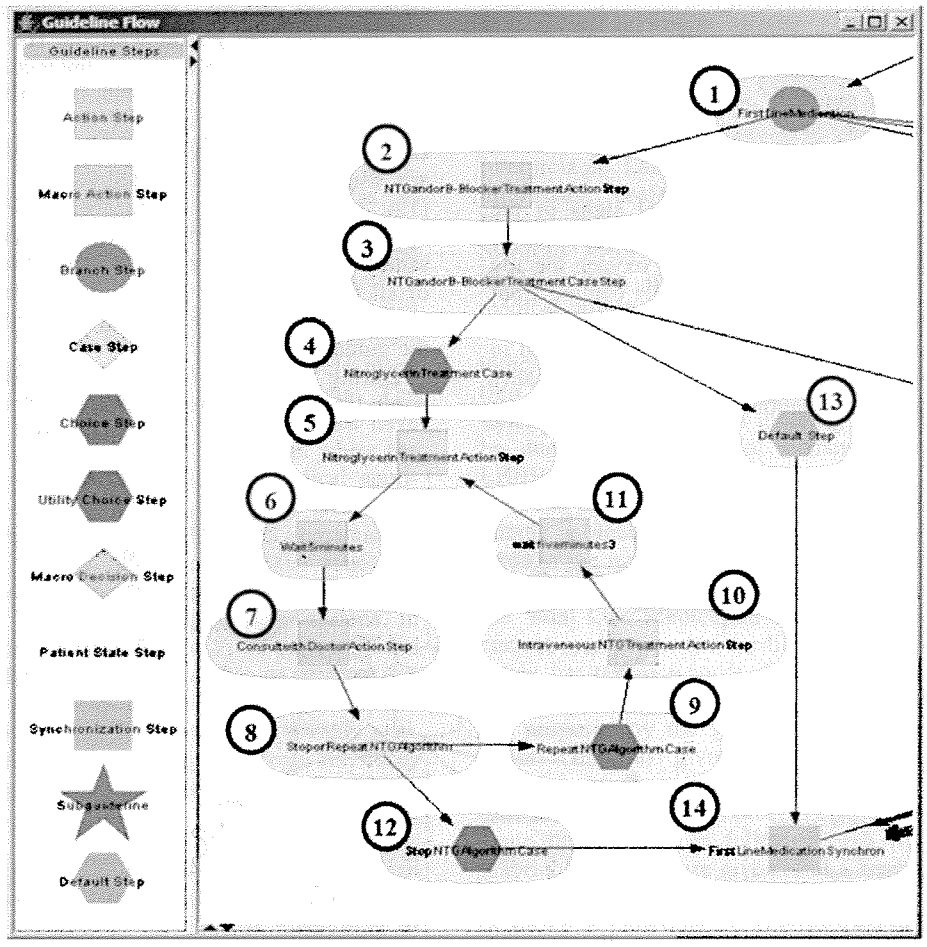


Fig. 5 *Flowchart of a small portion of the guideline*

Service Type	Sensor Web Service
Input	<Data> <Patient_Data> <Id>10000006</Id> </Patient_Data> </Data>
Output	<Data> <Observation> <Id_OB>HeartRate</Id_OB> <Index_Value> <Index_IV>85.0</Index_IV> </Index_Value> </Observation> </Data>
Description	Value of "HeartRate" variable is retrieved from the sensor.

Fig. 6 *Retrieval of heart rate*

Example input, output structures for these Web Service calls and EHR accesses are represented in Figs. 6 and 7.

After collecting the necessary input, in step 4, the criterion presented in Fig. 8 is checked to decide whether NTG treatment is applicable to the patient.

In SAPHIRE, the decision criteria of the decision options should be defined with JavaScript as in the format shown in Fig. 8. The JavaScript executor module has been implemented using Rhino library [20]. With the retrieved values, this script evaluates to *true* and execution continues with Nitroglycerin Treatment Action Step (5). In this step, the doctor of the patient is informed that it is urgent to provide 0.4 mg NTG to the patient. This message is delivered to the doctor via SMS through the SAPHIRE alarm distribution mechanism as presented in Fig. 9. On the basis of this message, the doctor can check the guideline execution monitoring tool to see why such a decision is taken. In addition to this, this medication recommendation is stored to the HIS, through a Web Service call as presented in Fig. 10.

After this step, through similar mechanisms presented the status of the patient is monitored and it is decided to stop the NTG treatment algorithm, and continue with the First Line Medication Synchronization Step (14).

As it can be seen, our guideline execution seamlessly accesses the necessary patient data and the underlying clinical workflow through widely accepted standards, Web Services, IHE XDS architecture and HL7 CDA.

3 Homecare application

In the homecare scenario, the SAPHIRE platform is used to implement services that benefit patients at a later state of their treatment. The hospital scenario deals with the patient in the subacute phase of myocardial infarction, while the homecare scenario deals with physiological recovery and training in secondary prevention [21]. Another difference between the two scenarios is the patient's role. After being discharged from the (rehabilitation) clinic, the patient must assume an active role in the treatment, and the SAPHIRE system empowers the patient to do so. It allows the patient to advance from simply being the recipient of care to being an important partner in the endeavour of tackling the disease. The daily feedback can be mutual, that is the patient can easily inform the doctor about problems that require attention but are not deemed severe enough to warrant an immediate visit. Also the doctor can involve the patient more directly by giving feedback to the patient, and informing them about the progress the patient is making.

The key component of the homecare scenario is a modified bike ergometer that is shown in Fig. 11. A panel PC has been mounted to the bike and serves both as a gateway for the sensor network and as a user interface for the patient. Adding this component to the SAPHIRE system allows the patient to emulate a supervised training at home. Being able to perform training with medically sound

Service Type	EHR
Input	10000006 (The reference to the EHR Document located in the deployment phase)
Output	<Data> <Observation> <Certainity_OB>0</Certainity_OB> <Mood_Cd_OB> <Symbol_Value_S>EVN</Symbol_Value_S> </Mood_Cd_OB> <Service_Cd_OB> <Concept_Id_C>C0428860</Concept_Id_C> <Concept_Name_C> SystolicBloodPressure </Concept_Name_C> <Concept_Source_C>UMLS</Concept_Source_C> </Service_Cd_OB> <Severity_OB>0</Severity_OB> <Text_Value>100</Text_Value> </Observation> </Data>
Description	Value of "BaselineBP" variable is extracted from the HL7 CDA document retrieved from the XDS Repository.

Fig. 7 *Retrieval of baseline blood pressure*

JavaScript Code	function NitroglycerinTreatmentCriterion(SystolicBP, BaselineBP, HeartRate, AnginaStatus) {
	if (!((SystolicBP.getValue().getIndex() < 90) ||
	(SystolicBP < (BaselineBP.getValue().getIndex() - 30)) ||
	(HeartRate.getValue().getIndex() < 60)) &&
	(AnginaStatus.getValue().getText() == "true") &&
	(HeartRate.getValue().getIndex() <= 100))
	return true;
	else
	return false;
	}

Description	This criterion has 3 main blocks of conditions: if(not(E1) and E2 and E3). In E1, it is checked whether *Systolic Blood Pressure (BP)* of the patient is less than 90 mmHg, or *Systolic BP* is less than *Baseline BP* minus 30 mmHg or *Heart Rate* is less than 60 bpm. In E2, it is checked whether the patient has *Angina* and in E3, it is checked whether *Heart Rate* is less than or equal to 100 bpm.

Fig. 8 *The evaluator script of nitroglycerin treatment criterion*

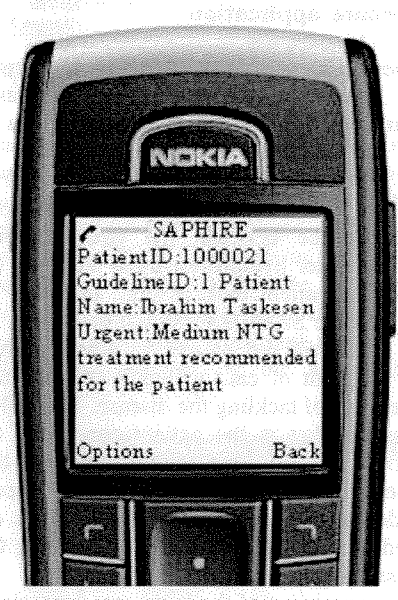


Fig. 9 *NTG treatment alarm message*

guidelines and automatic supervision in effect without having to travel to the rehabilitation clinic is already an improvement compared with the current state of the art, where supervision after discharge is sketchy at best and patient compliance is relatively low. Patients living in rural areas without quick access to the resources of the rehabilitation clinic will be the ones benefiting the most from the SAPHIRE system.

Although there are no suitable clinical guidelines for the fitness training, it is considered as one of the most important building blocks of a successful treatment of patients who have undergone a coronary angioplasty (PTCA). Assisted home-based training, as it is envisioned for the homecare application, will generally be composed of two consecutive phases, the inpatient phase and the outpatient phase.

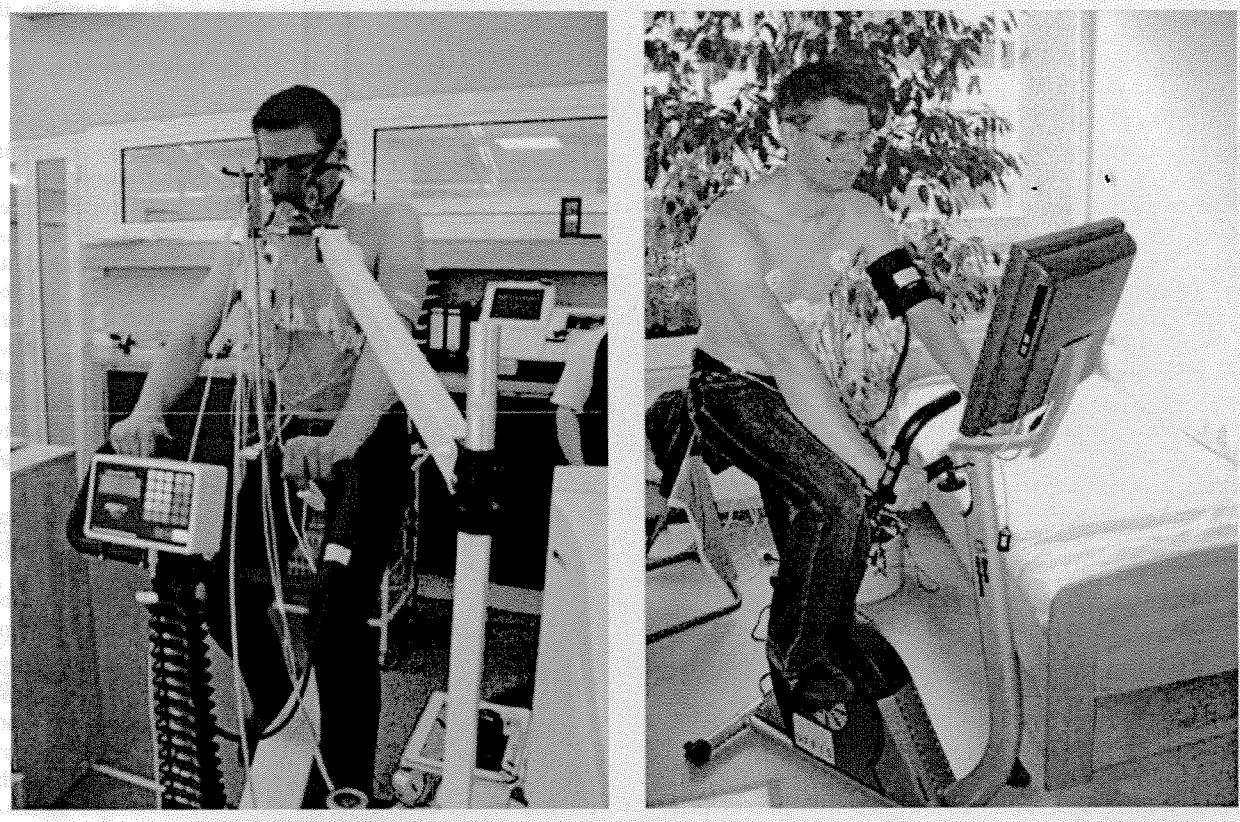
In the inpatient phase, after a successful revascularisation procedure, patients will train in the rehabilitation clinic using the homecare equipment under the supervision of a physician. During this phase, the patient will become acquainted with the equipment (ergometer bike and sensors, Fig. 11*a*). In this phase, which lasts ~3 weeks, the parameters (training programme) and reference data for the home-based training will be acquired using spiroergometry.

In the outpatient phase, the patient will proceed with the training at home. The results of each training session are summarised in a report that will be reviewed by a physician once per day. If there is an event requiring immediate assistance, the physician will be notified at once.

Most of the software used in the homecare application is implemented as bundles for the OSGi framework [22]. As Fig. 12 illustrates, the homecare application consists of several components with several dependencies among the components. The OSGi framework allows an easy

Service Type	Medical Web Service
Input	<storeMedication> <medication> <application> <route>sublingual</route> </application> <dosageQuantity>0.4</dosageQuantity> <notes></notes> <person> <PID>10000006</PID> </person> <service> <conceptID>C0017887</conceptID> <conceptName>Nitroglycerin</conceptName> <conceptSource>UMLS</conceptSource> </service> </medication> </storeMedication>
Output	"Medication inserted successfully"
Description	0.4mg Nitroglycerin medication recommendation is saved to HIS

Fig. 10 *Prescription of 0.4 mg nitroglycerin to the patient*



a b

Fig. 11 *Bike ergometer training*
a Inpatient phase at the clinic
b Outpatient phase at home (modified form)

management of these highly interconnected components. As some components such as the training programme and the patient GUI are very likely to require updates, the ability to update bundles remotely will make it easier to offer maintenance for systems deployed at the patient's homes.

As in the hospital pilot application, homecare scenario employs several aspects and variants of decision support to facilitate a safe and efficient training session.

3.1 Sensor placement support

The correct placement of the sensors is vital for the functioning of the SAPHIRE system. The hospital application can rely on nurses and doctors being able to attach the sensors correctly on a patient, but in the homecare scenario, no such professional help is available, leaving this important task to the patient. Although the patient has been instructed during the inpatient phase in the rehabilitation clinic, it is likely that the system will have to instruct the patient on how to attach the sensors, and to detect missing or incorrectly positioned sensors. Missing ECG electrodes, for example, are detected easily, but in order to detect switched or misplaced leads, reference ECGs have to be used for comparison.

3.2 Patient status and stabilisation

Before the actual exercise begins, the patient's vital parameters are acquired from the sensors, and the patient has to answer a number of questions. These questions cover the patient's current status (i.e. 'Do you have breathing difficulties?') but also the patient's experience after the last

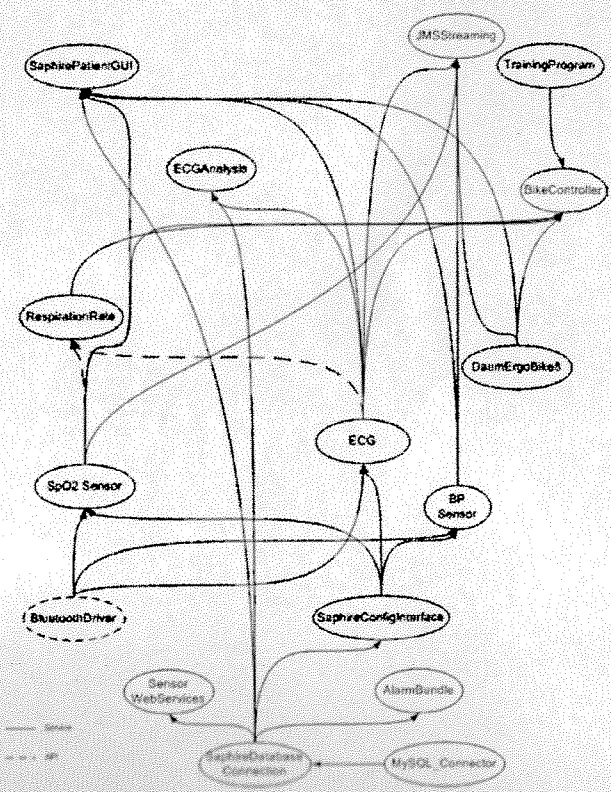


Fig. 12 *SAPHIRE OSGi bundles*

training session (i.e. 'Did you experience chest pain after the exercise ended?').

On the basis of the patient's answers, and taking the sensor data into account, the system determines whether or not it is safe for the patient to commence the ergometer training. Several options are available as a result:

- The patient's vital parameters are only temporarily outside the 'safe corridor', without being critical. If this is the case, the system will instruct the patient to wait for a few minutes. During this, the acquisition of data by the sensors and monitoring will continue.
- The patient's vital signs are within the 'safe corridor', but the answers given indicate that the patient should not continue the training session. An example for this case would be a patient experiencing pain after the last training session that faded and is not felt in rest. The system will instruct the patient to contact the rehabilitation clinic.
- The patient's vital parameters are critical. In this case, the patient is asked to contact the rehabilitation clinic right away. The SAPHIRE alert system allows an alert to be sent to the clinic, or to an emergency dispatcher.
- The patient's vital parameters are within the 'safe corridor', but the patient indicates that the medication has been changed. In this case, the training might commence, using a more conservative guideline. Information regarding the medication would be sent to the rehabilitation clinic, so it can be taken into account for a new training programme.

Once the patient begins the exercise, leaving the state of rest, monitoring the status becomes even more crucial. During the training, the patient can signal events (such as pain and dyspnoea) at any time using the touch screen. This action will abort the training and trigger an alert that will be sent to the clinic. Sensor data are used to determine the patient's status during the exercise session. Depending on the patient's recommended training programme, the ergometer bike needs to control the resistance in order to maintain target wattage or a target heart rate. Adapting the resistance to maintain a set heart rate is a feature offered by most ergometers. We believe that taking into account the patient's breathing frequency, blood pressure, the oxygen saturation and the ECG – in addition to the heart rate – allow for a much safer training, creating a combination of fitness training and ongoing diagnostics.

Fig. 13 shows a simplified state machine that describes the states of an early experimental system during a training session where the ergometer's power is regulated by a PI (proportional–integral) controller to maintain a target

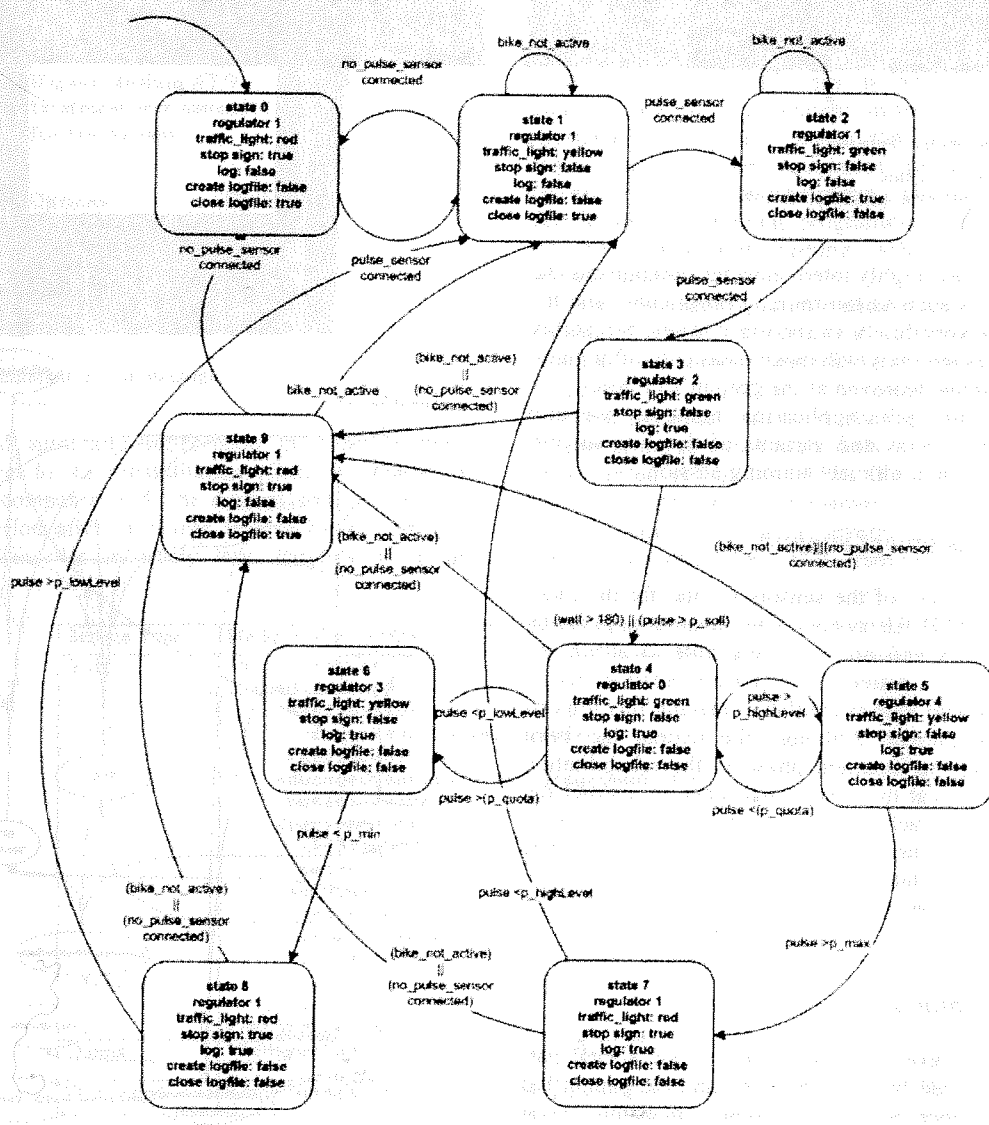


Fig. 13 *State machine describing the training*

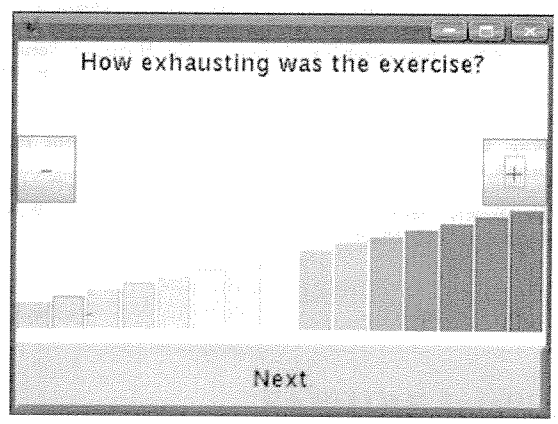


Fig. 14 *Example dialog querying the Borg value*

pulse. In the homecare application, more sensors will be used, but the pulse will remain as the main variable used to control the resistance.

After the training session, while the patient is cooling down and the sensors are still acquiring blood pressure and other data, several questions are asked to determine how the patient is feeling. Although sensor data reveal a lot about how the patient was able to cope with the training programme, having this additional input is very helpful. Among other things, the system asks the patient about the perceived exertion. Fig. 14 maps the degree of exertion to the Borg scale [23]. This perceived exertion ranges between 'very, very light', which is usually the case in rest, to 'exhaustion'. The American College of Sports Medicine (ACSM) has recommended rate of perceived exertion (RPE) since 1986 for both fitness and cardiac rehabilitation purposes [24]. Guidelines of the ACSM recommend a RPE range of 12 ('fairly light') to 16 ('hard') as the perceived exertion range associated with a cardiovascular training effect; roughly corresponding to 60–85% of the maximal heart rate ($HR_{max} = 220 - age$).

Capturing the sensor data and correlating it with the patient's answers (especially the RPE) allow the physicians to determine the efficiency of the training.

For the pilot application, patients after an acute myocardial infarction have been chosen. However, the system can be used for other patient groups as well. For example, by providing scales, the system would be suitable for patients with chronic heart failure, who would benefit greatly from light exercise and a thorough monitoring of the weight. It would also be possible to turn the SAPHIRE system into a tool for prevention. A setup consisting of an ergometer, weight scales and a glucose meter could be used to manage diabetes while aiding the necessary change of lifestyle and combating risk factors for heart infarctions. The feedback that can be provided through a system like SAPHIRE makes it a powerful tool for people who wish to pursue actively the goal of personal health.

4 Conclusions

The SAPHIRE system proposes an intelligent healthcare monitoring architecture based on clinical guidelines. The architecture is supported with a semantic interoperability platform in order to access the Electronic Healthcare Records and vital signs of the patients seamlessly, the absence of which was presented as a major reason for the failure of adoption of clinical guidelines by the hospitals. The system is tested with two pilot applications, one in homecare, one in hospital environment. Through these pilot applications, the system aims not only to reduce the workload in the hospital and to diminish probability of human error, but also in future aims to reduce the medical costs by cutting down intensive care days and complex diagnostic investigations or therapeutic approaches where these are not necessary and supporting remote rehabilitation at homes. The SAPHIRE system in the homecare scenario can be seen as one step towards p-Health (personal health) applications as well as an extension of healthcare services to the patient's home. With some modifications of the sensor setup and new computer-executable guidelines, the system can be adapted for other diseases easily.

5 Acknowledgments

This work is supported by the European Commission through IST-27074 SAPHIRE project and in part by the Scientific and Technical Research Council of Turkey, Project No: EEEAG 105E133.

6 References

1 Eurostat news release 48\2005. Available at: http://epp.eurostat.cec.eu.int/pls/portal/docs/PAGE/PGP_PRD_CAT_PREREL/PGE_CAT_PREREL_YEAR_2005/PGE_CAT_PREREL_YEAR_2005_MONTH_04/3_08 042005-EN-AP.PDF, accessed on 8 April 2005

2 PHS Consultation Workshop Final report. Available at: http:\\ec.europa.eu\information_society\activities\health\docs\events\phealth2006\fp7phs _consultation_workshop_final-report.pdf

3 Moody, G.B., Mark, R.G., Zoccola, A., and Mantero, S.: 'Derivation of respiratory signals from multi-lead ECGs', *Comput. Cardiol.*, 1985, **12**, pp. 113–116

4 Java Message Service (JMS). Available at: http://java.sun.com/products/jms/

5 IHE IT Infrastructure Technical Framework. Available at: http://www.ihe.net/Technical_Framework/upload/ihe_iti_tf_2.0_vol1_FT_2005-08-15.pdf

6 HL7 Clinical Document Architecture, Release 2.0. Available at: http://hl7.org/library/Committees/structure/CDA.ReleaseTwo.CommitteeBallot03.Aug.2004.zip

7 Boxwala, A.A., Peleg, M., Tu, S., *et al.*: 'GLIF3: a representation format for sharable computer-interpretable clinical practice guidelines', *J. Biomed. Inform.*, 2004, **37**, (3), pp. 147–161.

8 Wang, D., Peleg, M., Tu, S.W., Boxwala, A., Ogunyemi, O., Zeng, Q., Greenes, A.R., Patel, V.L., and Shortliffe, E.H.: 'Design and implementation of GLIF3 guideline execution engine', *J. Biomed. Inform.*, 2004, **37**, pp. 305–318

9 Entwiste, M., and Shiffman, R.N.: 'Turning guidelines into practice: making it happen with standards – Part I', *Healthcare Inform. Rev.*, 2005

10 Seyfang, A., Miksch, S., and Marcos, M.: 'Combining diagnosis and treatment using Asbru', *Int. J. Med. Inform.*, 2002, **68**, (1–3), pp. 49–57

11 Laleci, G., Dogac, A.: 'A semantically enriched clinical guideline model enabling deployment in exterogeneous health care environments', submitted to *Inf. Syst.*

12 Van de Werf, F., Ardissino, D., Betriu, A., Cokkinos, D.V., Falk, E., Fox, K.A.A., Julian, D., Lengyel, M., Neumann, F.J., Ruzyllo, W., Thygesen, C., Richard Underwood, S., Vahanian, A., Verheugt, F.W.A., and Wijns, W.: 'The Task Force on the Management of Acute Myocardial Infarction of the European Society of Cardiology: Management of acute myocardial infarction in patients presenting with ST-segment elevation', *Eur. Heart J.*, 2003, **24**, pp. 28–66

13 Extended GLIF model of the guideline. Available at: http://144.122.230.15/gl/scubGL1.owl

14 Gennari, J., Musen, M.A., Fergerson, R.W., Grosso, W.E., Crubezy, M., Eriksson, H., Noy, N.F., and Tu, S.W.: 'The evolution of protégé: an environment for knowledge-based systems development', *Int. J. Human-Comput. Stud.*, 2003, **58**, (1), pp. 89–123

15 Dogac, A., Cingil, I., Laleci, G., and Kabak, Y.: 'Improving the functionality of UDDI registries through web service semantics'. 3rd VLDB Workshop on Technologies for E-Services (TES-02), Hong Kong, China, 23–24 August 2002

16 ISO/IEEE 11073_10101:2004. Health informatics – Point-of-care medical device communication – Part 10101: Nomenclature, available

at: http://www.iso.org/iso/en/CatalogueDetailPage.CatalogueDetail?CSNUMBER=37890&scopelist=CATALOGUE
17 An Example CDA document used in SAPHIRE. Available at: http://144.122.230.12:8080/saphire/cda_documents/SCUBGL1CDA.xml
18 Logical observation identifiers names and codes LOINC. Available at: http://www.regenstrief.org/loinc/
19 Dogac, A., Laleci, G., Kirbas, S., Kabak, Y., Sinir, S., Yildiz, A., and Gurcan, Y.: 'Artemis: deploying semantically enriched web services in the healthcare domain information systems journal (Elsevier)', 2006, 31, (4–5), pp. 321–339
20 Rhino: JavaScript for Java. Available at: http://www.mozilla.org/rhino/
21 Hein, A., Nee, O., Willemsen, D., Scheffold, T., Dogac, A., and Laleci, G.B.: 'SAPHIRE – Intelligent healthcare monitoring based on semantic interoperability platform – The Homecare Scenario' 1st European Conf. on eHealth (ECEH'06), 12–13 October, 2006, Fribourg, Switzerland, pp. 191–202
22 OSGi Alliance. Available at: http://www.osgi.org
23 Borg, G.: 'Perceived exertion as an indicator of somatic stress', *Scand. J. Rehabilit. Med.*, 1970, **2**, pp. 92–98
24 Franklin, B., (ed.): 'American College of Sports Medicine Guidelines for exercise testing and prescription' (Williams and Wilkins, Baltimore, 2000, 6th edn.)

SAPHIRE-Intelligent Healthcare Monitoring Based on Semantic Interoperability Platform

A. Dogac[1], G. Laleci[1], M. Olduz[1], I. Tasyurt[1], M. Yuksel[1], A. Okcan[1]
[1] Software Research & Development Center, Middle East Technical University, asuman@srdc.metu.edu.tr, Turkey

Abstract: **The SAPHIRE project has developed an intelligent healthcare monitoring and decision support system to address the problem of an ever-increasing workload in medical fields due to the increasing percentage of elderly people in Europe's population. In the SAPHIRE project, the patient monitoring is achieved by using agent technology where the "agent behaviour" is supported by intelligent decision support systems based on clinical practice guidelines. In SAPHIRE system, patient history stored in medical information systems is accessed through an Integrating the Healthcare Enterprise XDS (Cross Enterprise Document Sharing) based architecture to tackle the interoperability problem. In this way, not only the observations received from wireless medical sensors but also the patient medical history is used in the reasoning process.**

Introduction

Clinical Decision Support Systems (CDSS) broadly refer to providing clinicians or patients intelligently filtered and processed clinical knowledge and patient-related information in order to enhance patient care. Recently, there has been an explosion in basic and clinical research on disease pathophysiology and treatment. Coupled with increased demands on healthcare delivery systems, this rapid growth has made the practice of medicine increasingly complex. The response of healthcare community to this growing complexity has been to develop clinical practice guidelines to simplify and improve healthcare delivery. Despite the widespread publication of clinical standards and practice guidelines, however, healthcare professionals have difficulties in understanding and applying these guidelines in the clinical care setting. This necessitates computerized decision support systems automating clinical guidelines to support the health professionals. One of the major challenges in developing computerized decision support systems executing these guidelines is accessing the many disparate data sources needed to retrieve patient-specific information. The SAPHIRE Project [1] has made use of semantically enriched Web service architecture and semantic mediation mechanisms to tackle this problem.

The SAPHIRE system aims to continuously monitor the patients through dedicated agents and to support the healthcare professionals through intelligent decision support system. Additionally, it generates and sends notifications and alerts to the related people.

System Architecture

When used at the point of clinical care, automated, computer-based CDSSs can improve healthcare professional compliance with specific treatment guidelines. However, the challenge is in providing timely and complete access to the many disparate data sources to retrieve patient-specific information. In the SAPHIRE system, patient vital signs are received through Web services from the sensor devices from the Gateway computer, and patient history stored in medical information systems is accessed through Integrating the Healthcare Enterprise (IHE) [2] Cross Enterprise Document Sharing (XDS) Registry/Repository to tackle the interoperability problem. In this way, not only the observations received from biosensors but also the patient medical history is used in the reasoning process in the clinical decision support system. This is an essential component, because in clinical guidelines, the physiological signs received from wireless medical sensors, the patient care plan and the medical history retrieved from Electronic Healthcare Records (EHRs) (such as previous diagnosis, medication list, allergy/adverse drug reactions) all affect the clinical path to be followed.

In Figure 1, the overall architecture of SAPHIRE system is illustrated. The SAPHIRE system is based on an agent based network which has been implemented as Java Agent Development Framework (JADE) [3] Agents. Each main module of the SAPHIRE is modelled as an agent which handles and abstracts the communication of the modules for enabling the flexibility of the whole system. The main modules of the SAPHIRE system are provided briefly in the subsections below. For the details of the architecture please refer to [4]. The medical processes for diagnosis, treatment and observation of the patient, on the other hand, have been modelled as clinical guidelines using Guideline Interchange Format (GLIF) [5] by complying with the latest European Cardiology Guidelines [6].

Agent Factory

Agent Factory is the primary agent which controls all other agents in the system. When the Agent Factory is launched, first it registers itself to the "Main-Container" of JADE then it launches several other agents listed below:

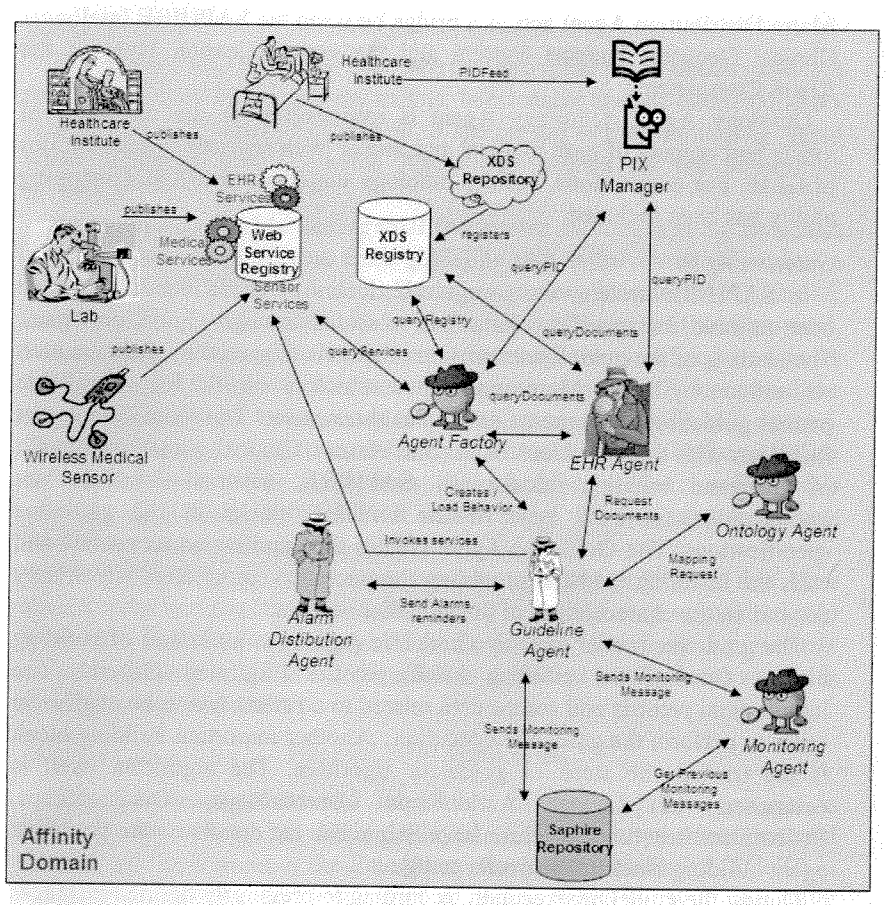


Figure 1 Overall Architecture of the SAPHIRE Initial Prototype

EHR Agent: This is the first agent that Agent Factory Agent creates. EHR Agent is responsible of querying and retrieving the patient electronic healthcare record documents in that clinical affinity domain. There exists one EHR Agent in each affinity domain. In SAPHIRE, EHR Agent queries and retrieves the EHRs in Health Level 7 (HL7) [7] Clinical Document Architecture (CDA) format from IHE XDS Registry/Repository.

Alarm Distribution Agent: This agent is responsible of delivering the alarm entities, which are generated during the execution of the guideline, to the related healthcare people such as doctors, nurses, patient relatives, etc.

Alarm Distribution Agent acts as a bridge between the SAPHIRE Intelligent Clinical Decision Support System and the communication layer of the Alarm Component.

Ontology Agent: Ontology Agent is responsible for the mappings of inputs and outputs of both medical and sensor Web services to data entities of GLIF. For conversions, besides ontology mapping, also XSLT mapping ability has been included.

Guideline Agent

In SAPHIRE system, the computerized clinical guidelines which have been tailored for specific patients are executed through Guideline Agent. The starting of the guideline execution process is triggered by the execution request coming from a Monitoring Agent, which reveals the list of available patient-guideline assignments to the healthcare user. The Guideline Agent functions with the operations of handler classes. Clinical guidelines consist of different building blocks. In SAPHIRE, separate handlers are implemented for each type of the building blocks in the guideline specifications. The Guideline Agent creates the handlers when needed and calls the handling methods providing the necessary parameters which may depend on the characteristic of the guideline section.

The guideline model used by Guideline Agent has a number of building blocks. One of the building blocks is the Eligibility Criteria. The longitudinal records and sensor data related to a patient has to be eligible in order to perform the guideline execution. Another important building block is the tracing the steps of guideline algorithm. The algorithm itself is composed of several ActionSteps, BranchSteps, DecisionSteps, PatientStateSteps and SynchroniationSteps. For the details of the guideline agent building blocks please refer to [8].

During the guideline execution, according to previously defined instances of AlarmMessage class, Guideline Agent creates alarm messages and reminders to be sent to doctors, nurses or healthcare relatives and passes the content of Alarm Messages to Alarm Distribution Agent. Alarm Distribution Agent is responsible of delivering these alarm messages to the communication layer of the Alarm Component in order to be delivered to related people according to rules by means of email, SMS or instant messaging.

Guideline Agent also interacts with Ontology Agent for mapping the input/output structure of Web services while it is accessing the EHR documents, Medical and Sensor Web services. This way, the interoperability is achieved.

Monitoring GUI

The Monitoring GUI is the user interface provided to the doctors and it is responsible from displaying any events, latest status and the detailed history related with the guideline execution. It sits on top of the Monitoring Agent which interacts with Guideline Agent and gathers all the monitoring messages.

When created, the Monitoring GUI first parses and displays the flowchart of guideline in GLIF format. Each guideline step defined in GLIF is drawn with a different shape and colour. During execution, Monitoring Agent user interface also provides a live history of what happened until that moment. The messages are kept in XML format but they are visualized in a more tabular format. Besides data retrieval from sensors and EHR records of the patient, sometimes it is necessary to get information from directly doctors. These information requests are defined with consult actions in the guideline definition which waits input from the practitioner. For handling this property, the Monitoring GUI has a consult panel which is used for getting input from the practitioner.

Using Monitoring GUI, doctors can also monitor guidelines that have already been executed and finished. During execution each completed step is saved to SAPHIRE repository. When the doctors want to see a previously executed guideline, the selected guideline execution is simulated as if it is running at that moment.

Hospital Pilot Application

After the development and integration of the SAPHIRE project is finalized, the system has been deployed to The Internal Medicine and Cardiology Department of the Emergency Hospital of Bucharest (SCUB) in Romania for testing and evaluation of the system [9]. With this pilot application, the SAPHIRE platform will be evaluated using the real data from the real patients having myocardial infarction.

For the prototype, four clinical guidelines have been defined and modelled which are for management of myocardial infarction with non ST-segment elevation and management of myocardial infarction with ST-segment elevation.

In the pilot application, all the patients sign an informed consent admitting to use SAPHIRE system. Then, the sensors developed within the scope of the SAPHIRE are placed on the patient, the necessary configuration is performed and the patient is started to be monitored by the automatic guideline execution. As the guideline execution continues,

appropriate recommendations and prescriptions are suggested and alarms are sent to the related people in the hospital whenever necessary.

Conclusion

The SAPHIRE Project aimed to develop an intelligent healthcare monitoring and Decision Support System (DSS) for reducing the ever-increasing workload in medical fields due to the increasing percentage of chronic diseases. This system which is based on agent framework has been designed to continuously monitor the patients and to deliver recommendations, prescriptions and alarms to the medical personnel as needed. The system also enables the remote monitoring of the patient since the continuous monitoring is achieved with extensive support for timely and complete data retrieval from different sources (from EHRs, sensors and medical people).

At the moment, the SAPHIRE System is being validated by two pilot applications; a hospital application and a homecare application. With the Hospital Pilot Application, it is expected to be demonstrated that the SAPHIRE system can provide bedside intelligent monitoring wirelessly and provide patient-specific clinical decisions in accordance to the European Cardiology Guidelines. The pilot application also tests the safety and the accuracy of the sensor data, the Alerts System, the accuracy of the history data from the EHRs and the recommendations generated by the system by correct interpretation and implementation of the guidelines.

References

[1] The SAPHIRE Project, http://www.srdc.metu.edu.tr/webpage/projects/saphire/

[2] IHE, http://www.ihe.net/

[3] JADE, http://jade.tilab.com/

[4] Gokce Banu Laleci Erturkmen ,Asuman Dogac, Mehmet Olduz, Ibrahim Tasyurt, Mustafa Yuksel, Alper Okcan, SAPHIRE: A Multi-Agent System for Remote Healthcare Monitoring through Computerized Clinical Guidelines, Book Chapter in "Agent Technology and e-Health, Whitestein Series in Software Agent Technologies and Autonomic Computing, -, -, 01 2008, published

[5] GLIF, http://www.glif.org/

[6] European Cardiology Guidelines, http://www.escardio.org/knowledge/guidelines/

[7] HL7, http://www.hl7.org/

[8] The Saphire Project Deliverable 5.1.1 Development of Intelligent Clinical Decision Support Systems, http://www.srdc.metu.edu.tr/webpage/projects/saphire/deliverables/D5.1.1Final.doc

[9] Nee O. ,Hein A., Gorath T., Hulsmann N., Laleci G., Yuksel M., Olduz M., Tasyurt I., Orhan U., Dogac A., Fruntelata A., Ghiorghe S., Ludwig R., SAPHIRE: Intelligent Healthcare Monitoring based on Semantic Interoperability Platform - Pilot Applications, IEE Proceedings Communications-Special Issue on Telemedicine and e-Health Communication Systems, June 2007, Accepted