

İşlevsel Büyüklük Ölçümlerinde Formalizasyon Çalışmaları ve Bir İşlevsel Büyüklük Ölçüm Aracı

Barış Özkan¹

Ali Yıldız²

Onur Demirörs³

^{1,3}Enformatik Enstitüsü, Orta Doğu Teknik Üniversitesi, Ankara

¹Bilgi Grubu Ltd, Ankara

¹e-posta: bozkan@ii.metu.edu.tr

²e-posta: ali.yildiz@bg.com.tr

³e-posta: demirors@ii.metu.edu.tr

Özetçe

Yazılım İşlevsel büyüklüğü erkenden ölçülebilir, yazılım dil, platform ve geliştirme yönteminden bağımsız niceleyebilmesinden dolayı proje yönetiminde kullanımı tercih edilen temel bir yazılım büyüklüğüdür. Albrecht'in, işlevsel büyüklük kavramını ilk kez ortaya atmasından bu yana otuz yıl geçmesine rağmen işlevsel büyüklük ölçümü (İBÖ) yazılım dünyasında yaygın uygulanmamaktadır. Bu durum, İBÖ yöntemleri yapılarındaki ve ölçüm pratiklerindeki sorunlarla açıklanarak tartışılmalıdır. Bu çalışmada¹, İBÖ sorunlarını tanımlama ve çözmeye yönelik formalizasyon çalışmalarını gözden geçirilip, bu konuda yapılan önerilerin katkıları ve etkinliği değerlendirilmiştir; değerlendirme sonuçları dikkate alınarak geliştirilmekte olan bir İBÖ ölçüm aracı tanıtılmıştır.

1. Giriş

İşlevsel büyüklük yazılım işlevselliğini niceleyen bir yazılım büyüklük boyutudur. Erkenden ölçülebilmesi ve insan, geliştirme yöntemi, dili ve platformdan bağımsız olması özellikleri ile yazılım kestirimi süreçlerine temel bir girdi olmuştur. Kavramın ilk kez tanıtılmasından [1] bu yana farklı İBÖ yöntemleri ve çeşitlemeleri geliştirilmiştir. Günümüzde, Uluslararası Standardizasyon Örgütü (ISO), COSMIC FSM, IFPUG FSM, FİSMA FSM, MARK II FPA ve NESMA ölçüm yöntemlerini standart olarak yayımlamıştır [2], [3], [4], [5], [6].

İşlevsellik büyüklük, planlama, izleme, maliyet kestirimi gibi birçok proje yönetim sürecini destekleyebilen bir metrik olmasına [21] ve İBÖ yöntemleri otuz yıla yakın bir süredir var olmasına rağmen, İBÖ yaygın bir yazılım pratiği olamamıştır. Yapılan çalışmalarda, işlevsel büyüklük ölçümünde yapı ve uygulama sorunları araştırılmış, İBÖ'nün yaygınlaşmasını engelleyen güçlükler tartışılmıştır [22], [23], [24], [25], [26], [27], [34], [35].

İBÖ ile ilgili problemlere tepkilerin bazıları, İBÖ'ne formalizasyonlar getirme önerileri içeren çalışmalarla gelmiştir. Bu çalışmaların asıl konuları, nesnel ölçüm, ölçüm sürecinin otomasyonu ve bir İBÖ yönteminin belirsizliklerden arındırma amacıyla anlamsal ve sözdizimsel öğeler ile tanımlanmasıdır.

Bu çalışmada, İBÖ yöntemleri ve uygulamalarındaki sorunlara, formalleştirme yaklaşımıyla çözüm arayan

çalışmalar değerlendirilmiştir. Burada bahsedilen formalizm İBÖ yöntem kavramlarının, ölçüm kurallarının ve ölçümün formel gösterimleri olarak anlaşılmalıdır. Çalışmadaki temel amaç, İBÖ sorunlarına çözüm önerilerini bir formalizasyon yaklaşımı izleyerek getiren çalışmaların değerlendirmelerini araştırmacılara sunmak, formalizasyon çalışmaları ile adreslenebilecek İBÖ sorunlarını tespit etmektir. İkincil amaç, çalışma sonuçları ışığında bir ölçüm aracının geliştirilmesi ile bazı formel önerileri uygulanabilir ve sınanabilir hale getirmektir.

Çalışmanın ikinci kısmında formalizasyon çalışmalarını değerlendirme süreci anlatılmaktadır. Üçüncü kısımda çalışmalar gruplandırılıp açıklanmaktadır. Dördüncü kısımda değerlendirme sonuçları tartışılıp, beşinci kısımda formalizasyon çalışmaları önerileri üzerine temellendirilen bir İBÖ aracı tanıtılmıştır. Son olarak altıncı bölümde bu çalışmanın sonucu verilmiştir.

2. Değerlendirme Süreci

2.1. Çalışma Seçimi İçin Kriterler

Bir çalışmayı değerlendirmeye katmak için belirlenen ana kriter, yapılan araştırmanın İBÖ üzerine olması ve araştırmanın belirlenen İBÖ sorunlarına karşı önerilerinde bir formalizasyon yaklaşımı izlemesidir. Formalizasyon yaklaşımı, İBÖ yöntemlerini ve ölçümlerini açıklamak üzere model, notasyon ve semantik tanımlamaları ile kısıtlı tutulmuştur. Çalışmalarda konu olan İBÖ yöntemleri, en bilinen yöntemler olan IFPUG, COSMIC ve MARK II² olarak belirlenmiştir. Formalizasyon yaklaşımlarının formal bir gösterimle ifade edilmesi bir kriter olarak konulmamış; konuşma diliyle ya da yarı formel bir gösterimle açıklanan formalizasyon yaklaşımları çalışmaya dahil edilmiştir. Çalışmaların bir kurum, konferans ya da dergi tarafından yayınlanmış olması önkoşul olarak belirlenmiştir ve doktora, yüksek lisans çalışmaları, teknik raporlar da incelemeye alınmıştır. Değerlendirmeye alınacak çalışmaların İBÖ yöntem kavramlarına ve kurallarına sadık olması, önerilerin yöntemlere uyarlama ve değişiklik getirmemesi gerekli görülmüştür. Bir İBÖ prosedürünün otomasyonu gibi [17], orijinal bir formalizasyon önerisinin devamı niteliğinde olan

¹ Bu çalışma TÜBİTAK tarafından 107E010 projesi kapsamında desteklenmektedir.

² Bildiri boyunca, bahsedilen üç İBÖ yöntemi, "tüm yöntemler" ve "tüm İBÖ yöntemleri" olarak da adlandırılmıştır.

çalışmalar da değerlendirmeye katılmıştır. Benzer biçimde formalizasyon önerisinin zaman içinde değişip evirilerle anlatıldığı çalışmalar da, önerilerin geçerlenmesi ve değişikliklerin izlenebilmesi amacıyla değerlendirmeye dahil edilmiştir [11], [12], [13]. Değerlendirmeye İngilizce basılmış yayınlar alınmıştır. Fransızca bazı yayınların konu ile ilgili olduğu ve değerlendirilmeye dahil edilen bazı makalelerce referans verildiği gözlemlenmiştir [28], [29].

2.2. Çalışmaların seçimi

Bu çalışmanın yazarları İBÖ konusunda araştırma yapmaktadırlar ve çeşitli konferans ve dergilerde yayımlanmış dört yüzü aşkın yayını içeren bir sayısal kütüphaneye sahiptirler. İlk olarak bu kütüphanede yayın başlığı ve özetçeleri taranarak dört yayın bulunmuştur [15], [12], [7], [9]. Bu yayınlar okunduktan sonra, yayında anlatılan çalışmaların referans verdiği diğer yayınlar internet arama motorları ve yayıncı sitelerinde taranıp bulunmuş, bulunan her bir yayın için bu işlem baştan tekrarlanmıştır. Bunun dışında yayınların genelde bulunduğu konferans ve dergi sitelerinde ve atıf izleme sitelerinde anahtar kelime aramaları yapılmıştır. Sonuç olarak değerlendirme katılım kriterlerine uyan üç dergi makalesi [17], [15], [13], yedi konferans bildirisi [11], [12], [7], [9], [10], [18], [19] ve dört teknik rapor [8], [20], [15], [14] değerlendirilmeye alınmıştır.

2.3. Çalışmanın geçerliliğine yönelik tehditler

Eksik Yayınlar: Değerlendirmeye alınan yayınlardan anlaşıldığı üzere, değerlendirilmeye alınmayan İngilizce dışında yazılan bazı yayınların konu ile ilgili olması olasıdır. Değerlendirilen yayınlarda bulunan İngilizce dışı yayınların özetlerinin ve açıklamalarının yayınları temsil ettiği düşünüldüğünde ve yazarların diğer makaleleri incelendiğinde [28], [18], [19], [29], [17], bu yayınların değerlendirilmeye alınmamasının değerlendirme sonuçlarında ciddi değişikliklere neden olamayacağı düşünülmüştür.

Anakronizm: Çalışmada diğer bir zafiyet yapılan çalışmaların 1991-2006 arası gibi İBÖ yöntemlerinin evirildiği, yeni sürümlerinin yayımlandığı, yeni İBÖ yöntemlerinin geliştirildiği geniş bir zaman aralığına yayılmasıdır. Bunun yanında, yazılım mühendisliğinde bu dönemde paradigma kaymaları ve değişiklikleri gerçekleşmiştir. İBÖ ilkeleri büyük ölçüde aynı kalsa da değerlendirmeye alınan formalizasyon çalışmaları ve bu değerlendirme çalışmasının sonuçları geçersiz ya da kullanışsız kalmış olabilir. Bu çalışmadaki analiz ve gözlemler İBÖ yöntemlerinde bulunan ve 1990larda ve günümüzde geçerli olduğu düşünülen bu temel prensip ve tanımlara dayandırılarak yapılmıştır.

3. Çalışmaların Özeti

Çalışmalar formalizasyon amaçlarına göre net bir şekilde ayrıştırılabilen iki kategoriye bölündü. Her iki kategorideki çalışmaların ortak olan motivasyonları, İBÖ yöntem kavram ve kurallarındaki belirsizlikleri ortadan kaldırmak, farklı kural ve kavram yorumlamaları ile oluşan ölçümcüler arası değişkenlikleri azaltmak, ölçüm hatalarını azaltmak ve otomasyon fırsatları oluşturmaktır. Kategoriler arası farklılaşma, formalizasyonlar uygulayarak sorunlara tepki verme biçimlerinde. İlk kategorideki çalışmalar formalizasyonun belli bir notasyon, spesifikasyon dili yada model ile tanımlanmış yazılım işlevselliğinin ölçümüne nasıl

katkıda bulunabileceğini anlatmaktadır. Bu kategori “Formal Spesifikasyonlardan İşlevsel Büyüklük Ölçümüne İlişkin Formalizasyon Çalışmaları” olarak adlandırılmıştır. İkinci kategori çalışmaları İBÖ yöntem kavramları ve ölçüm fonksiyonları için formel gösterimler önermiştir ve kategori “İBÖ Temellerindeki Formalizasyonlar” olarak adlandırılmıştır.

3.1. Formal Spesifikasyonlardan İşlevsel Büyüklük Ölçümüne İlişkin Formalizasyon Çalışmaları

İBÖ yöntemlerinde anlatılan ölçüm süreçlerinde ilk basamak ölçümcünün, yöntemin daha sonraki basamaklarda sayısal değer atayarak niceleyeceği, soyut yazılım modelindeki [12], [32] işlevsel bileşenlerin³, işlevsel kullanıcı gereksinimlerini analiz ederek çekip çıkartmasıdır. Bu işlem sırasında bileşenler yöntem tanımları ve kurallarını çerçevesinde belirlenir. Bu basamak gerek yöntemlerdeki eksikliklerden ve belirsizliklerden, gerek ölçümcü yorumlarına ve hatalarına açık olmasından dolayı, aynı işlevsel gereksinimlerden aynı ölçümcünün farklı ölçüm sonuçları ya da farklı ölçümcülerin ölçüm sonuçları arasında büyüklük farklılıkları oluşmasına sebep olabilmektedir [34], [35]. Bu kategorideki çalışmaları tipik özelliği İBÖ yöntemlerinin soyut yazılım modelindeki [12], [32] işlevsel bileşenler ile bir yazılımın işlevselliğinin tanımlanmasında kullanılan spesifikasyon dili öğelerinin eşleştirilmesidir (Örnek: veri akış gösteriminde bir verideposu (datastore) elemanı IFPUG yöntem modelinde bir mantıksal dosyaya (logical file) karşılık gelir). Böylelikle işlevsel büyüklük ara bir işleme gerek kalmadan doğrudan ölçülmesi ile daha tutarlı ve doğru ölçümler yapılacaktır. Önerilen formalizasyonlar da, bahsedilen eşleştirmeleri yöntem kavram ve kurallarının bir spesifikasyon dili için yorumlayarak, dil bağlamında oluşturulan formel gösterimli bir kurallar kümesi oluşturulmuştur.

Bir grup çalışma formel spesifikasyon dilinde ifade edilmiş işlevsel gereksinimlerin işlevsel büyüklüğünü ölçmek için prosedürler önermiştir. Frappier [8] formel spesifikasyon dillerini sınıflandırarak IFPUG kavramlarına uyumluluğunu ve doğrudan ölçümler için yeterliliğini değerlendirmiştir. Çalışmanın devamında [7], B spesifikasyon dilinin sözdizimsel analizi sonrası, IFPUG kavramları ve B spesifikasyonları arası eşleştirmeler ve işlevsel bileşenlerin belirlenmesi için formel kurallar tanımlanmıştır. Ayrıca formel bir yaklaşımın yöntem yapısındaki problemlerin gösterimi için de kullanılabileceği örnek durumlarla gösterilmiştir. Diab [9] aynı yaklaşımı ROOM (Real-Time Object-Oriented Modeling) spesifikasyonları ve COSMIC yöntemi için izlemiştir. Çalışmada COSMIC kuralları ve kavramları ROOM için yorumlanmış, birinci seviye mantık ve küme teorisi ile gösterilmiştir. Çalışmanın devamında [17], kuralların programlanmasıyla ROOM spesifikasyonlarından otomatik ölçüm yapan bir araç geliştirilmiş, bir durum çalışmasıyla bir COSMIC uzman ölçümcünün elle yaptığı ölçüm ile otomatik ölçüm sonuçları karşılaştırılmış, farklılıkların nedenleri araştırılmıştır. Benzeri biçimde Miyawaki [10] IFPUG

³ Çalışma boyunca “İşlevsel Bileşen” İBÖ yöntemlerindeki yazılım soyut modellerinde tanımlanan kavramları ifade etmek için kullanılmıştır. (IFPUG modelinde ILF, EIF mantıksal dosyaları, temel süreç, işlem tipi ya da COSMIC modelinde veri grubu, veri hareketi, işlevsel süreç, katman gibi.)

yöntemi ile VDM-SL (Vienna Development Method-Specification Language) dilindeki işlevsel gereksinimlerin ölçülebilmesi için geliştirilen kuralları matematiksel gösterimle sunmuştur.

Bu gruptaki çalışmaların temel motivasyonu işlevsel büyüklük ölçümünün otomasyonu olmuştur. Her bir formel spesifikasyon dili için formel gösterimler ile ifade edilen yöntem yorumları, yöntemlerin prensip, model ve işlevsel bileşenlerine uyumlu ve tekrarlanabilir olmaları sebebiyle tutarlı ölçüm sonuçları alınmasına ortam sağlamıştır.

İki çalışmadan oluşan diğer bir grup çalışma IFPUG yöntemi ile Varlık-İlişki(ER) ve Veri Akış Diyagramı(DFD) ile ifade edilen işlevsel gereksinimlerin büyüklüklerinin ölçülmesi için prosedürler önermiştir. Rask [20] DFD elemanları ile IFPUG kavramları arasında kavramsal denklikler önermiştir. Anlatılan prosedürde bir seri algoritma DFD ve ER elemanları cinsinden öğeler ve ifadeler cinsinden verilmiştir. Daha sonra, bu prosedür işlevsel büyüklük ile diğer bir işlevsel büyüklüğü karşılaştıran bir simülasyon çalışmasında [30] kullanılmıştır. Gramantieri [14] benzeri bir yaklaşımla geliştirdiği prosedürde farklı olarak DFD verideolarını ER 'varlık ve ilişkileriyle göstermiştir. IFPUG kurallarının anlatıldığı koşullar, bu notasyonla formel kurallara dönüştürülmüştür ve devamında prolog mantıksal programlama diliyle programlanmıştır. Lamma [13] bu çalışmayı durum çalışmaları ile genişletmiş, elle ve otomatik olarak yapılan ölçümlerin birbirine yakın olduğunu göstermiştir. Bu ikinci grup çalışma ilk gruptaki çalışmalar ile benzer karakteristikler gösterir. Her ikisinde de kuralların uygulanabilirliği kabullenmelere ve IFPUG sayım ve işlevsel eleman belirleme kurallarının yorumlanmasına bağlıdır. DFD ve ER gösterimlerin yorumlanmasındaki bir farklılık IFPUG yöntem kurallarının ve kavramlarının halihazırda veri analiz terminolojisi ile açıklanması sebebiyle daha rahat eşleştirilebilmesidir. Son IFPUG sayım uygulamaları el kitabı sürümünde [33] IFPUG kurallarını ve işlevsel elemanlarını ER gösterimleri ile tanımlanmasına rehberlik eden bir bölüm içermektedir. Farklı bir çalışmada Abran ve Paton [16], DFD benzeri bir notasyon ile IFPUG kurallarını ifade ederek, yöntemin hangi kurallarının otomasyona uygun olduğunu belirlemişlerdir. Notasyon süreçlerin, veri depolarının, veri manipulasyonlarının ve yazılım sınırının gösterilebileceği bir küme grafiksel sembollerden oluşmaktadır. Olası işlevsel gereksinim kalıpları ve durumları bu notasyon ile gösterilmiş kurallar uygulanarak değerlendirilmiştir. Bunun sonucu ölçüm sırasında insan müdahalesi ve kararları gerektiren durumlar belirlenmiştir. Çalışmada, notasyon yöntemce dikkate alınan veri alanları, veri dosyaları, kayıt tiplerin gösteren elemanlarla genişletilerek bu notasyonla ifade edilmiş gereksinimler için bir de ölçüm prosedürü tanımlanmıştır.

Tablo 1'de bu kategoride anlatılan, İBÖ kavram ve kurallarının otomasyon amacıyla formel biçimde bir spesifikasyon dili için yeniden tanımlayan çalışmaların özeti verilmiştir.

3.2. İBÖ temellerindeki formalizasyonlar

İşlevsel büyüklük ölçme yöntemleri işlevsel büyüklüğü işlevsel kullanıcı gereksinimlerinden ölçer. Tüm İBÖ yöntemleri tanımlanan yazılımın, yazılım işlevselliği ile ilgili olduğu düşünülen kavramlardan oluşan bir yöntem yazılım

modeline kurallar kümesi aracılığı ile soyutlanmasına ve dönüştürülmesine gerek duyar [32], [12]. Bu kategorideki çalışmalar, İBÖ soyut modellerindeki işlevsel bileşenlerinin ve bileşenlere sayısal değerlerin atandığı ölçüm fonksiyonlarının gösterimlerinde formalizasyonlar önermektedir.

Tablo 1: Farklı yazılım tanım dillerinden işlevsel büyüklük ölçümündeki formalizasyon çalışmaları özeti.

Araştırma referansı		Spesifikasyon Dili	İBÖ yöntemi
Orijinal Çalışma	Devam Çalışması		
[9]	[17]	ROOM	COSMIC
[20]	[30]	ER,DFD	IFPUG
[14]	[13]	ER+DFD	
[7]		B	IFPUG
[10]		VDM-SL	IFPUG

Çalışmaların amacı İBÖ modellerine formel tanımlar getirerek yöntemlerdeki belirsizliklerden kaynaklanan sorunları daha açık göstererek çözümüne katkı sunmak, değişik İBÖ yöntemleri arasındaki yapısal farkları göstermek, yöntemlerin iyileştirilebilmesi ve otomasyonu için fırsatları görünür hale getirmek, kısaca İBÖ yöntemlerini daha anlaşılır kılmaktır.

Fetcke [11] çalışmasında IFPUG FPA, MARK II FPA ve COSMIC FFP yöntemleri için geliştirilmiş bir yapı tanımlanmıştır. Çalışmada, ölçüm adımları, gereksinimlerin veri yönelimli bir soyutlama ile gösterilmesi ve bu gösterimdeki elemanlara sayısal büyüklük değerleri atanması olarak belirlenmiştir. Etkinlik tipi kavramını geliştirilmiş yapıya ekleyerek, yöntemler arasındaki ortaklıkları ve farklılıkları gösterilmesine olanak sağlamıştır. Bu soyutlama bir matematiksel model ile gösterilmiş, çalışmanın devamında [12] model genel gösterim COSMIC FFP, IFPUG and MARK II yöntemlerinin kavram tanımları ve kurallarına karşı sınımlanmıştır. Hericko [15], Fetcke'nin tanımladığı geliştirilmiş modelin her üç yöntem için matematiksel gösterimi ve sembolik kod notasyonu ile ölçüm fonksiyonlarının gösterimi verilmiştir. Birleştirilmiş Modelleme Dili (UML) ile ifade edilen işlevsel gereksinimlerim genel modele dönüştürüldüğü ve işlevselliğin formel olarak gösterildiği, nesne yönelimli spesifikasyonlar için bir örnek süreç uygulaması verilmiştir.

Bevo [18], [19] çalışmalarında bir İBÖ yöntemi uygulama sürecini ontolojik formalizasyon ile göstermiştir. Çalışmada İBÖ uygulamalarındaki güçlükler ve otomatik ölçüm için araç eksikliği vurgulanmış, İBÖ'de alan ve faaliyet ontolojilerinin gerekliliği açıklanmıştır. Bevo önerilen ontolojik formalizasyonların, işlevsel büyüklük ölçüm süreçlerinin daha iyi anlaşılmasına katkı bulunacağı gibi bir İBÖ yönteminin uygulanma prosedürünün yapılandırılması, ifade edilmesi ve bilgi paylaşılması için bir buluşma noktası oluşturacağını düşünmüştür. Ontoloji gösterimi için nesne eğilimli formalizasyonlar uygun görülmüştür. Çalışmada IFPUG, COSMIC ve MARK II FSM yöntemleri için ontolojiler örneklendirilmiştir.

4. Değerlendirme Sonuçları

İBÖ yöntemleri işlevsel büyüklüğü işlevsel kullanıcı gereksinimlerinde tanımlanan yazılım işlevselliğinden ölçer. Tüm İBÖ yöntemleri işlevsel kullanıcı gereksinimlerinde tanımlanan yazılımın, yöntemce kurallar ve tanımlarla belirlenmiş bir yazılım modeline soyutlanmasına, dönüştürülmesini gerekli görür.

Bu çalışmada değerlendirilen ve İBÖ literatüründe bulunan diğer birçok çalışma ortak bir görüşle, İBÖ yöntem uygulamalarının soyut yazılım modellerine yorumlamalar olmaksızın dönüştürülmesi, pratikte, işlevsel büyüklüğü doğru ve tam olarak ölçülemediği sonuçlara yol açacağı saptamasında bulunmaktadır. İşlevsel kullanıcı gereksinimleri birçok farklı gösterimle ifade edilebileceği için ölçümcünün soyutlama yaklaşımı ve yorumlama biçimine bağlı farklı ölçüm sonuçları elde edilebilir. İlk kategorideki çalışmalarda formal ifade edilen yorumlamalar, bir spesifikasyon dili için tekrarlanabilir ve tutarlı ölçümleri olanaklı kılarak soruna kısmi çözüm sağlamışlardır. Yorumlamaların öznel olması, yine bir formal gösterimle ifade edilebilen ve tutarlı sonuçlar üreten başka bir yorum ile farklı sonuçlar bulunması sorunu beraberinde taşımaktadır. Bu nedenden dolayı ilk kategori çalışmaları bağlamında öznel ölçüm problemi geçerli kalmaktadır. İlk kategori çalışmalarının formal spesifikasyon dillerine odaklanması bir tesadüf değildir. Temel amaçları ölçüm otomasyonu olduğu için daha formal biçimde ifade edilen gereksinimlerden, gereksinimlerdeki belirsizliklerden kaynaklanan ölçüm eksiklikleri ile ilgili sorunların izole edip gereksinim mühendisliği altına delege ederek, ölçüm yapma daha fazla otomasyon fırsatı sunmaktadır.

İkinci kategori yayınların temel uğraşı İBÖ yöntemlerinin soyut yazılım modellerinin, model öğelerinin, öğeler arası ilişkilerin ve sayısal değer atama fonksiyonlarının formal gösterimi ile yöntemlerin ve yöntemle ilgili sorunların daha açık ve anlaşılır kılınmasıdır. [12] deki genelleştirilmiş soyutlama modeli tüm İBÖ yöntemlerince yazılım işlevselliği ile ilgili görülen bütün bilgileri içermek üzere yapılandırılmıştır. İBÖ yöntemleri aralarında benzerlikler olmasına rağmen, her yöntem işlevsellikle ilgili olduğunu düşündüğü farklı şeyleri ölçer [23], [36]. [15] teki çalışmada genelleştirilmiş model genel bir ölçüm sürecinin odağına konulmuştur fakat ölçüm sürecinde “boşluk” problemi devam etmekte, öznel yorumlar yapılarak yazılım tanımlama öğeleri genelleştirilmiş model öğeleri ile tutarlı biçimde eşleştirilmektedir. Her İBÖ yönteminde işlevsel bileşen tanımlama kuralları verilmiştir. Örneğin COSMIC yönteminde veri grupları işlevsel bileşeni belirleme için şu kural verilmiştir: “Her veri grubu doğrudan işlevsel kullanıcı gereksinimlerindeki bir ilgi nesnesiyle ilgili olmalıdır”. Açıkça ifade edildiği gibi, ilgi nesnelerin doğru bir şekilde belirlenmesi, belirlenmiş bir veri grubun geçerlenmesi için önkoşuldur. Genelleştirilmiş model İBÖ kurallarını soyutlamamaktadır.

5. İşlevsel Büyüklük Ölçüm Aracı Tanıtımı

Formalizasyon çalışmalarının değerlendirmesi sonuçları çalışmaların genelinde, IFPUG, COSMIC ve MARK II İBÖ

yöntemlerinin yazılım modelleri farklı kullanıcı işlevsel gereksinimleri ifade sivil ve biçimleri için soyut kaldığı, farklı gösterim dilleri için yöntem kılavuzları bulunmadığı, bu nedenle yöntem modeline dönüşümlerde farklı ölçümcü yorumlarının farklı ölçüm sonuçlarına yol açabileceği saptandığını göstermiştir. Yöntem kurallarının ve kavramlarının anlatıldığı el kitaplarında anlatması güç ya da belirsizlik görülen yerlerde Varlık-İlişki gibi gösterimlere başvurulması bu sorunun geçerliliğinin bir göstergesi olarak düşünülebilir. Çalışmalarda önerilen bir yaklaşım farklı gereksinim gösterimleri için formal olarak tanımlanmış yorumlar geliştirmek yoluyla tekrarlanabilir ve tutarlı ölçümler yapılabilmesidir. Diğer bir genel sonuç, üç yönteminde yazılım işlevselliği algısını ve anlayışını temsil edebilen tek bir genelleştirilmiş ve formelleştirilebilen modelin yöntemler arası farklılıkları daha açık gösterebildiği gibi, ölçümcülerin ölçüm yöntemlerinin detaylarını bilmeden gereksinimleri tek bir soyut modele dönüştürerek her üç yöntem cinsinden de sonuçlar elde etme olanağını bulabilmesidir. Genelleştirilmiş modelin bu tür bir kullanımdaki karşılaşılan bir sorun, tüm modellerde ortak bulunan veri grubu ve alt grupları tanımların, yöntemlerde ki farklı veri grubu belirleme ve geçerleme kuralları ile yöntemlerde farklı kavramlara karşılık gelmesidir.

Bu genel sonuçlar ışığında geliştirilmesi düşünülen bir ölçüm aracı için şu öneriler getirilmiştir:

Gereksinimde tanımlanan verilerin Varlık-İlişki notasyonu ile modellenmesi-Ölçüm yöntemlerindeki ortak kavram olan ve doğru belirlenmesi ölçüm için oldukça kritik olan veri gruplarının işlevsel kullanıcı gereksinimlerinden tutarlı olarak belirlenmesi için gereksinimlerde anlatılan verilerin modellemeye yaygın olarak kullanılan Veri-İlişki (ER) notasyonu ile gösterilerek, araca girdi sağlaması önerilmiştir. Bu şekilde kullanıcı gereksinimlerinin bir parçası olan veriler tutarlı ve ölçüm için her yöntem için yeterli bir detay ve formalitede ifade edilebilir. Her üç yöntemin kural ve tanım anlatımlarında ve uygulama örneklerinde kullanılan varlık-ilişki(ER) formalitesi yorumlamaların tutarlılığın ötesinde, yöntemlerce doğrudan ya da dolaylı şekilde doğrulanmasını da sağladığı için en uygun gösterim olarak seçilmiştir. Hali hazırda IFPUG ER pratikleri rehberinde vermiştir. Benzeri bir çalışma COSMIC yöntemi için de yapılmıştır [38]. ER formalitesindeki çokluk (multiplicity), katılım (participation) ve bağımlılık (dependency) gösterimleri veri grupların belirlenmesi ve alt grupların ayrıştırılmasında yeterli olmaktadır.

Genelleştirilmeyle geliştirilmiş birleşik bir yazılım soyut modelinin baz alınması- İşlem veri grupları gibi ölçümde belirlenmesi kritik olan bir kavramdır. Bu kavramın yapılan yöntem analizlerinde [36] her üç yöntem için ortak ve aynı kavrama denk geldiği saptanmıştır. İşlem tipleri Fetcke'nin [12] de yaptığı çalışma temel alınarak işlemler içinde gerçekleşen aktivitelerin belirlenmesi ile belirlenebilir. Veri gruplarının tanımlanmasında ER formalitesinin kullanılması birlikte düşünüldüğünde, tek bir soyut model ile her üç yöntemin ihtiyaç duyduğu detay ve formalite elde edilmiş olacaktır. Bu model birleşik model olarak adlandırılacaktır.

Yukarıdaki öneriler uygulandığında ölçüm için aşağıdaki adımlardan oluşacaktır:

1-İşlevsel kullanıcı gereksinimlerinden işlemlerin belirlenmesi, ER modelinin oluşturulması.

2-ER modelinin araca tanıtılması sonucu veri gruplarının ve alt grupların ve veri elemanlarının otomatik oluşturulması ve ilişkilendirilmesi.

3-Birleştirilmiş modele uyumlu işlemlerin araçta tanımlanması ve işlemler içinde aktivitelerinin ilgili olduğu veri tipi(grup, alt grup, eleman) tanımlanması.

4-Büyüklik ölçüm fonksiyonlarının otomatik olarak çalıştırılıp üç yöntem için teknik etmenlerden bağımsız ayarlanmamış (unadjusted) işlevsel büyüklüğün hesaplanması.

Bu önerilerin temelini oluşturduğu, bir prototip ile geliştirilmeye başlanmış olan araçta yukarıdaki odak özellikler dışında:

- Ölçülecek yazılım projelerini yönetilmesini,
- Ölçüm uygulamaların kapsam, amaç gibi bilgilerinin saklanması,
- Proje yönetiminde ihtiyaç duyulan, proje süresince gerçekleşen işlevsel değişikliklerin büyüklüklerini ölçümünü destekleyecek otomatik ölçüm karşılaştırmasını,
- Birleşik yöntemden bağımsız seçilen tek bir yöntem için, daha çok birleşik yöntemle gerçekleştirilen ölçümlerle karşılaştırmak amacıyla, doğrudan ölçüm yapılmasını,
- ISBSG proje veri tabanına uygun biçimde veri aktarabilmek üzere soru listesinin doldurulmasını
- Projelerin, ölçümlerin ve ölçüm bilgilerinin versiyonlanmasını,
- Ölçüm ve karşılaştırmaların sonuçlarının raporlanmasını

destekleyecek özellikler eklenmesi planlanmaktadır.

Aracın geliştirilmesi sonucunda, çalışmada incelenen bazı formalizasyon yaklaşımları uygulanmış, bu formalizasyonların ölçümün otomasyonu, doğruluğu ve tekrarlanabilirliği ile ilgili sorunların çözümüne katkısı olup olmadığı sınanabilir hale getirilmiş olacaktır. Bununla beraber üç yöntem cinsinden otomatik ölçülen büyüklükler, bu yöntemlerin herhangi biriyle ölçülmüş geçmiş proje verilerini işgücü kestirimi gibi amaçlar için kullanılabilir hale getirecektir.

6. Sonuç

Bu çalışmada IFPUG, COSMIC ve MARK II İBÖ yöntemlerinde ve uygulamalarındaki formalizasyon çalışmalarını incelenip değerlendirilmiştir. İBÖ sorunlarının çözümüne katkıda bulunabilecek noktalar saptanmış, bu katkıları sınamak için formalizasyon önerileri bir ölçüm aracı ile uygulanmaya başlanmıştır.

Aracın geliştirilmesiyle birlikte birleşik model temelli ölçümlerin, ölçümlerdeki doğruluğu, tutarlılığı ve tekrarlanabilirliğini artırması beklenmektedir. Yapılacak durum çalışmalarında, araçta uygulanan birleşik model temelli ölçümlerin etkinliği, bu kalite özniteliklerinin performansları ile değerlendirilecektir. Bununla beraber hayli zaman ve işgücü alan ve hali hazırda az sayıda bulunan İBÖ yöntem uzmanlarına gerek duyulan doğrudan ve elle ölçüm, formalizasyonlarla otomatikleştirilerek, bu olumsuz etkileri azaltması, dolayısıyla İBÖ uygulamalarının yaygınlaşmasına katkıda bulunması beklenen diğer bir sonuçtur.

7. Kaynakça

- [1] Albrecht, A.J.: Measuring application development productivity. IBM Application Development Symposium, pp. 83–92 (1979)
- [2] ISO/IEC 19761:2003 COSMIC Full Function Points Measurement Manual v. 2.2.
- [3] ISO/IEC 20926:2003 Software Engineering - IFPUG 4.1 Unadjusted FSM Method - Counting Practices Manual.
- [4] ISO/IEC 20968:2002 Software Engineering - Mk II Function Point Analysis - Counting Practices Manual.
- [5] ISO/IEC 24570:2005 Software Engineering - NESMA Functional Size Measurement Method v.2.1 - Definitions and counting guidelines for the application of Function Point Analysis
- [6] ISO/IEC 29881:2008 Information technology -- Software and systems engineering -- FiSMA 1.1 functional size measurement method
- [7] Diab, H., Frappier, M., and St.-Denis, R. 2002. A Formal Definition of Function Points for Automated Measurement of B Specifications. Formal Methods and Software Engineering :483-494.
- [8] Frappier, M.: An Overview of Formal Specification Languages and their Adequacy for Formalizing the Definition of Function Points, Technical Report, D'epartement de math'ematiques et d'informatique, Universit'e de Sherbrooke, Sherbrooke, February,
- [9] Diab, H., Frappier, M., and St-Denis, R. 2001. Formalizing COSMIC-FFP Using ROOM. AICCSA 2001.
- [10] Miyawaki, T.; Iijima, J.; Sho Ho ,Measuring function points from VDM-SL specifications, ICSSSM 2008: 1-6.
- [11] Fetcke, T., 1999a. A generalized structure for function point analysis. IWSM 1999, Canada: 1-11.
- [12] Fetcke, Th.; Abran, A.; Dumke, R.: A Generalized Representation for Selected Functional Size Measurement Methods In Current Trends in Software Measurement, R. Dumke and A. Abran, Eds. Shaker, 2001, pp. 1-25.
- [13] Evelina Lamma, Paola Mello, and Fabrizio Riguzzi. A system for measuring function points from an ER-DFD specification. The Computer Journal, 47(3):358-372, May 2004.
- [14] F. Gramantieri, E. Lamma, P. Mello, and F. Riguzzi, "A system for measuring function points from specifications" Technical Report, DEIS-LIA-97-006, 1997.
- [15] Heričko, M., Rozman, I., and Živkovič, A. 2006. A formal representation of functional size measurement methods. J. Syst. Softw. 79, 9 (Sep. 2006), 1341-1358.
- [16] Abran, A.; Paton, K.: A Formal Notation for the Rules of Function Point Analysis. Research Report 247, University of Quebec, Montreal, April 1995
- [17] Diab H., Koukane F., Frappier M., St-Denis. R.µCROSE: Automated Measurement of COS-MIC-FFP for. Rational Rose Real Time. Information and Software. Technology, 2005, 47(3) : 151-166.

4. ULUSAL YAZILIM MÜHENDİSLİĞİ SEMPOZYUMU - UYMS'09

- [18] Evariste Valéry Bévo Wandji, Ghislain Lévesque, Jean Guy Meunier: Toward an ontological formalisation for a software functional size measurement method's application process: The FPA case. RIVF 2004: 113-118
- [19] Evariste Valéry Bévo Wandji, Ghislain Lévesque, Jean Guy Meunier: Toward an ontological formalisation for a software functional size measurement method's application process: The COSMIC FFP case. IWSM 2003: 113-118
- [20] Rask, R., 1991, Algorithms for Counting Unadjusted Function points from Dataflow Diagrams, Research Report, University of Joensuu, Finland.
- [21] Baris Ozkan, Oktay Türetken, Onur Demirörs: Software Functional Size: For Cost Estimation and More. EuroSPI 2008: 59-69
- [22] Lother, M., Dumke, R.: Points Metrics - Comparison and Analysis. In: International Workshop on Software Measurement (IWSM 2001), Montréal, Québec, pp. 155–172 (2001)
- [23] Çigdem Gencel, Onur Demirörs: Functional size measurement revisited. ACM Trans. Softw. Eng. Methodol. 17(3): (2008)
- [24] The Problem with Function Points Kitchenham,B.; Software, IEEE Volume 14, Issue 2, Mar/Apr 1997 Page(s):29 – 31
- [25] G. Poels, “Why Function Points Do Not Work: InSearch of New Software Measurement Strategies”, Guide Share Europe Journal, vol. 1, 1996, pp. 9-26.
- [26] B. Kitchenham and K. Känsälä, “Inter–item correlations among function points,” in Proc. 1st Int. Metrics Symp., Baltimore, MD, May 21–22, 1993, pp. 11–14.
- [27] Meli R., "Functional metrics: problems and possible solutions", Proceedings of the FESMÁ98, 1998.
- [28] Bevo V., 2005, Analyse et formalisation ontologiques des mesures associées aux méthodes de mesure de la taille fonctionnelle des logiciels: de nouvelles perspectives pour la mesure, Ph. D. thesis in Cognitive Informatics, UQAM.
- [29] Diab, H. 2003. Formalisation et automatisation de la mesure des points de fonction. Ph.D. Thesis. Département d'informatique, Faculté des sciences, Université de Sherbrooke, Sherbrooke (Québec) Canada.
- [30] Simulation and comparison of Albrecht's function point and DeMarco's function bang metrics in a CASE environment, Rask, R.; Laamanen, P.; Lyytinen, K.; Software Engineering, IEEE Transactions on Volume 19, Issue 7, July 1993:661 - 671
- [31] Beatriz Marín, Giovanni Giachetti, Oscar Pastor: Measurement of Functional Size in Conceptual Models: A Survey of Measurement Procedures Based on COSMIC. IWSM/Metrikon/Mensura 2008: 170-183
- [32] Jacquet, J.-P.; Abran, A., From Software Metrics to Software Measurement Methods: A Process Model , (ISESS'97) , Walnut Creek, CA , IEEE Computer Society , 1997:128-135 .
- [33] IFPUG, Counting Practices Manual, Release 4.2, IFPUG, Princeton Junction, NJ, January 2004, ISBN 0-963-1742-9-0
- [34] Turetken, O., Demirors, O., Gencel, C., Ozcan Top, O., and Ozkan, B. 2008. The Effect of Entity Generalization on Software Functional Sizing: A Case Study. , PROFES 2008:105-116.
- [35] Oktay Türetken, Ozden Ozcan Top, Baris Ozkan, Onur Demirörs: The Impact of Individual Assumptions on Functional Size Measurement. IWSM/Metrikon/Mensura 2008: 155-169
- [36] Çigdem Gencel, Onur Demirörs: Conceptual Differences Among Functional Size Measurement Methods. ESEM 2007: 305-313
- [37] The Common Software Measurement International Consortium (COSMIC): Guideline for Sizing Business Applications Software Using COSMIC-FFP, Version 1.0 (2005)
- [38] Turetken, O. (2008): Mapping IFPUG & COSMIC FSM Terminologies to ER Concepts , METU II. Technical Report METU/II-TR-2008-22. Middle East Technical University, Informatics Institute.