



PHP & Scrum: das Dream Team

EntwicklerTage | 15. Feb. 2006 | Björn Schotte



Björn Schotte - „PHP Pionier“

- 1998/1999: PHP-Center.de
- 2000: weltweit erster PHP Kongress
- 2001: Chefredakteur PHP Magazin
- 2001: International PHP Conference
- 2001: Mitgründer/Initiator von ThinkPHP
- 2004: Verschmelzung zur Mayflower GmbH



ThinkPHP – DER PHP-Spezialist in DACH



- Premium Softwareentwicklung seit 2001
- PHP Trainings für mehr als 300 Personen aus Großunternehmen
- PHP Anwendungssupport – <http://www.thinkphp.de/support/>
 - Enterprise PHP Support
 - Garantierte Reaktionszeiten bis 2 Std.
 - Basic, Professional, Gold
- Förderung der PHP Community und PHP an sich
- „ThinkPHP by Mayflower“ – Division der Mayflower GmbH

- mehr als 30 Mitarbeiter
- „ThinkPHP – by Mayflower“
- Kunden, meist Top50 Unternehmen
 - Siemens COM
 - Vaillant Group
 - Telefónica Deutschland
 - Landeshauptstadt München
 - HypoVereinsbank
 - Sixt AG
 - citiworks AG
 - Stadtwerke München (SWM)
 - Bundeskriminalamt
 - E.ON
 - ...
- 4 GF-Gesellschafter (Gründer von ThinkPHP)
- 2 Standorte (München und Würzburg)
- Bekanntes Renommée in der PHP- und OpenSource Community
- Kompetenz weit über PHP hinaus (High Performance Webarchitekturen, .NET, WebServices, SAP & Co.)
- Projektkompetenz für Großprojekte im Mannjahresbereich mit PHP (HVB, Vaillant, Sixt, Telefónica)
- OpenSource Software Hersteller (z.B. PHPProjekt)

Agenda

- Herausforderung in Großprojekten
- Entwicklungsmethoden (XP, Scrum, Prototyping, ...)
- Tiefer Einstieg in Scrum
 - die Theorie
 - Scrum in der Praxis



Die Herausforderung

- Kunden wollen für Webprojekte nicht mehr siebenstellige Summen hinlegen, und nur 3 Funktionalitäten dafür rausbekommen
- Kunden wollen das Webprojekt am liebsten gestern auf dem Markt haben
- Kunden wollen, dass das Webprojekt sich den sich täglich verändernden Bedingungen am Markt schnell anpaßt

Die mögliche Lösung

- Verwendung von PHP („LAMP-Stack“) als leichtgewichtige Lösung
- Auch für komplexe Projekte geeignet („enterprise ready“)
- Evolutionäre und am Prototypen ausgerichtete Entwicklung

■ READINESS APPLICATION DEVELOPMENT

Agenda

- Entwicklungsmethoden, best practice:
 - Prototypenentwicklung
- Entwicklungsmethoden, „wissenschaftlich“:
 - eXtreme Programming
 - Agile Programmierung
 - Scrum



Entwicklungsmethoden: Vorwort

- Vieles, was wir auflisten, kommt uns bekannt vor. Wir wenden es bereits täglich unbewußt an



Entwicklungsmethoden: Prototypen I

- Entwicklung von Prototypen
- Innerhalb von 2-3 Tagen erste sichtbare Ergebnisse
- Weiterentwicklung anhand des Prototypen
- Iterative Herangehensweise



Entwicklungsmethoden: Prototypen II

- Schrittweises Annähern an das Ergebnis zusammen mit dem Kunden (Kundenfeedbacks)
- Eine Parallelität entsteht: während der Entwicklung erreichen uns die Change Requests des Kunden
- Tägliche Verbesserungsmöglichkeiten
- Steigerung der Kundenzufriedenheit: Kunden wissen oft nicht genau, wie das Endprodukt aussehen soll; DevTeam kann sehr viel leichter auf geänderte Marktbedingungen beim Kunden reagieren

eXtreme Programming

- Pair programming
- Iterative Entwicklungszyklen
- Test first, code second
- Customer on-site
- Run cards



eXtreme Programming: Pair Programming

- Pair programming: 4 Augen sehen mehr als 2
- Hat man oft schon erlebt: man findet lange Zeit einen Fehler nicht, bis der Kollege vorbei kommt und den Fehler anzeigt
- Büro-Ausstattung darauf ausrichten
- Pair programming nicht immer möglich, auch aus Budgetgründen des Kunden



eXtreme Programming: Customer on-site

- Customer on-site: Kunde ist direkt beim Projektteam
- Durchaus sinnvoll, gerade bei Entwicklung von fachlichen Anwendungen (Beispiel: Gisyweb für E.ON)
- Aus Kundensicht nicht immer möglich, stattdessen monatliche Meetings, Telefonate und Austausch von E-Mails



eXtreme Programming: Run Cards

- Benutzung von kleinen Karteikarten
- Definition von Funktionsblöcken auf den Karten mit anschließender Priorisierung
- „handgeschriebene“ Karten
- Kunde kann einen Stapel Karten für die nächste Iteration bestimmen
- Realität: nicht immer nutzbar, stattdessen via Wiki oder z.B. Mantis zum Festhalten von Change Requests und Priorisierung

eXtreme Programming: Iterative Entwicklung

- Idee: in möglichst kleinen Teilschritten, einem Inkrement, einen sichtbaren Fortschritt ermöglichen
- Literatur spricht von 1 Inkrement = 30 Tage
- Im PHP-Bereich wird man Inkremente schon im Bereich von 1-2 Tagen oder 1 Woche definieren



■ „Mehr wert:“

- Individuen und Interaktion
- Funktionierende Software
- Kundenzusammenarbeit
- Auf Veränderungen reagieren

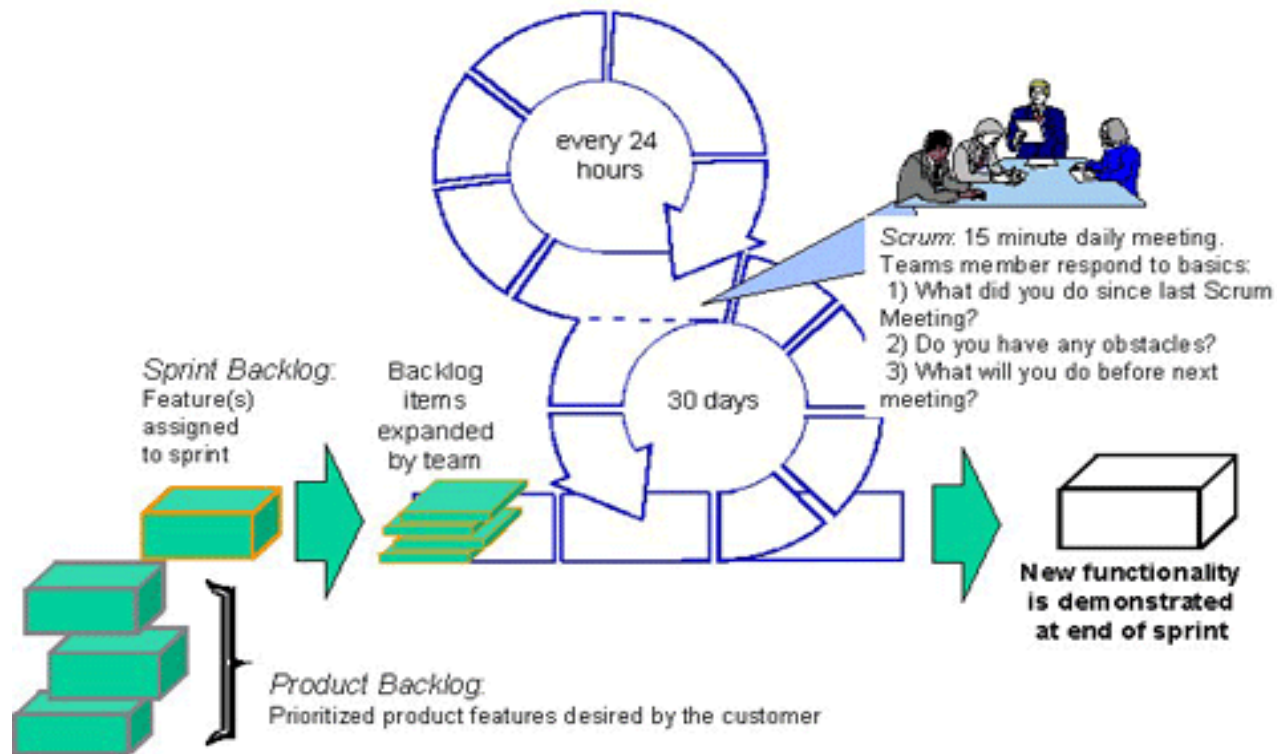
■ „Weniger wert“:

- Prozesse und Tools
- Ausführliche Dokumentation
- Vertragsverhandlungen
- Einen Plan exakt verfolgen

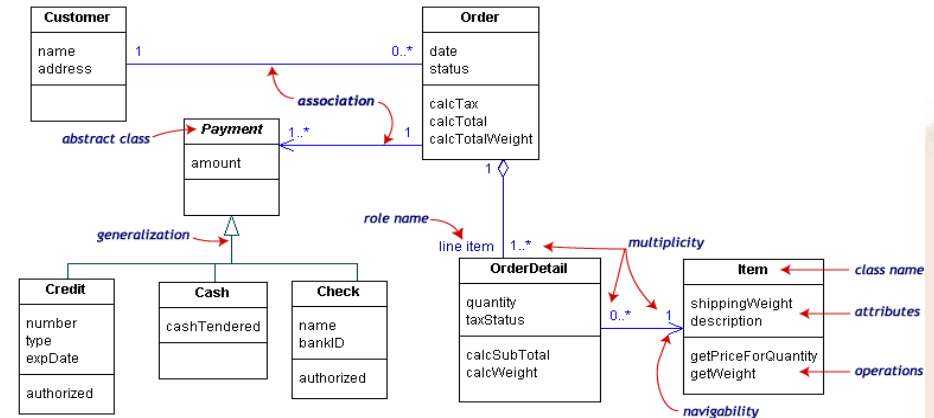
- Konzentriere dich darauf, dass Version 0.1 funktioniert.
- Versuche nicht von vornherein, auf alle Eventualitäten zu achten – du wirst scheitern
- Arbeite dich in evolutionären Schritten voran
- Reagiere flexibel auf Änderungen
- Verändere in kleinen Schritten/Iterationen
- Werde dir bewußt, welche Tools du wirklich benötigst

Scrum

■ <http://www.controlchaos.com/>



- Unified Modelling Language
- Standard der OMG (Object Management Group)
- <http://www.uml.org/>
- Strukturierte Herangehensweise via Ablaufdiagramme, Planung von Klassen und ihren Methoden
- Automatische Codeerzeugung
- Tools wie ArgoUML, UML2PHP etc.
- PHPs USP ist: get things done fast. Ist UML hierfür geeignet?



PHP will ...

- ... eine Sprache sein für „getting things done fast“
- ... eine Sprache sein, mit der man auch komplexe Dinge auf einfachem Weg erledigen kann (z.B. via SimpleXML extension)
- „In Java kümmert man sich zu 90% um die Struktur, zu 10% um die Problemlösung. In PHP ist es genau umgekehrt.“ (Rasmus Lerdorf)

Der Kunde will ...

- Kunden wollen oft innerhalb von 3-4 Stunden oder 1-2 Tagen anfassbare Ergebnisse sehen
- Kunden wissen bei großen Projekten oft gar nicht, was sie so genau wollen, es existieren keine umfangreichen Spezifikationen
- Iterative Herangehensweise anhand von Prototypen



Tieferer Einstieg: Scrum

■ Problematik:

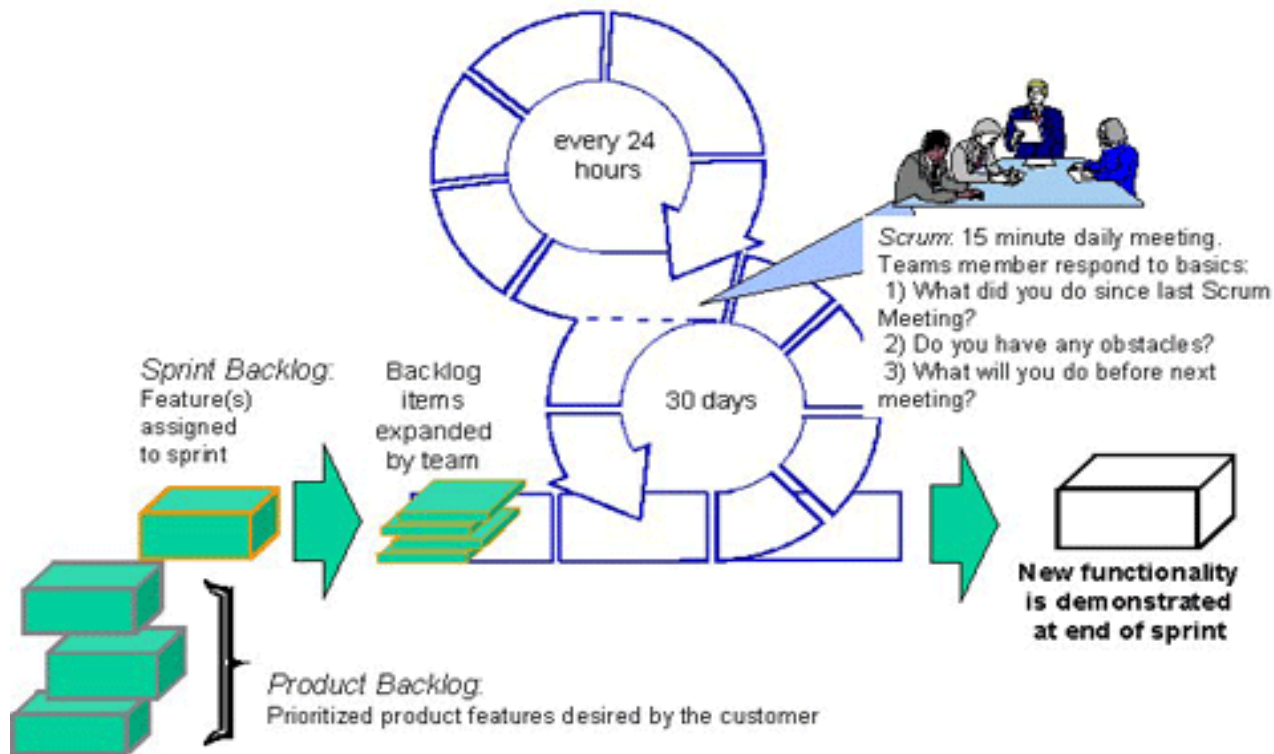
- sich schnell ändernde Bedingungen
- Kunden wissen oft gar nicht ganz genau, was sie wollen
- daher inkrementelle Herangehensweise
- dennoch oft im Kontext von Festpreisangeboten

■ Lösung:

- Vertrauen schaffen
- Kommunikation ist wichtiger als Verträge oder Dokumentation
- schnelle Reaktionszeiten



Scrum: die Methodik

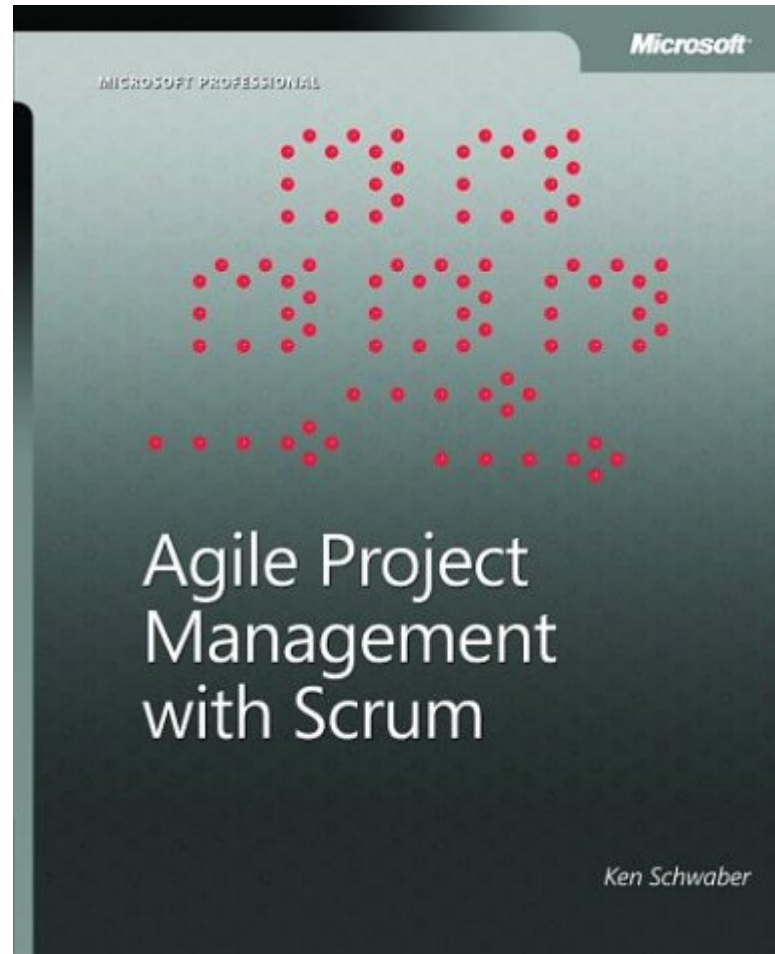
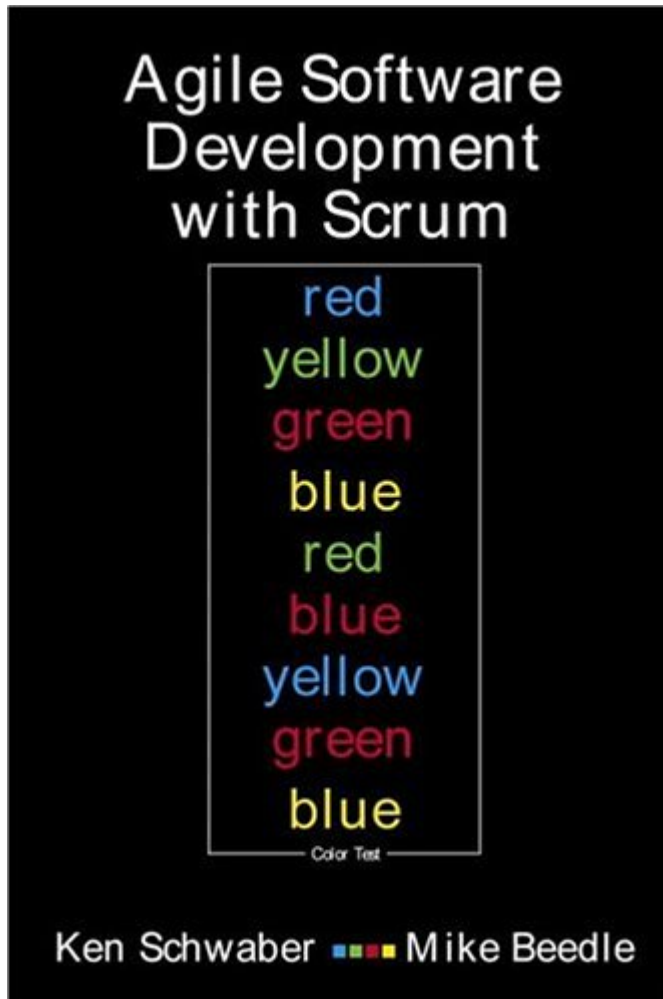


Tieferer Einstieg: Scrum

- „Scrum“ ursprünglich von Rugby her kommend
- Begrifflichkeiten:
 - Product Owner
 - Product Backlog
 - Scrum Master
 - Sprint
 - Sprint Planning Meeting
 - Sprint Backlog
 - Daily Scrum
 - Sprint Review Meeting



Scrum: Lesefutter



Scrum: Product Owner

- In der Regel Ihr Kunde
- definiert mit Ihnen als Dienstleister die Funktionalitäten des Systems
- zum Beispiel am besten via Excel = Product Backlog
- In herkömmlichen Projekten ähnlich der Requirements Specification, jedoch hier oft/meist stichpunktartig



Scrum: Product Backlog

- Zum Beispiel in Excel gepflegte Liste
- Tasks („geschäftskritisch“), die der Product Owner insgesamt erledigt haben möchte, inkl. Priorität
- Ergänzt mit Aufwand-Abschätzung durch Developer, ~h

| Date | Feature | Priority (1..n) | Est | Realisation by | Task broken down |
|------------|--|-----------------|-----|----------------|------------------|
| 40-18-2005 | kwk-> mbar | 1 | 1 | Alex | Björn |
| 40-19-2005 | erste Parameterliste in Überschrift mitziehen (2,3 wie bisher) | 1 | 8 | Alex | |
| 40-19-2005 | Eigenschaften in rechte-Box | 1 | 6 | Alex | |
| 40-19-2005 | Bug: im IE falscher Font/nicht fett | 1 | 2 | Alex, Olli | |
| 40-19-2005 | Bug: falsche Einrückung bei erstem Link | 1 | 4 | Alex, Olli | |
| 40-19-2005 | Geräteansicht: Details in rechte-Box | 1 | 0,5 | Alex | |
| 40-19-2005 | Geräteansicht: "31 result(s)" weg | 1 | 0,5 | Alex | |
| 40-19-2005 | Details: "Datenpunkte pro Seite" als Einstellung pro Gerät | 1 | 16 | Alex | |
| 40-19-2005 | "Checkliste": keine Unterstriche/Bullets, gleiches Aussehen-w | 1 | 1 | Olli | |
| 40-19-2005 | Bug: Boxüberschrift fehlt wenn kein Device selektiert | 1 | 2 | Alex | |
| 40-19-2005 | Loginbox: Statustext in "My Profile" | 1 | 2 | Alex | |
| 40-19-2005 | Objektübersicht: Wenn kein erster Boiler da, dann erster Regl | 1 | 0,5 | Björn | |
| 40-19-2005 | Objektübersicht: Wenn kein Regler, dann Controller (bei comp | 1 | 0,5 | Björn | |
| 40-19-2005 | Display-Chart: Fenster etwas länger (zweizeilige überschrift) | 1 | 0,5 | Alex | |
| 40-19-2005 | winkelmoser: Logsession 5 kann nicht geplottet werden | 1 | 8 | Martin | |
| 40-19-2005 | Graphics SD Bug: Template vom ersten Boiler ziehen | 1 | 2 | Alex | |
| 40-24-2005 | Mantis-1581: Switchpos besser positionieren | 1 | 1 | Alex | |
| 40-24-2005 | Mantis-1615: SD-Graphics_Tpl-Keywords-noch-unklar | 1 | 3 | Alex | |
| 40-25-2005 | Mantis-1633: EF-Message-not-displayed, Tickets-not-closed | 1 | 4 | Alex | |
| 40-25-2005 | Mantis-1643: Anzahl-DPs in boileroverview falsch | 1 | 4 | Alex | |
| 40-25-2005 | Mantis-1594: Logsession lässt sich nicht abbrechen | 1 | 2 | Martin | |
| 40-26-2005 | Eigentümer-Link muss auch in Messages funktionieren | 1 | 3 | Alex | |

Scrum: Scrum Master

- auf Dienstleister-Seite
- der Leitwolf des Developer-Teams
- weniger Projektmanager im Sinne von Bottom-Up TODO, sondern eher „Enabler“
- hat dafür Sorge zu tragen, dass das Team in Ruhe arbeiten kann
- leitet die *Daily Scrums*
- definiert mit dem Team die Tasks für den *Sprint*



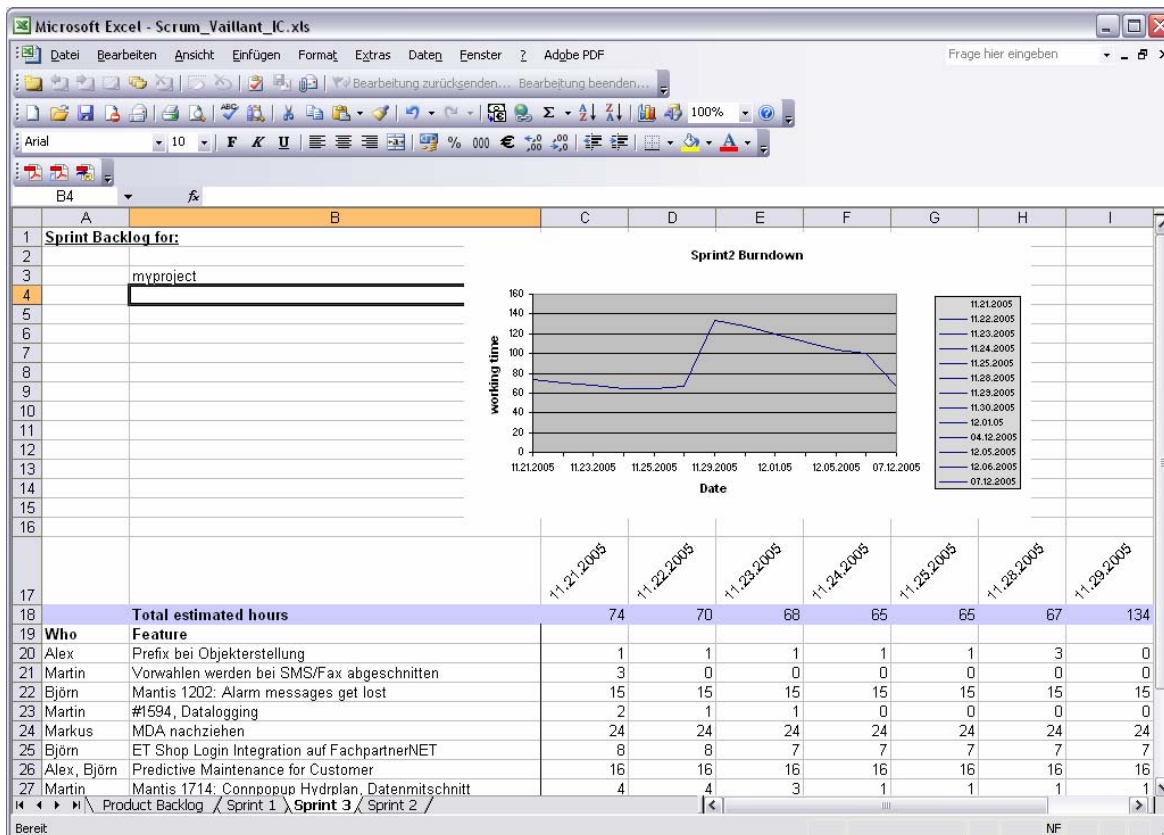
Scrum: Sprint

- ein *time-boxed* Zeitraum, in dem entwickelt wird
- Features stehen via *Sprint Planning Meeting* fest
- Developer Team konzentriert sich ausschließlich auf die Erledigung dieser festgelegten Tasks
- It. Literatur Zeitraum von 30 Tagen, kann jedoch auch kürzer (5, 10) oder länger (40) sein
- *Herausforderung*: Changes während eines Sprints!



Scrum: Sprint Planning Meeting (1)

- Zusammen mit dem Scrum Master + evtl. Product Owner wird definiert, welche Tasks im kommenden Sprint zu erledigen sind



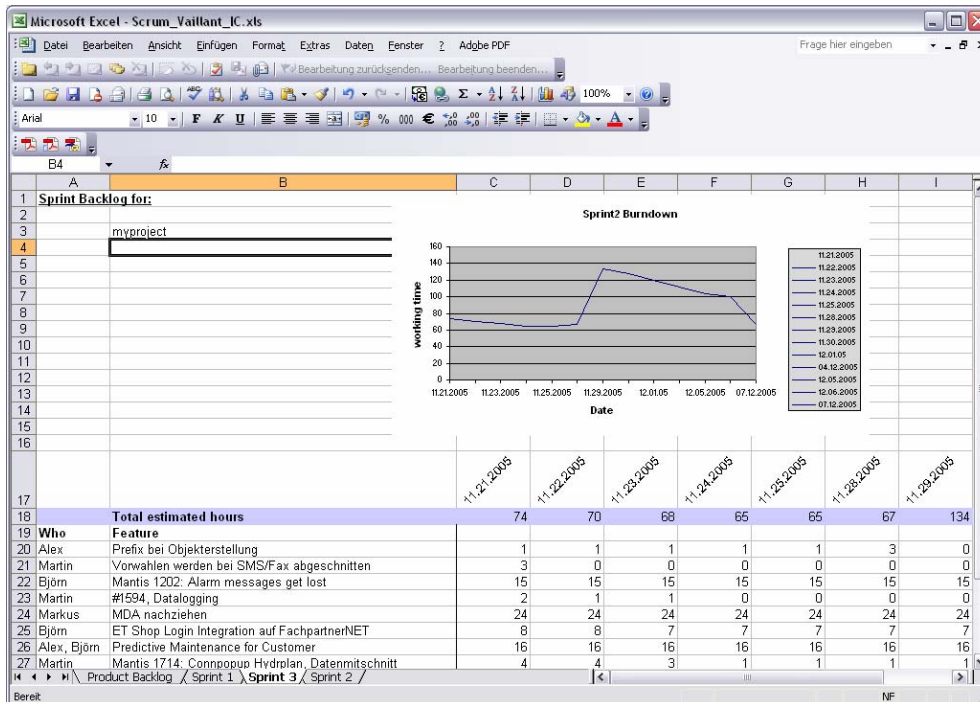
Scrum: Sprint Planning Meeting (2)

- Definition der Tasks aus dem Product Backlog
- via Daily Scrum Aktualisierung der Stundenanzahl
- Motivation fürs Team: „*Sprint Burndown Chart*“ – Ziel: auf 0h Restzeit innerhalb des Sprints zu kommen



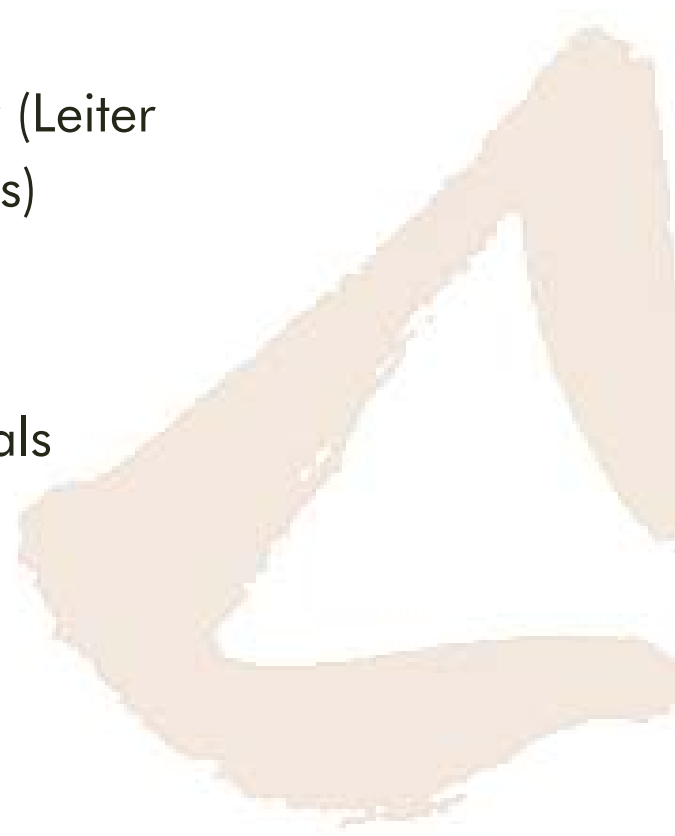
Scrum: Sprint Backlog

- Siehe Screenshot, spalten-/zeilenartige Auflistung der Restzeiten



Scrum: Daily Scrum (1)

- Kommunikation wichtiger als Verträge oder dicke Pflichtenhefte
- daher: **tägliches Daily Scrum** mit dem Scrum Master (Leiter des Meetings) sowie den beteiligten Entwicklern (Pigs)
- Wer zu spät zum Meeting kommt: z.B. 5,-- EUR in gemeinsame Kasse
- freiwillige Anwesenheit anderer (Management etc.) als Chickens (kein Rederecht)
- maximal 15 Minuten Dauer!



Scrum: Daily Scrum (2)

- Reihum 3 Fragen beantworten:
 - Was hast du seit dem letzten Daily Scrum gemacht?
 - Was wirst du bis zum nächsten Meeting tun?
 - Was sind deine Impediments?
- wahlweise (wenn z.B. nur Excel zur Hand) Aktualisierung der Restzeiten pro Task für diesen Tag
- Nach 15 Minuten striktes Beenden des Meetings
- Techie-Diskussionen auf später verlegen



Scrum: Daily Scrum (3)

- Es empfiehlt sich, das Meeting morgens zu machen, Vorteil: direkt alle auf dem aktuellen Stand
- 5x hintereinander Sprint Task auf der gleichen h-Anzahl?
 - Indikator, dass der Scrum Master mit dem Developer abseits reden sollte
- Benefits:
 - *Synchronisation* aller Developer über den aktuellen Stand
 - Scrum Master trägt dafür Sorge, die *Impediments aus dem Weg zu räumen*
 - Kommunikation ist alles!

Scrum: Sprint Review Meeting

- Nach Abschluss des Sprints steht ein fertig laufendes Inkrement der Applikation zur Verfügung
- Im Review Meeting zusammen mit dem Product Owner schauen, ob soweit alles passt
- Feedback auch mit den Entwicklern, wie der Sprint gelaufen ist
- Ggf. Verbesserungen für den nächsten Sprint einarbeiten
- Durchführung des Review Meetings problematisch bei kurzen Sprints (z.B. 5 Tage) und ständigen Changes während eines Sprints

Scrum: soweit die Theorie ...

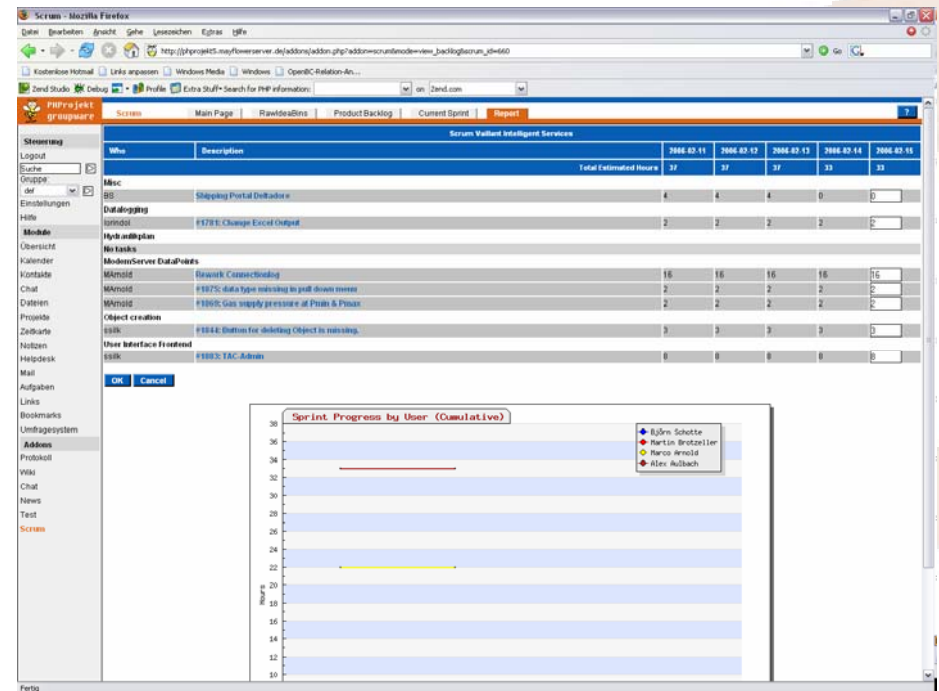
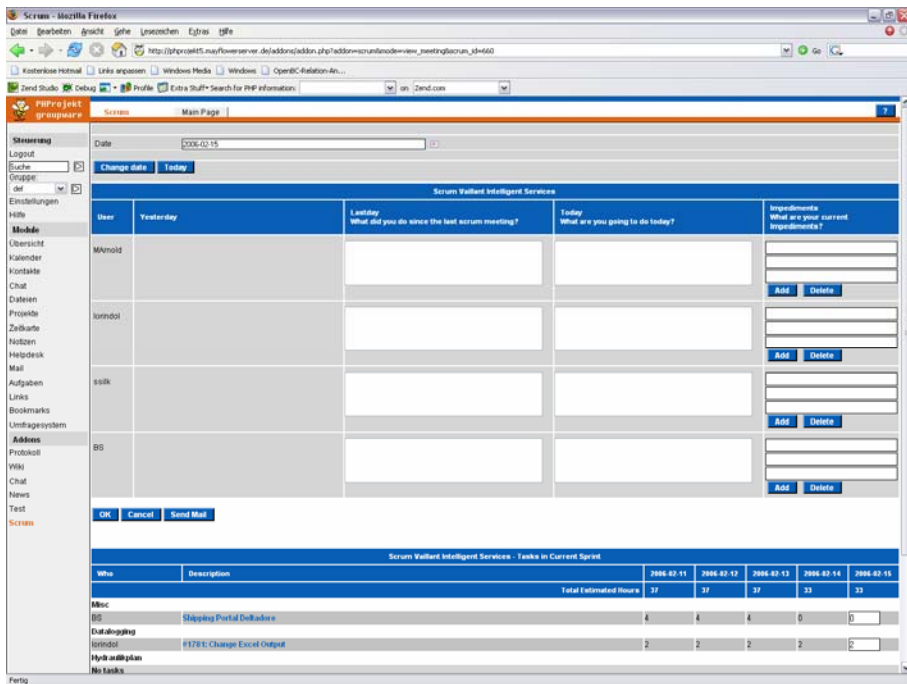
- Erfahrungs- und Praxisberichte:
 - Scrum in Projekten mit hohen Change Request Raten
 - Einarbeitung von Changes während eines Sprints
 - Diskussion!

- Bessere Softwareunterstützung?



Softwareunterstützung

- www.scrumalliance.org
- Wenig webbasierte Tools verfügbar – Excel tut's auch
- aus dem Eigenbedarf entwickelt, für PHPProjekt: Scrum Modul (GPL)





Vielen Dank für Ihre Aufmerksamkeit

Björn Schotte

Mayflower GmbH

Pleichertorstr. 2

97070 Würzburg

+49 (931) 35 9 65 - 0

schotte@mayflower.de

