

# Çevik Metotların Uyarlanması Üzerine Bir Çalışma

Mehmet Nafiz Aydın<sup>1</sup>, Robert Stegwee<sup>2</sup>, Kees van Slooten<sup>3</sup>, Frank Harmsen<sup>4</sup>,  
<sup>1,2,3</sup> Twente Üniversitesi, 7500 AE, Enschede, Hollanda  
<sup>1</sup>m.n.aydin@utwente.nl, <sup>2</sup>r.a.stegwee@utwente.nl, <sup>3</sup>cvs@utwente.nl

<sup>4</sup>Cap Gemini Ernst & Young, Hollanda  
frank.harmsen@cgey.nl

**Özet.** Bilişim Sistemleri(BS)nin geliştirilmesinde metotların etkin kullanılması üzerine yapılan çalışmalar metotların tanımlandığı şekilde kullanılmadığı ve eldeki duruma göre değiştirildiğini gösteriyor. Bu gerçek, metotların doğal olarak durumlara göre *uyarlanması* gereğini ortaya koyar. Son yıllarda ‘geleneksel’ metotların duruma uyarlanması zorluğuna dikkat çeken araştırmacılar, ‘çevik’ metodlar olarak bilenen bir yaklaşımın BSnin geliştirilmesi süreci üzerindeki etkilerini araştırmaktadır. Bu çalışma, bu motivasyon çerçevesinde Avrupa’nın finansal alanda lider organizasyonlarından birinin sistem geliştirme biriminde bir çevik metodun (DSDM-Dynamic System Development Method) uyarlanmasına ilişkin gözlemi iki ayrı teorik yaklaşımdan – mühendislik ve sosyal-örgüt yaklaşım – ele alarak gerek araştırmacılara gerekse pratikte metot kullanıcılarına önemli mesajlar sunmaktadır.

## 1 Giriş

Bilişim sistemler(BS)inde metotların etkin kullanılmasına ilişkin çalışmalar gerek akademik alanda gerekse pratikte güncelliğini korumaktadır. Metodun etkin olması ile anlatılmak istenen sistem geliştirmede üstlendiği rolün problem çözmeyi amaçlayan bir araç olması gerçeğidir [1]. Ancak araştırmalar pratikte metodun araç olmak yerine bazen amaç haline getirilmeye çalışıldığını ve bu yönüyle pratisyenlerin işlerine destek olup kolaylaştırmak yerine, onları sınırladığı ve belli bir kalıba uygun düşünmeyi ve çalışmayı zorladığını göstermektedir [2]. Metotların pratisyenler tarafından çoğunlukla tam uygulanmadığı veya bir kısmının uygulandığı ya da dışlandığı ve tamamen reddedildiği gerçeği ortaya çıkarıldı [2] [12]. Yine benzer araştırmalar, çalışma alanında sıkça kullanılan marka haline gelmiş, doğrusal ya da ‘ağır’ metotlar olarak bilinen geleneksel metotların “her projeye aynı kalıp” kabulünü taşıdığı göstermektedir [3] [4]. Oysa sistem geliştirme projelerinin doğal gereği olarak, her proje sahip olduğu karakter ile kendini diğer projelerden farklılaştırır ve her projenin özellikleri bulunduğu duruma bağlı olarak özeldir. O halde metodun projenin sahip olduğu duruma göre uyarlanabilmesi metodun projede kullanımını kolaylaştırabilir ve etkinliğini artırabilir.

Son yıllarda çevik metodlar olarak isimlendirilen metotların insan unsuruna, son kullanıcıya, sistem geliştirmede süreç yerine ürüne odaklı sistem geliştirme yaklaşımın önemine dikkat çekilmektedir. Biz bu çalışmada bu yaklaşımı temsil eden bir metodun farklı sistem geliştirme projelerinde nasıl uyarlandığı konusundaki gözlemlerimizi sunacağız. Bu bağlamda Avrupa’nın finansal alanda lider organizasyonlarından birinin sistem geliştirme biriminde metot uyarlanması ile ilgili gözlemlerimizi akademisyenler ve pratisyenlerle paylaşmak istiyoruz.

## 2 Araştırmanın Tasarımı

Bu bölümde kısaca gözlemin yapıldığı organizasyona, metoda, araştırmadaki temel sorulara, araştırmada kullanılan yaklaşım ve metotlara, veri kaynaklarına kısaca değinilecektir. Daha sonraki bölümde erişilen bulguların sunumu ve iki ayrı yaklaşım kullanarak ilgili bulguların değerlendirilmesi yapılacaktır. Son bölümde araştırmanın sonuçları ve gelecekteki olası araştırma konularına dikkat çekilecektir.

### 2.1 Organizasyon ve Metod Hakkında Özet Bilgi

Söz konusu organizasyon aktif değerleri açısından dünyanın en büyük on finans kuruluşundan biridir. Araştırma üç stratejik iş biriminden biri olan bireysel bankacılık iş biriminin altında bulunan sistem geliştirme biriminden yapılmıştır. Bu birimde 2,000 dolayında sistem geliştirme uzmanı çalışmaktadır. Bu birim, CMM modelini ve DSDM (Dynamic System Development Method – Dinamik Sistem Geliştirme Metodu) adlı bir metodla entegre etmeye çalışması açısından özel bir vaka olarak kabul edilebilir. Bu birim, paket BS uygulaması (örneğin SAP, PeopleSoft), uyarlanmış sistem geliştirme, bileşenlere veya nesneye dayalı sistem geliştirme gibi farklı uygulamalarla ilgili bir çok projenin gerçekleştirilmesinden sorumludur. Söz konusu bölüm, son yıllarda DSDM adındaki metodun tüm projelerde kullanımı kolaylaştırmak için metot uzmanlarından oluşan bir grup oluşturulmuştur. Bu uzman grup metodun daha etkin kullanılması konusunda proje liderlerine yardımcı olmayı amaçlamıştır. Bu grup bir anlamda eğitmen rolü ile proje takımında koç olarak isimlendirilmektedir. Araştırmanın yazarlarından biri bu grubun bir elemanı olarak gözlemlerde bulunmuştur ve aktif olarak bazı projelere katılmış, proje liderlerine destek olmuştur.

Söz konusu metot (DSDM) bir kuruluşun bağımsız olarak Avrupa’daki şirketlerin ağırlıklı olarak bulunduğu bir konsorsiyum tarafından ‘de facto’ metod olarak kabul edilmektedir [5]. Metot dokuz ana prensip üzerinde kurulu olup beş aşamayı içerir. Prensipler arasında *yinelemeli sistem geliştirme*, *aktif kullanıcı katılımı*, *entegre test*, *iş fikrine odaklı sistem geliştirme* gibi kavramlar öne çıkar. Önemli teknikleri arasında *MoSCoW* ile ihtiyaçların önem sırasına göre sıralanması, *zaman kutuları* ile sonuca odaklı sistem geliştirme, *protip* kullanımı ile kullanıcıdan erken geri dönüşüm

almak amaçlanmaktadır. Metot hangi tür projelerde kendisinin daha uygun olabileceğini belirlemek üzere Risk ve Uygunluk (R/U) analizi adında bir araç sunar. Bu araçtaki sorulara verilen cevaba göre metodun projeye uygunluğuna karar verilir. Bazı durumlarda metodun tümü ya da bir kısmı uygun bulunurken bazı durumlarda metodun kavramları ve teknikleri bir şekilde uyarlanarak sistem geliştirme sürecinde kullanılmaya çalışılır. Metodun detayları hakkında bilgiye ulaşmak için [5]'e bakınız.

## 2.2 Araştırmanın Ana Soruları ve Veri Toplama Metodu

Daha önce belirtildiği gibi amacımız çevik metodun pratikte farklı proje durumlarına göre nasıl uyarlandığını ortaya çıkarmaktır. Bu amaca uygun olarak bazı sorulara cevap bulmaya çalıştık. Bunlar şöyle sıralanabilir: (a1) Genel olarak pratisyenlerin metodu uyarlanmasına ilişkin ortak ve farklı adımlar nelerdir? (a2) Metot uyarlanması sırasında bir araç kullanılıyor mu? (a3) Pratisyenler projenin durumunu nasıl analiz ediyorlar? (a4) Projenin karakterini bazı faktörler kullanılarak açıklanabilir mi? (a5) Evet ise, bu faktörlerden hangileri metodun uyarlanması sırasında etkili oluyor? (b1) Metodun hangi kısımları uyarlanıyor? (b2) Bu kısımların varyasyonları var mı? (c1) Bu varyasyonların seçimi nasıl yapılıyor?

Soruların doğası gereği araştırma yaklaşımının bir parçası olarak araştırmacının aktif olarak projelerde bulunması, proje dökümanlarının analizi, mülakatların yapılması uygun veri toplama tekniği olarak görülmüştür. Mülakatlarda sorular açık uçlu, yarı-kapalı olup ilgili soruların açıklık kazanması için mülakatın içeriği serbest bırakılmıştır. Mülakat yapılan roller arasında koçlar (CMM ve DSDM uzmanları), yazılım geliştirme üst kurulu, proje liderleri, değişim yöneticisi, proje sponsorları, program geliştiriciler, son kullanıcılar, üst düzey sistem mimarları bulunmaktadır.

## 3 Araştırmanın Bulguları

Bu bildiriye kısıtlı sayfa sayısı nedeniyle daha önce değinilen tüm soruların detayına inmek yerine genel hatları ile okuyucu genel bilgiler sunulacaktır. Bazı soruların detaylı cevapları yayınlanmış maddelerde bulunabilir. Özellikle a1-...-a5 sorularının cevapları CMM bağlamında [6]'da sunulmuştur. Biz bu bildiriye özellikle b1(*Metodun hangi kısımları uyarlanıyor?*) ve b2 (*Bu kısımların varyasyonları var mı?*) ile ilgili sorulara cevap vereceğiz ve ayrıca c1 (*Bu varyasyonların seçimi nasıl yapılıyor?*) son bölümünde ele alınacaktır.

### 3.1 Metodun Daha Çok Uyarlandığı Bölümler

Araştırma sırasında eldeki zamanın kısıtlı olması nedeniyle metodun uyarlandığı tüm parçaların detaylı olarak incelenmesi mümkün olmadığı anlaşıldı. Bu nedenle pratisyenlere projenin verimliliği ve etkinliği açısından en önemli olan ve sıkça uyarlamak zorunda kaldıkları parçaların hangileri olduğu soruldu. Daha sonra bu kısımlar üzerinde odaklanılarak ilgili varyasyonları anlaşılmasına çalışıldı. Bu verilere geçmeden önce bu kısımların sınıflandırılması ile ilgili üç boyuttan bahsedeceğiz: metodun *felsefesi*, *çatısı*, *teknik ve araçları*.

*Metodun felsefesi* boyutu metodun prensiplerini içerir, *metodun çatısı* boyutu ise metodun önerdiği aşamalar, aktivitelerin detayı, sırası, roller, sunulacak ara ve son ürünleri kapsar. Gözlemlerimiz gösterdiği prensipler özellikle metod kullanıcıları için uyarlanması gereken en önemli ve sıkça karşılaşılan kısımlardır. Bu prensipler arasında *yinelemeli-artışlı (iterative-incremental) sistem geliştirme* ve *son kullanıcının sistem geliştirme sürecine aktif katılımı* prensipleri öne çıkmıştır. Bunun yanında *zaman kutuları*, *ihtiyaçların önem sırasına göre sıralanması tekniği (MoSCoW)*, *enterge test* en çok uyarlanan teknikler olarak tespit edilmiştir. Metod çatısı altında ise, şablonlara dayalı proje rollerinin uyarlandığı görülmüştür.

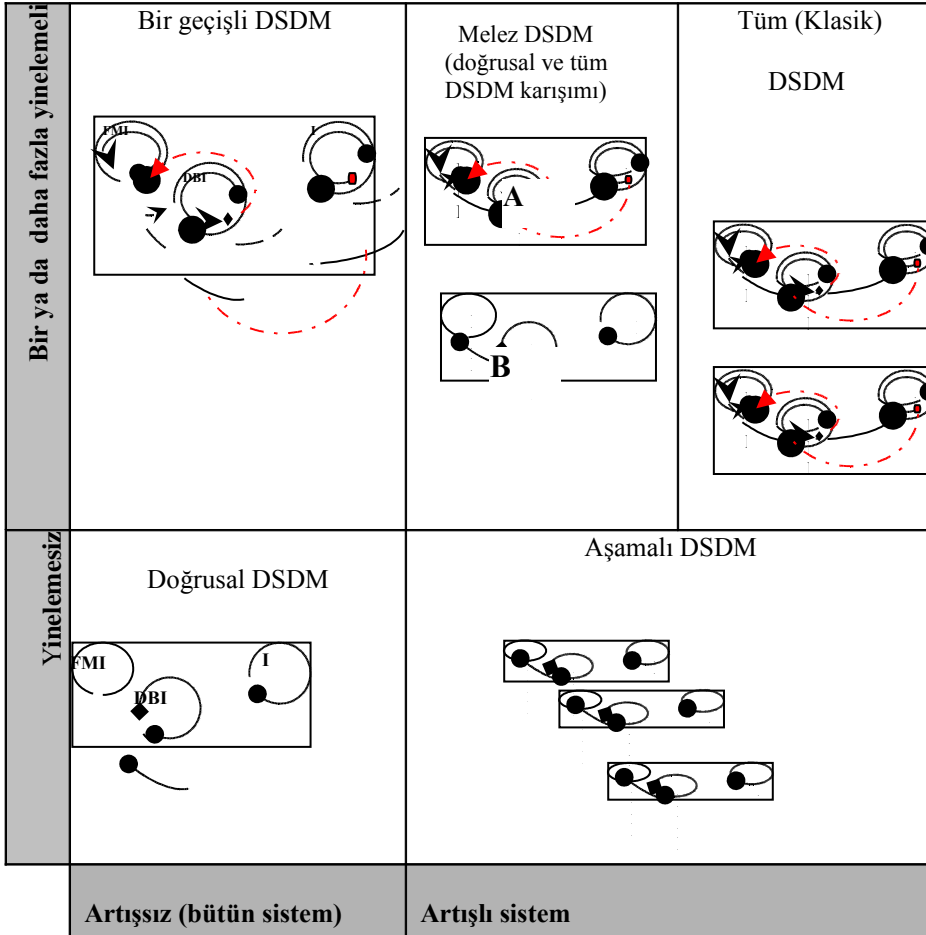
Metot kısımlarının kullanılması sırasında pratisyen genel olarak üç temel stratejiyi dikkate alır: (1) metodun mevcut kısmını değiştirmeden kullanma stratejisi, (2) değiştirme (uyarlama) stratejisi, tamamen başka bir parça ile değiştirme stratejisi. Şimdi bunları örneklemek üzere gerçekleşen bir uyarlama deneyimini aktaracağız.

Zaman kutulama tekniğinin temel amacı projenin bilinen bir zaman diliminde sistemin her parçasını zaman kutularına yerleştirerek ele almak ve bu yönüyle zamana, sonuca ya da ürüne odaklı bir sistem geliştirme yaklaşımı sağlamaktadır. Bu teknik metodun felsefesini de anlatması açısından önemlidir. Bir durum analizinde projenin genel olarak kaynaklarının (insan kaynakları) bulunduğu ancak proje sonlanma tarihinin kesin olarak belirlenmediği ortaya çıktı ve bu zaman kutulama tekniğinin gerek şartını sağlamadığı anlaşıldı. Bu durumda bu tekniğin bilinen şekli ile kullanılmayacağı belliydi. Öyleyse ya bu teknik kullanılmıyacak ve yerine başka bir teknik kullanılmalıydı ya da bir şekilde uyarlanacaktı. Bu tekniğin arkasındaki fikri baktığımızda önemli olan şartın bir şekilde kutulama yapılabilmesidir. Söz konusu projede insan kaynağı belliydi, projenin bu özelliği kullanarak kutulamayı proje insan kaynağına göre mümkün olduğu ortaya çıktı. Bir başka ifade ile zaman kutulaması yerine proje kaynakları kutulaması şeklinde bir teknik ortaya çıktı. Bunun gibi bir çok örnek gösterdi ki, pratisyenler gerçekte çoğu zaman yaratıcı olma kimliği ile metodun bir çok kısmını kendi durumlarına göre uyarlamaktadır.

### 3.2 Metod Uyarlanmasında Varyasyonları Belirlenmesi

Uyarlanan metod kısımların varyasyonlarını daha detaylı bir inceleme yapmak üzere yinelemeli-artışlı (iterative-incremental) sistem geliştirme prensibi üzerine odaklandık. Amacımız, bu prensibin nasıl hayata geçirildiği konusunda

detaylı bilgi almak idi. Şekil 1 bu bilgiyi yansıtmaktadır. İlgili kısımlardaki her kutu sistemin bir bölümünü ya da bir bütün olarak kendisini ifade etmektedir. Kutunun içindeki üç halka (FMI, DBI ve I) metodun üç temel sistem geliştirme aşamasını göstermektedir. Her bir halkadaki çember sayısı örnek olması açısından kaç tekrarın olması gerektiğini yansıtır. Örneğin, melez DSDM stratejisi sistemin bölümlendiğini ve bölümler için iki ayrı tekrarlama stratejisini yapılabileceğini gösterir. A-bölmesi yinelemeli, B-bölmesi tekrarsız strateji uygulanacağını gösterir. Böylelikle Şekil 1’de gösterildiği gibi en az beş ayrı varyasyon bulunmuştur. Burada dikkat edilmesi gereken nokta sistem geliştirme stratejisinin değişkenleri olarak *tekrar* ve *bölümleme* kriterleri kullanılmıştır. Bu kriterlere ek olarak örneğin (bölümlerin) *sıralama* kriteride kullanılarak sistem geliştirme varyasyonlarının artırılması mümkün olabilir.



Şekil 1. Yinelemeli-artışlı sistem geliştirme strateji varyasyonu

Bu varyasyonlar ortaya çıkarılması sonrasında önemli olan hangisinin hangi proje durumunda daha uygun olduğu belirlemektir. Diğer bir ifade ile şu soruya cevap aramaktaydık: Varyasyonu seçimi nasıl yapılmaktadır?

Bu sorunun cevabı proje durumunun nasıl analiz edildiği ile yakından ilgilidir. [6]’de açıklandığı gibi metod uzmanları kendi deneyimlerini kullanarak R/U analizi yapmayı sağlayan bir araç geliştirmişlerdir. Bu araçtaki 27 soru ile projenin karakteri ortaya çıkarılarak hangi varyasyonun eldeki karaktere uygun olduğu kararı verilir. Bu analiz sırasında kullanılan faktörlerin nasıl ilişkilendirdiğini ortaya çıkarmak için iki çalıştay düzenlenmiş ve bilgi modellemesi grafik teorisi kullanılarak ortaya çıkarılmıştır. Bu konunun detaylandırılması gelecekteki araştırma konularından biri olarak önümüzde durmaktadır.

#### 4 Sonuçlar ve Olası Araştırma Konuları

Giriş bölümünde bahsedildiği üzere *metodun proje durumuna göre uyarlanması* fikri pratikteki önemli bir ihtiyaca karşılık gelmektedir. Bu fikir gerçekte akademik dünyada on yıldan beri özel bir araştırma alanında (metot mühendisliği) tartışılmaktadır [7]. Bu alanda [9] tarafından yapılan çalışmalar -fizibilite çalışması- sözü edilen fikrin pratikte mümkün olabileceğini göstermiştir. Bu çalışmanın paralelinde, [10] tarafından Duruma dayalı Metod Mühendisliği (DMM) teorisi sunulmuştur. Bu teori özünde metodun parçalar halinde yapılandırılabilirliği ve olası proje senaryosuna bağlı olarak farklı parçaların (gerekirse farklı metod kaynakları kullanılarak) kullanılmasının mümkün

olduğunu ileri sürülmektedir. Buna ek olarak gerekli parçaların seçilmesi, birleştirilmesi gibi birçok adım teorik anlamda sunulmuş ve bunları desteklemek üzere ontoloji (ürün ve süreç parçalarının varlığı ve bilişim sistemleri referans modelleriyle ilişkilendirilmesi), metod mühendisliği dili ve ortamı (CAME-computer aided method engineering) sunulmuştur. Biz bu alanda yapılan çalışmaları *mühendislik yaklaşımı* olarak değerlendiriyoruz.

Metodların duruma uyarlanması konusunda diğer bir yaklaşım ise [12] tarafından sistem geliştirme sürecinde metodu projenin sosyal-örgüt olma özelliğinin bir parçası olarak kabul eder. Bu yaklaşımda metodun mühendislik açısından öne çıkarılan yapısal ve problem çözme özelliği yerine proje katılımcıları arasında iletişimi, uyumu sağlayan ve proje örgütünün politik, sosyal, kültürel ve daha da önemlisi insan faktörünün önemine dikkat çekilir. Bu yaklaşımdaki diğer önemli nokta metod parçalarının sistem geliştirme süreci sırasında pratisyen tarafından bir şekilde *yaratıldığı* görüşüdür. Biz her iki yaklaşımın birlikte kullanılarak metodun uyarlanması pratiğinin daha iyi anlaşılabilceğini düşünüyoruz. Şimdi bu konuda düşüncemizi pratikteki gözlemlerimizle irdeleyeceğiz.

Metodun söz konusu organizasyonda uyarlanması farklı *örgütsel seviyelerde* ve *kapsamlarda* ele alınarak açıklanabilir. Bu bildiri içinde sadece örgütsel seviyelere dikkat çekeceğiz. Kapsama bağlı uyarlamayla ilgili bilgi [8]'de bulunabilir. Örgütsel açıdan dört seviye dikkate alınabilir: *tüm organizasyon seviyesinde*, *uygulama alanı seviyesinde*, *proje seviyesinde* ve son olarak *birey seviyede* metod uyarlanması görülmektedir.

CMM'in temel amaçlarından birisi örgütsel seviyede standart sistem geliştirme süreçlerini mümkün kılmaktır [6]. Uygulama alanı seviyesindeki uyarlamanın temel amacı, geliştirilecek sistemlerin ortak özellikleri dikkate alınarak bir metodu proje seviyesindeki detaya inmeden uyarlamaktır. Örneğin, web tabanlı ve nesneye dayalı bir sistem geliştirme projesi ile ana bilgisayar üzerinde çalışan cobol programına dayalı bir sistem geliştirme projesinde metod kullanmak ile ilgili özellikler farklıdır ve benzer sistem geliştirme projeleri için benzer metod parçaları önerilebilir. Söz konusu organizasyonda projeden bağımsız uyarlama ile ilgili bu tür uyarlamanın yapıldığı ortaya çıkarılmıştır. Bu uyarlama seviyesinde 'rota haritası' olarak isimlendirilen metodun çatısı ve tekniklerinin uyarlandığı bir yapı kullanıldığı tespit edilmiştir. Örneğin, paket uygulamalar için hangi aşamaları, hangi akvitelerin ve ilgili aşamalarda hangi tekniklerin kullanılmasının uygun olacağı bir rota haritası düzenlenmiştir. Bu yapı yukarıda açıklandığı gibi DMM teorisindeki parçalara karşılık gelmektedir.

Öte yandan, proje seviyesindeki uyarlamada sosyal-örgüt yaklaşımı yapısal olmayan parçaların proje sırasında nasıl yaratıldığını anlamak açısından etkili olduğunu görmekteyiz.

Sonuç olarak metodun uyarlamasında iki farklı uygulama gözlemlenmiştir. *Statik uyarlama* olarak adlandırdığımız uyarlama, metodun projelerin bazı ortak özelliklerine dayanarak yapısal ve projeden bağımsız bir uyarlamayı anlatır. Diğer yandan *dinamik uyarlama* proje esnasında statik uyarlamadaki parçaların detaylanarak bir çok unsuru ele alarak duruma bağlı uyarlamayı anlatır. Her iki uyarlamanın bir şekilde pratikte gerçekleştiği düşünülürse bu tür deneyimlerin mümkün olduğu ölçüde şirketler tarafından kayda geçirilmesi ve birbirini tamamladığını düşündüğümüz iki ayrı açıdan ele alınması gereklidir.

Özellikle uyarlama konusundaki deneyimlerin toplanması ve paylaşılması için bir bilgi sistemi kullanılabilir. Yaptığımız ön çalışma böyle bir sistemin çok önemli iş yararlarını ortaya koymuştur. Özellikle proje riskinin ve süresinin azaltılması, kaynakların daha verimli kullanılması gibi bir çok yarar öngörülmektedir. Çalışmamızın bundan sonraki aşamasında tecrübeleri esas alan '*proje durumunu dayalı metod uyarlaması*' sürecini destekleyen bir karar destek sistemini geliştirmeyi amaçlıyoruz.

## Teşekkür

Çalışmanın yazımı aşamasında *Müge Taşkın*'a vermiş olduğu destek için teşekkür ederiz.

## Kaynakça

1. Sol, H. (1982). Simulation In Information Systems Development. (Dissertation). The University of Groningen.
2. Fitzgerald, B., Russo, N. L & O'Kane, T. (2003) Software Development Method Tailoring at Motorola. Communications of ACM, (46) 4.
3. Cockburn, A (2001). Agile Software Development. The Agile Software Development Series, Massachusetts: Addison Wesley Longman.
4. Martin, J. (1991). Rapid Application Development. Macmillan Publishing, New York.
5. DSDM Konsorsiyum, <http://www.dsdm.org>
6. Aydın, M.N. & Harmsen, F. (2002). Making a Method Work for a Project Situation in the Context of CMM. Proceedings, Lecture Notes in Computer Science, 2559: 158-171.
7. Kumar, K. & Welke, R. J. (1992). Methodology Engineering: A Proposal for Situation-Specific Methodology Construction. Challenges and Strategies for Research in Systems Development, Wiley & Sons Ltd.
8. Tolvanen, J.-P. (1998) Incremental Method Engineering with Modeling Tools: Theoretical Principles and Empirical

Evidence. (Dissertation) University of Jyväskylä, Finland.

9. Slooten, C. & Brinkkemper, S. (1993) A Method Engineering Approach to Information Systems Development. Proceedings of the IFIP WG 8.1, Elsevier Science Publishers B.V., North Holland.
10. Harmsen, F. (1997). Situational Method Engineering. (Doktora tezi). University of Twente.
11. Rolland, C. & Prakash, N. (1996). A Proposal For Context-Specific Method Engineering, Proceedings of the IFIP TC8, WG8.1/8.2 Working Conference on Method Engineering, Atlanta, USA.
12. Baskerville, R. & Stage, J. (2001). Accommodating Emergent Work Practice: Ethnographic Choice of Method Fragments. IS Development: The Social and Organizational Perspectives, New York.