

# IBM CLM Çözümleriyle Çevik Yazılım Süreçleri

Canberk Akduygu  
&  
Koray Okşar



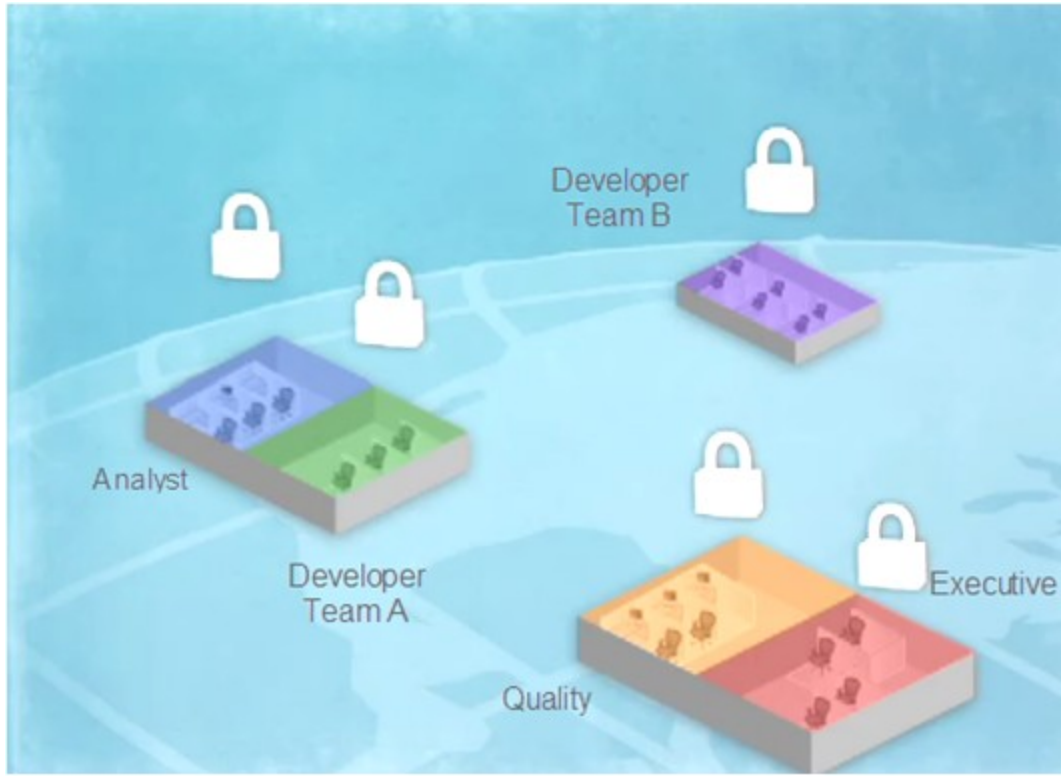
# Günümüzde Yazılım Geliştirme

- Proje takımları farklı bölgelerde çalışabilir ve iletişim eksikliği doğabilir
  - Gebze
  - Maltepe
  - Odakule
  - VS...
- IT Projeleri daha kompleks yapılar olmaya başladı, farklı teknolojiler dünyamıza girdi.
- Bu yüzden insan etkileşimi önemli hale geldi.



# Günümüzde Yazılım Geliştirme

- Bilgi havuzları takımlara göre farklılık gösteriyor
- Doğru bilgiyi,doğru zamanda paylaşmak kolay değil.
- Tool'lar arasındaki iletişim zayıf



# Sonuç

- Verimlilik düşer
- Dokümanlar zaman aşımına uğramış olabilir.
- Bütçe aşılır
- Kaynakları doğru yönlendirmek zorlaşır
- Projelerin zamanında bitmesi zorlaşır.

# Bu Duruma IBM Ne Çözüm Getiriyor?

IBM Collaborative Software Delivery konsepti için Jazz platformunu geliştirdi.

Bu ortam sayesinde bir Yazılım Projesinin Geliştirilme aşamaları

- İşbirlikçi
- Transparan
- Üretken

Takımların daha etkin çalışmasını sağlıyor ve Çeviklik kazandırıyor.

Bu konseptler **Collaborative Lifecycle Management** 1

# Yazılım Yaşam Döngüsü

## İşbirlikçi yazılım Yaşam Döngüsü



# IBM Rational Jazz Ürünleri



## Rational Requirements Composer

**Requirements solution for collaborative, fast-paced projects with high stakeholder involvement**



## Rational Team Concert

**A collaborative software development environment**



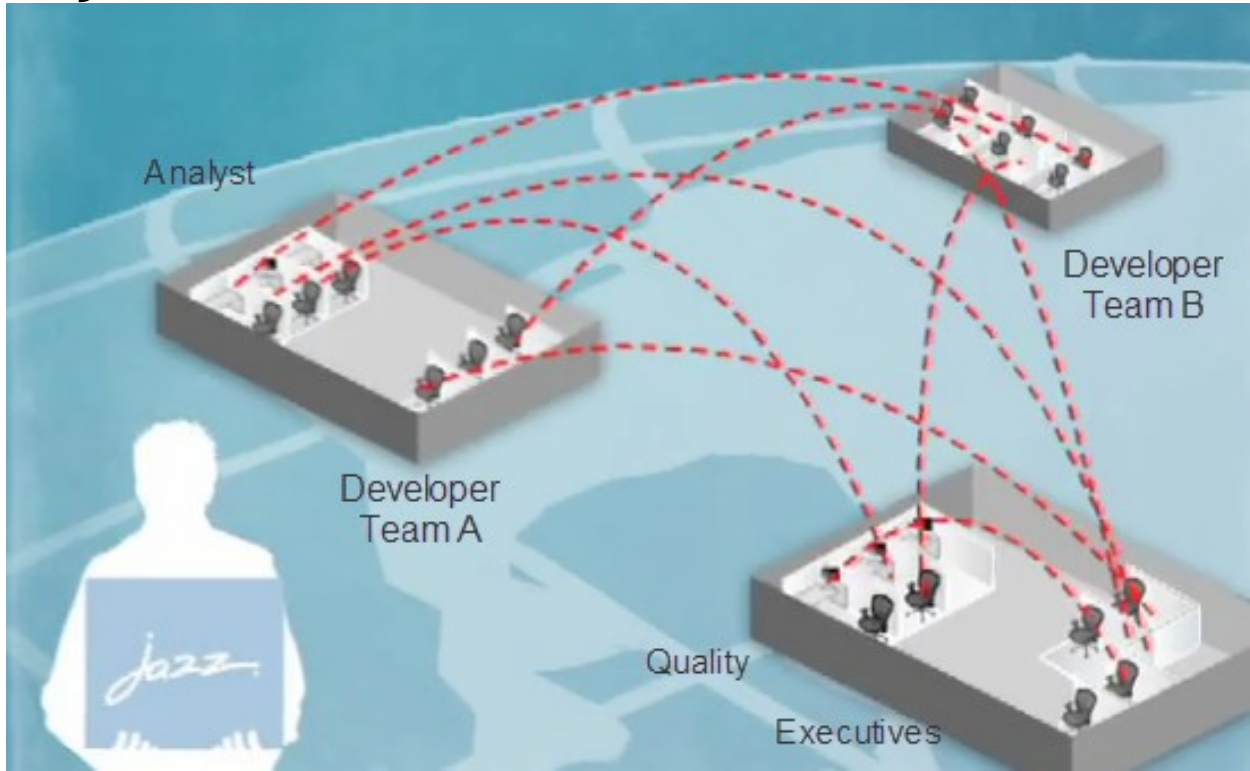
## Rational Quality Manager

**A collaborative ALM software quality delivery environment**



# Jazz Çözümlerinin Getirdiği Faydalar

- Tek bir Platform sayesinde takımları entegre hale getirir
- Bilginin akışını kolaylaştırır
- Üretim süreçleri kontrol edilebilir hala gelir
- Zaman yönetimi yapılabilir ve Proje planlaması kolaylaşır





# Jazz Platformu ile Nasıl Çevik Olunur?

- Jazz ürünleri Yazılım Geliştirme Süreçlerini destekleyen belli şablonlar ile gelir
- Agile & Scrum Süreçler bu şablonlara dahildir ve proje takımlarını Çevik Yazılım süreçlerinin içine sokar.

# Analist Bakış Açısı

- Analist ve İş Birimi:
  - Proje Gereksinimlerini toplar
  - İş Süreçlerini belirler
  - Ekran Prototipleri çıkarır

**REQUEST SHEET**

Çoklu işlem için onay, onaylandı işbu belgeyle

Maddelerden birini seçerek aşağıdaki alanları doldurunuz

Maddeler için işlemler

Maddeler için işlemler

Maddeler için işlemler

Maddeler için işlemler

**Şube Kasasına Para Verme**

Devir Tipi  
 Kullanıcı Kasasından Para Alma  Şube Kasasından Para Alma  Grup Merkezden Para Alma  
 Şube Kasasına Para Verme  Grup Merkezine Para Verme

İşlem Detay  
 Kullanıcı Kasası  ATM İşlem Tutarı 24,03 TL  
 Počet No A12345,A1222,A22222  
 ATM No ATM1006  ATM Kasası  Madeni Kasa  PLUS Kasası

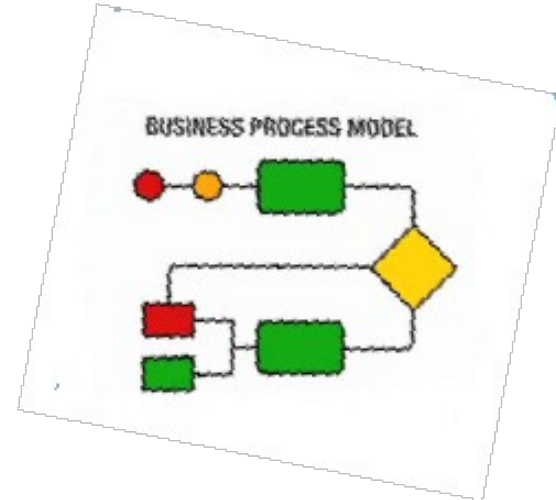
Kupür Sayımı

Banknot	SON YAPILAN KUPÜR SAYIMI			TOPLAM KUPÜR SAYIMI		
	Sistem Adedi	Maaş Adet	Tutar	Sistem Adedi	Maaş Adet	Tutar
200						
100						
50						
20						
10						
5						
1		22	22,00	22		22,00

Toplam Sayılan Para Tutarı: 24,03

Çoklu İşlem Listesi

Seç	İşlem Tutarı	Devir	Počet No
<input type="checkbox"/>	1.105,50	YTL	A1234567,A1222222



- Takımdaki diğer kişilerle Onay Süreci işleterek çıktılarını paylaşır.

# Analist Bakış Açısı

- Scrum Sürecin Analiste katkısı:
  - MosCoW analizi yapılmasını sağlar ve gereksinimleri “should, would, must” olarak kategoriler.
  - İterasyonlara göre yapılacak geliştirmelerin gereksinimlerini bölmelerini sağlar.
  - Kendi dashboard’un da gereksinimler ile geliştirme isteklerinin takibini yapar.
  - Kalite Ekibinin çıkaracağı Test Senaryoları için gereksinimler ile girdi sağlar.

☀ CRM Projesi gereksinim-workitem takibi Implemented By (6)	
📄	456: Activate the System requirement
✅	243: Activate the System requirement
📄	459: Checkout Flow Requirement
📄	245: Checkout Flow Requirement
📄	454: Login Requirement
📄	247: Login Requirement

# Rational Requirement Composer

- Dokümanlar için Onay Süreci
- MS Word dosyalarını içe aktarmak mümkün
- Üretilen dosyalar MS Word veya PDF olarak çıktıya dönüşür
- Use-Case, İş Akışı tanımlamayı sağlar
- Gerksinimlerin izlenebilir
- Takım ile bilgi paylaşmak kolaydır
- Proje Planı izlenir

# Geliştirici Bakış Açısı

- Takım Lideri
  - Iterasyon
  - Release Plan
  - Kişisel İş Listesiçıkartır.
- Yazılım Geliştiriciler
  - Kaynak Kod Yönetimi ve Build Sistemini kullanır
  - Doğru dosyayı doğru Stream'e aktarır
  - Gereksinimlere bağlı geliştirme listesini takip eder, eksikleri takip eder.
  - Test ekibinden gelen hataları düzeltir.

# Geliştirici Bakış Açısı

## Scrum Sürecin Yazılımcıya katkısı:

□ Iterasyon, product backlog ve geliştirme listesini tutar

□ Burndown Charts vs raporlar ile yazılım projesinin metriklerini toplar

□ Her yazılımcıya bir Scrum Board sağlayarak etkileşimli bir yönetim ekranı sağlanır

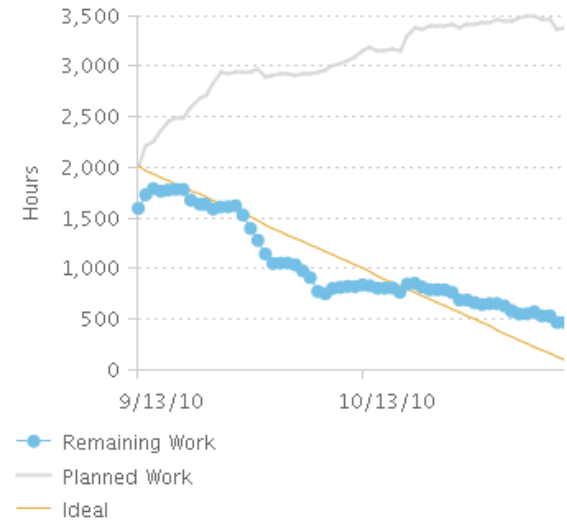
□ Anlık Mesajlaşma ve hem web client hem de IDE'lerde çalışan uyarı mesajları ile kullanıcıyı değişikliklere karşı uyarır.

Product Backlog

Team Area: Havannah | Iteration: Release 1.0 (6/1/09 - 8/31/09) | 0 Closed | 19 Open

As a player I can play against a weak engine that recognizes rings	9
As a player I'd like to be able to use the system to play against another human on my computer	10
As a player I can play against a weak engine that recognizes bridges	11
As a player I want the computer to recognize a winning shape	12
As a player I can play against a weak engine that recognizes forks	13
As a player I'd like to be able to choose between a wooden board and	14

Burndown



# Rational Team Concert

- Takım çalışması
  - Entegre Release/İş Planlama,Raporlama,Proje Yönetimi ve Kaynak Kod/Build yönetim özellikleri ile Proje Planı eksiksiz takip edilebilir
- Çevik Geliştirme
  - Hazır çevir Süreçler içerir
  - Süreçlerin değiştirilmesine olanak sunar
- Yönetimi Otomatize Eder
  - Projelerin statülerini izleyerek metrikler toplar



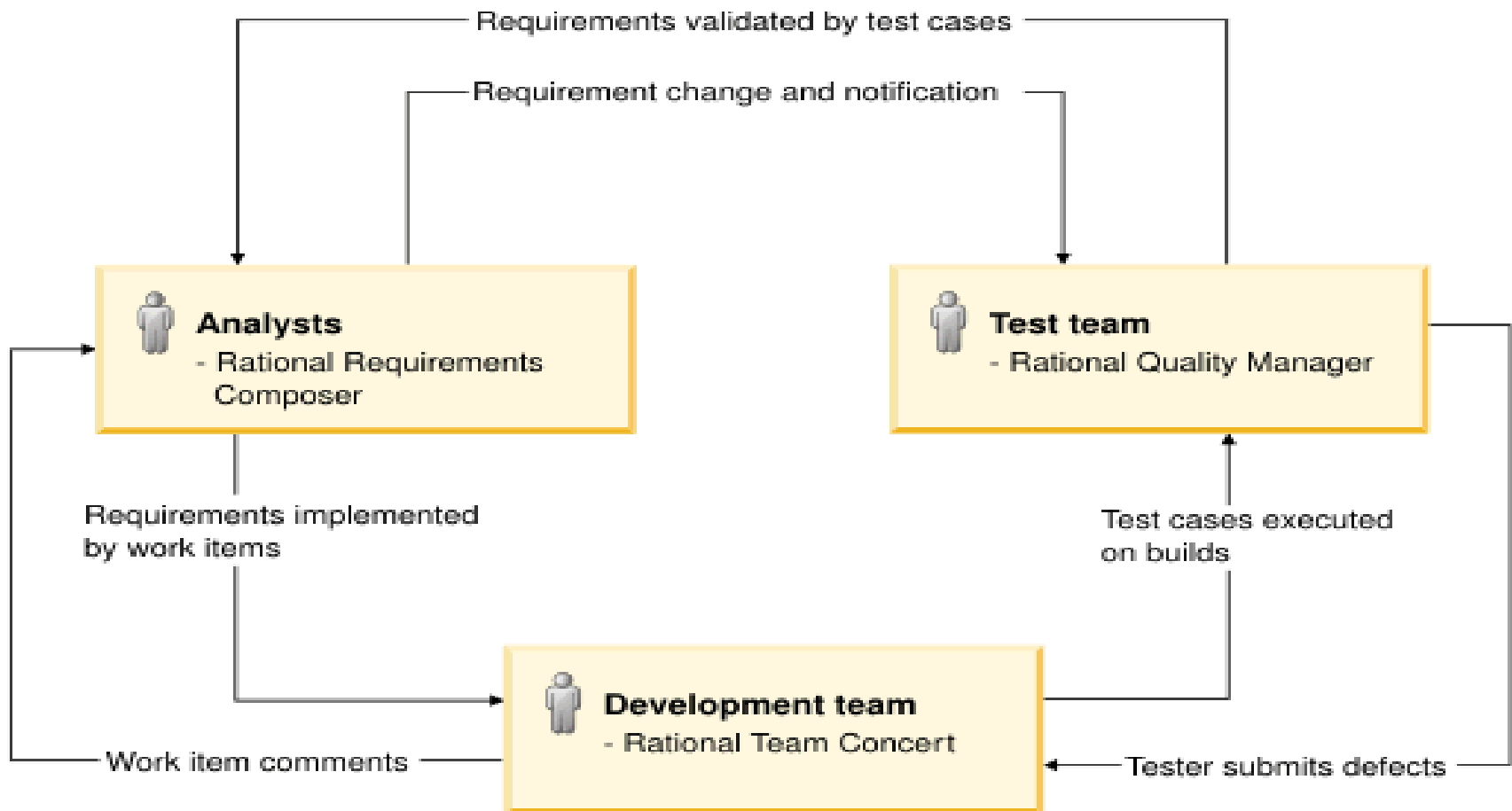
# Testçinin Bakış Açısı

- Test Ekibi Test Planı üstünde çalışır
  - Gereksinimler Analiz ekibinden gelir
  - Iterasyonlar belli olduğu için hangi testleri koşacağını bilir
  - Build'ları takip ederek testleri otomatize eder
  - Hangi gereksinimin değiştiği bilgisini takip eder.
  - Buldukları Yazılım Hatalarını yazılımcıya yollar.
- Test Koşmak
  - Hızlı
  - Etkili
  - Otomatize hale gelir.

# Rational Quality Manager

- Test Planları, Test Senaryosu ve Script'leri yaratılır
- Test verisi oluşturulur.
- Fonksiyonel test ve Performans test entegrasyonu ile testler otomatize edilir
- Test ortamları tanımlanır.
- Test sonuçlarına göre geliştiriciye hata kaydı açılır ve hata yaşam döngüsü başlatılır.

# Entegre Bakış Açısı



## Yönetim Bakış Açısı

- Manuel hazırlanan durum raporlarına ihtiyaç yoktur.
- Anlık/Trend rapor desteği ile hızlı metrik ölçümü



- **Telekom Sektörün**

- **Eski Durum:**

- Farklı yazılım dilleri var.
    - Kod havuzu ayrı
    - Build yönetimi farklı ekiplere bağlı
    - Gereksinim toplama süreci MS Tool'ları ile yürüyor.

- **Şimdi:**

- Tüm Diller tek bir kod havuzunda
    - Release Management yapan bir kişi var.
    - Gereksinimler tek yerde toplanıyor. İç müşteri de katılımda bulunuyor.

## • Finans Sektörü

### ▫ Eski Durum:

- Proje bazlı gereksinim toplanması
- Bilginin kişiye bağlı kalması
- Kod havuzunun farklı yerlerden idare edilmesi
- Testlerin MS Tool'ları ve farklı diğer toollar ile yürümesi

### ▫ Şimdi

- Uygulama Bazlı gereksinim toplanması
- Kod havuzu tek hale geldi. Diğer tool'lar ile entegre edildi
- Test ekibi Analiz ekibiyle birlikte çalışabiliyor.

- E-Ticaret Sitesi

- Eski Durum:

- Tek bir uygulama var + Onlarca ufak proje
    - Farklı dillerde geliştirme ortamı
    - Geliştirme-Test-Canlı ortamların birbirinden farklı olmaması.

- Şimdi

- Uygulamaları yaşatarak projelerin bu uygulamaya bağlantısı takip edilebiliyor.
    - Hangi ortamda hangi kod var izlenebiliyor.



## • 3rd Party Software Vendor

### ▫ Eski Durum:

- Onlarca farklı uygulamaları var.
- Tek yazılım dili.
- Proje Planlaması MSProject'de
- Analiz dokümanları farklı ortamlarda

### ▫ Şimdi

- Proje planı ve kişisel iş listesi entegre çalışıyor.
- Geliştirme ve gereksinimler birbirine bağlı.
- Kodlar ile gereksinimler bağlı.
- Çıkan hatalar gereksinime bağlanabiliyor.
- Trend raporu alınıyor.