



Apcon Workplace Solutions
Member of itelligence

Extreme Programming (XP) Integrativ?

Stefan Rook, roock@jwam.de

Universität Hamburg &
APCON Workplace Solutions GmbH
Vogt-Kölln-Strasse 30
22527 Hamburg
Germany




Apcon Workplace Solutions
Member of itelligence

Kontext & Erfahrungshintergrund

- XP ist noch jung und nimmt für sich in Anspruch, große Produktivitätspotenziale zu bieten.
- Erste Experimente mit XP an der Uni Hamburg Anfang 1999.
- Seit 2000 Einsatz von XP im WPS-Firmenumfeld.

- Projekte in unterschiedlichen Domänen
- Breite Spanne an Projektgrößen: 4 PM in 11 Wochen bis zu ca. 75 PJ in 3 Jahren)
- interaktive Multi-Channeling-Anwendungen
- Individual-Lösungen und Produktentwicklung
- Rahmenwerke (JWAM)

itelligence
Apcon Workplace Solutions
Member of itelligence

Hauptthese (1)

- XP *wirkt*
 - neu (eXtreme)
 - entwickler-zentriert (Programming)
 - ungeplant und chaotisch
 - unkontrolliert und unkontrollierbar
 - risikoreich
 - „Strukturierter Anarchismus“
- Das zieht Entwickler an und schreckt Manager ab.

itelligence
Apcon Workplace Solutions
Member of itelligence

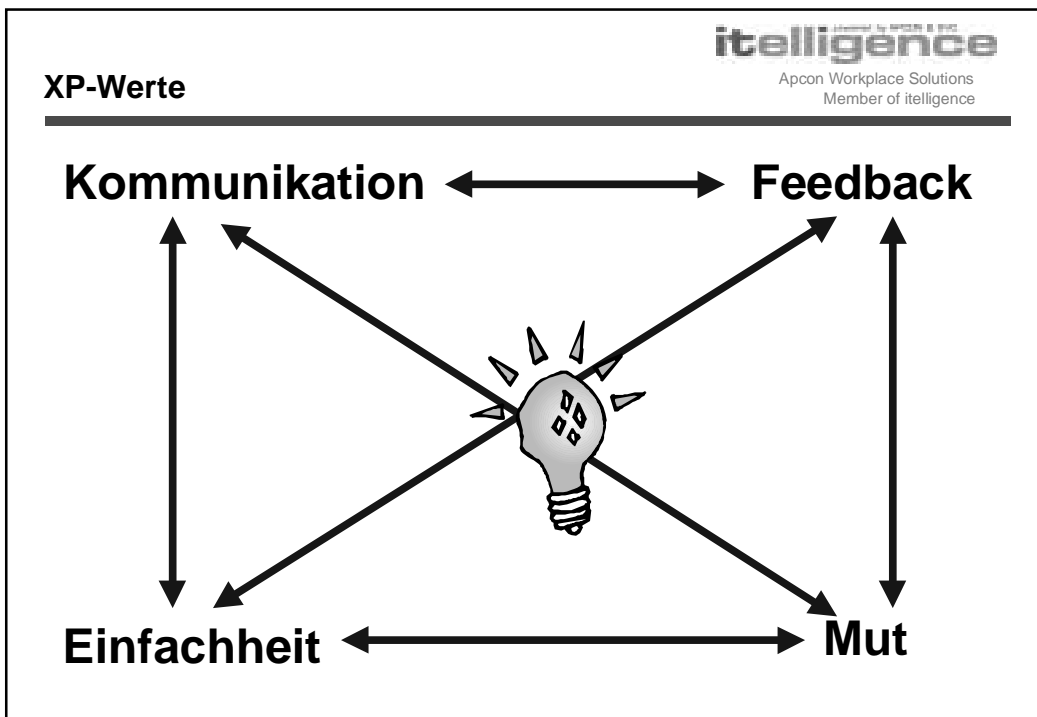
Hauptthese (2)

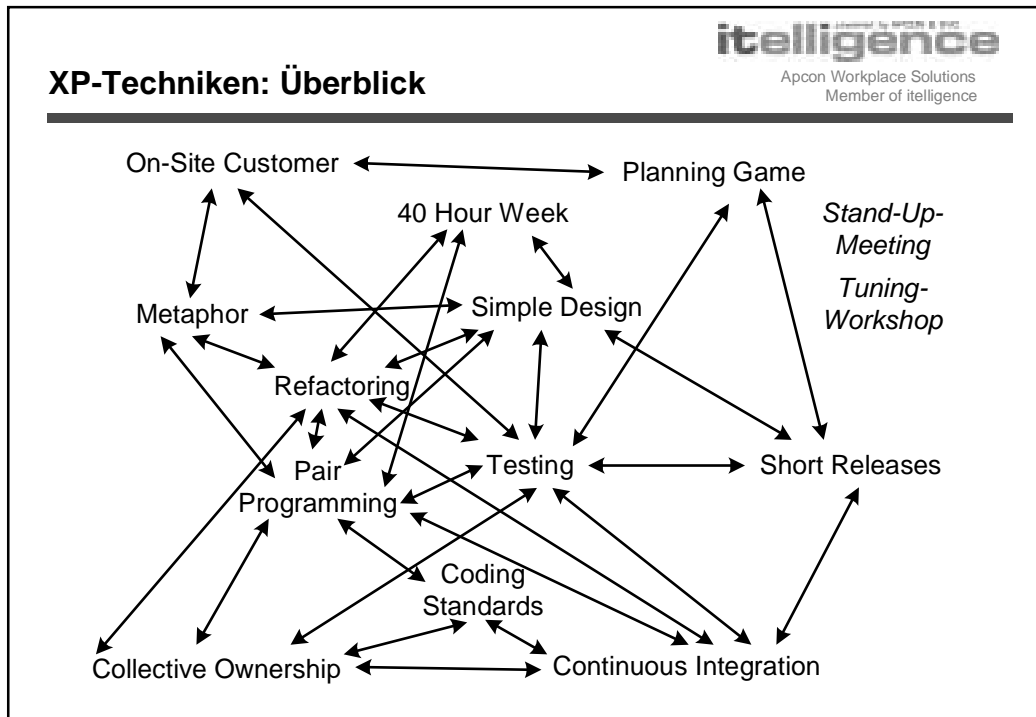
- XP *ist*
 - bewährt (keine Technik ist neu)
 - anwendungsorientiert (anforderungsgetrieben)
 - geplant und diszipliniert
 - kontrolliert und verlässlich
 - risiko-minimierend
 - „IT Kommunismus“
- XP hat zwei Seiten
 - für Entwickler: Freiheit, Flexibilität, Spaß
 - für Manager: Kontrollierbarkeit, Verlässlichkeit, Qualität

XP im Schnelldurchlauf

itelligence
Apcon Workplace Solutions
Member of itelligence

- Werte, Prinzipien und Techniken
- Zyklisch & Inkrementell
- Leichtgewichtig und Agil
- Personenorientiert
- Annahmen
 - *Eigenverantwortliche* Entwickler
 - *Kleines* Team
 - *Ein* Kunde

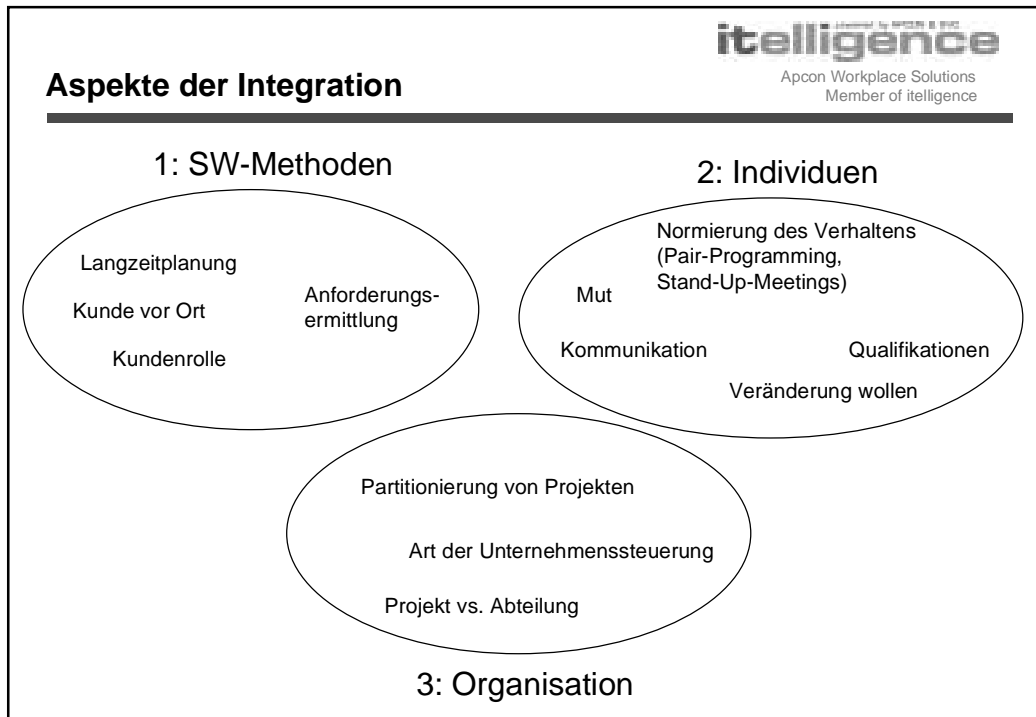





Fragen bei Einführung, Adaption und Integration

itelligence
Apcon Workplace Solutions
Member of itelligence

- Was sollte man aus der existierenden Landschaft übernehmen, was wird durch XP obsolet?
- Wo passen existierende Vorgehensweisen / Organisationsstrukturen nicht mit XP zusammen?
- Wo liegen die Grenzen von XP?
- Wie führt man die XP-Techniken ein (alles auf einmal, schrittweise, alphabetisch)?




-
- Projektmetaphern**
- 1. Ansatz: Vorgehensmodelle kompatibel, wenn Wertesysteme harmonieren.
 - Zu wenig greifbar
 - Wer würde die XP-Werte negieren?
 - Besser
 - geeignete Metapher(n) für Projektvorgehen finden
 - Vorgehensmodelle müssen mit den Metaphern harmonieren
- itelligence*
Apcon Workplace Solutions
Member of itelligence



Apcon Workplace Solutions
Member of itelligence

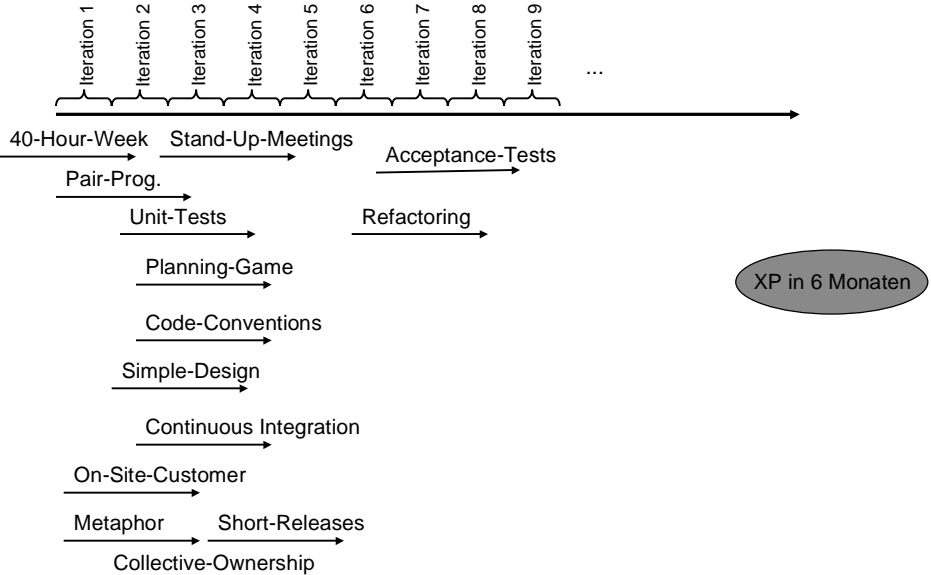
Einführung, Adaption und Integration

- XP muss **immer** angepasst werden.
- Immer ähnliche Fragestellungen führen zu immer ähnlichen Vorgehensweisen:
 - Metaphern finden
 - „Löse das größte Problem zuerst!“
 - Führe Tuning-Workshops durch!



Apcon Workplace Solutions
Member of itelligence


Beispiel: Einführung der Techniken



The diagram illustrates the introduction of various XP techniques over a series of iterations. A horizontal timeline at the top is marked with 'Iteration 1' through 'Iteration 9' and an ellipsis. Below this, arrows indicate when specific techniques are introduced:

- 40-Hour-Week:** Introduced at the start of Iteration 1.
- Pair-Prog.:** Introduced during Iteration 1.
- Unit-Tests:** Introduced during Iteration 2.
- Planning-Game:** Introduced during Iteration 3.
- Code-Conventions:** Introduced during Iteration 4.
- Simple-Design:** Introduced during Iteration 5.
- Continuous Integration:** Introduced during Iteration 6.
- On-Site-Customer:** Introduced during Iteration 7.
- Metaphor:** Introduced during Iteration 8.
- Short-Releases:** Introduced during Iteration 9.
- Collective-Ownership:** Introduced during Iteration 9.
- Stand-Up-Meetings:** Introduced between Iteration 2 and 3.
- Refactoring:** Introduced between Iteration 5 and 6.
- Acceptance-Tests:** Introduced between Iteration 7 and 8.


XP in 6 Monaten



Apcon Workplace Solutions
Member of itelligence

Anpassungen im Überblick (1)


- **Anforderungsfindung** kann aus anderen Methoden übernommen werden
 - z.B. Interviews, Szenarios und Glossar aus WAM
- **Multi-Channeling:** Jede Iteration hat einen Fokus. Anwender kann je Iteration neu festgelegt werden.
- **Kundenrolle**
 - Entscheidungsfindung explizit machen: Im Zweifel entscheidet der Geldgeber. Auftraggeber und Anwender unterscheiden.
 - Ggf. Produktmanager einführen.
 - *On-Site* Customer kann aufgeweicht werden.
- Projektetappen aus WAM, um lang laufende Projekte zu **planen**.



Apcon Workplace Solutions
Member of itelligence

Anpassungen im Überblick (2)


- Formale Reviews, um **sehr hohe Qualitätsstandards** zu halten.
- Dokumentation (Fachdokument) unter Bezugnahme auf Testfälle für **Wirtschaftsprüfer**.
- Diverse Techniken aus dem Bereich **Projektmanagement** (z.B. zum Risikomanagement).
- **Projektaufteilung** in Teams / Teilprojekte



Apcon Workplace Solutions
Member of itelligence

Probleme mit Story Cards


- Häufig schreiben Anwender die Stories vor dem Hintergrund existierender Systeme (z.B. maskenorientierte Hostsysteme)
 - ➔ Wiedergeburt der Host-Benutzungsmodells mit grafischer Oberfläche.
 - ➔ Inkonsistentes Benutzungsmodell.
 - ➔ Möglichkeiten aktueller Technologien nicht ausgenutzt.
- Anwender fokussieren auf ihren Arbeitskontext. Das führt bei Multi-Channeling-Anwendungen zu Problemen.
- Ansatz
 - Interviews und Szenarios (Business-Use-Cases)



Apcon Workplace Solutions
Member of itelligence

Management-Integration


- XP vermischt Auftraggeber und Anwender in der Kundenrolle.
- Das kann zu Problemen führen, weil Auftraggeber und Anwender unterschiedliche Perspektiven auf das Projekt haben:
 - Auftraggeber: Finanziert das Projekt und definiert geschäftspolitische Ziele.
 - Anwender: Nutzen das System (oder auch nicht) und kennen den Anwendungsbereich.
- Daher ist es mitunter sinnvoll, beide Rollen explizit zu trennen.



Apcon Workplace Solutions
Member of itelligence

Langfristige Planung

- XP konzentriert sich auf die aktuelle Iteration (max. 4 Wochen) und das aktuelle Release (max. 3 Monate).
- Auftraggeber interessieren sich für langfristige Planung, um die Machbarkeit des Gesamtprojektes einzuschätzen (z.B. Ablösung eines Hostsystems in einer Versicherung).
- Projektetappen geben eine Grobplanung für sehr lange Zeiträume.
- Projektetappen werden von Entwicklern in Zusammenarbeit mit dem Auftraggeber und ggf. Anwendern erstellt.




Apcon Workplace Solutions
Member of itelligence

Projektetappen

Teilziel	Wann
Tarifrechner für eine Sparte im Internet	31.3.00
Tarifrechner und Antragstellung für eine Sparte als Desktopsystem für Sachbearbeiter	16.5.00
Alle Sparten im Desktopsystem verfügbar.	30.8.00
...	...


- Anwendungsorientierte Ziele können von Außenstehenden verstanden und bewertet werden.
- Vision über das Vorgehen im Gesamtprojekt.
- Kann im Projektverlauf noch geändert werden.



Apcon Workplace Solutions
Member of itelligence

Integration auf individueller Ebene

- Neuer Prozess schwer plastisch darzustellen.
- Individuelle Neupositionierung - Komfortzone verlassen.
- Differierende individuelle Arbeitszeiten.
- Pair-Programming bei sehr ungleichen Qualifikationen wird zur "Lehrveranstaltung".
- Qualifikationsgrenzen.




Apcon Workplace Solutions
Member of itelligence

Projektaufteilung in Teams

- Teams mit mehr als 20 Entwicklern sinnlos.
- Horizontale und vertikale Aufteilung
- Gleichverteilung von Qualifikationen angestrebt

Einsatzkontexte	Berater	Schalter	Wertpapier-spezialist	Home-banking		
„Sparten“	<div style="display: flex; justify-content: space-around; align-items: center;"> <div style="border: 1px solid black; border-radius: 50%; padding: 5px;">Kredit</div> <div style="border: 1px solid black; border-radius: 50%; padding: 5px;">Wert-papier</div> <div style="border: 1px solid black; border-radius: 50%; padding: 5px;">...</div> </div>				}	Team je Sparte
Kernkonzepte	<div style="display: flex; justify-content: space-around; align-items: center;"> <div style="border: 1px solid black; padding: 5px;">Person Kunde</div> <div style="border: 1px solid black; padding: 5px;">Konto</div> <div style="border: 1px solid black; padding: 5px;">Produkt</div> <div style="border: 1px solid black; padding: 5px;">...</div> </div>				}	Architekturgruppe
Technologien						




Apcon Workplace Solutions
Member of itelligence


Fazit

- Werkzeugkisten-Ansatz: *Benutze, was nützt.*
- Es ist ungefährlicher, etwas zu XP hinzu zutun, als wegzunehmen.
- Integration mit anderen Methoden möglich, wenn sie mit den Metaphern harmonieren.
- XP kann als Konkretisierung von UP angesehen werden (wie auch anders herum).

- Termintreue!
- Reaktionsfähigkeit bei geänderten Kontextbedingungen.
- Gebrauchsqualität.
- Code-Qualität.




Apcon Workplace Solutions
Member of itelligence




Referenzen (1)

- Kent Beck: *Extreme Programming Explained - Embrace Change*. Addison-Wesley, 1999.
- Martin Fowler: *Refactoring : Improving the Design of Existing Code*. Addison-Wesley, 1999.
- **Lippert, Roock, Wolf: Software entwickeln mit eXtreme Programming - Erfahrungen aus der Praxis**. dpunkt-Verlag.
- Lippert, M., Roock, S., Wolf, H., Züllighoven, H.: *JWAM and XP - Using XP for framework development*, XP2000 conference. Cagliari, Sardinia, Italy (published in *Extreme Programming Examined* by Addison-Wesley, XP-Series, 2001).
- Lippert, M., Roock, S., Wolf, H., Züllighoven, H.: *XP in Complex Project Settings: Some Extensions*, XP2001 conference, Villasimius, Sardinia, Italy, 2001.


Apcon Workplace Solutions
Member of itelligence

Referenzen (2)

- Williams, Laurie, Kessler, Robert R.: *Experimenting with Industry's "Pair-Programming" Model in the Computer Science Classroom*, Journal on Software Engineering Education, December 2000.
- Williams, Laurie, Kessler, Robert R.: Cunningham, Ward, and Jeffries, Ron, *Strengthening the Case for Pair-Programming*, IEEE Software, July/Aug 2000.
- Williams, Laurie and Kessler, Robert R.: *All I Really Need to Know about Pair Programming I Learned In Kindergarten*, Communications of the ACM, May 2000.
- Williams, Laurie and Upchurch, Richard. In Support of Student Pair Programming, 2001 SIGCSE Conference on Computer Science Education, Charlotte, NC, February 2001.
- Cockburn, Alistair and Williams, Laurie: *The Costs and Benefits of Pair Programming*, eXtreme Programming and Flexible Processes in Software Engineering XP2000.
- Williams, Laurie and Kessler, Robert R.: *The Effects of "Pair-Pressure" and "Pair-Learning" on Software Engineering Education*. Conference of Software Engineering Education and Training 2000.


Apcon Workplace Solutions
Member of itelligence

XP-Ressourcen

- JUnit: <http://www.junit.org>
- WiKi-Web: <http://c2.com/cgi/wiki>.
- Laurie Williams:
<http://collaboration.csc.ncsu.edu/laurie/>
- <http://www.xprogramming.com>
- <http://www.ExtremeProgramming.org>
- <http://computer.org/seweb/dynabook/index.htm>
- <http://pairprogramming.com>

- JWAM framework: <http://www.jwam.org>