

## Fragebogen zu Extreme Administration (XA):

Name:	Bastian Ballmann
Beruf:	Systemadministrator Linux/Unix
Selbstbeschreibung: Interesse, Erfahrung, etc.	<ul style="list-style-type: none"><li>– Internet/Netzwerk-Sicherheit seit 1998</li><li>– OpenSource Entwickler Projekte Bluediving und P-A-T-H</li><li>– SysAdmin seit 2000</li></ul>

Alle agilen Methodiken der Softwareentwicklung weisen eine Gemeinsamkeit auf. Sie verfügen über Werte als Grundlage. Sie folgen Prinzipien und Grundsätzen und unterstützen diese mit praktischen Handlungen. Ausgehend von der Überlegung, dass die Systemadministration in vielen Punkten der Softwareentwicklung gleicht, hat der Autor eine Übertragung von agilen Methoden in die Welt der Systemadministration vorgenommen. Hieraus ist ein Gesamtkonzept für agile Systemadministration entstanden.

Dieser Fragebogen bittet Sie um Ihre Meinung zu der Anwendbarkeit dieser Ideen.

### **1. Themenbereich: Einstieg**

1. Frage: Haben Sie sich mit „Extreme Programming“ schon einmal beschäftigt?  
Ja Seit wann? 6 Monate
  
2. Frage: Mit einer anderen agilen Softwareentwicklungsmethode?  
Nein
  
3. Frage: Haben Sie sich mit Systemadministration schon einmal befasst?  
Ja

## 2. Themenbereich: XA Werte

- XP hat einen offenen und ehrlichen kommunikativen Ansatz.
- Die Werte von XP sind z. B. Kommunikation, Einfachheit, Feedback, Mut und Respekt
- In der Systemadministration gibt es kein direktes Äquivalent, aber es gibt den „Code of Ethics“ der SAGE ([www.sage.org/ethics/](http://www.sage.org/ethics/))
- Idee: Den „Code of Ethics“ übernehmen für XA
- Grundlage: Übersetzung des „Code of Ethics“ ins Deutsche
- Aufgabe: Bewerten Sie im folgenden die Relevanz der ethischen Grundsätzen mit [ - 1 2 3 4 5 6 + ]

### 1. Frage: Sind ethische Grundwerte für die Serveradministration sinnvoll?

Ja, weil Systemadministration die komplette Kontrolle über IT Infrastrukturen hat. Mit Macht sollte man gewissenhaft umgehen können.

#### **Code of Ethics:**

I.,,Professionalism: Keinen User durch persönliche Gefühle oder wegen seiner Ansichten unfair oder unprofessionell behandeln.“ (Hier finden sich die agilen Werte von Respekt und Ehrlichkeit wieder.)

*Wie wichtig ist Ihnen dieser Grundsatz [- 1 2 3 4 X 6 + ]*

II.,,Personal Integrity / Helfen: Ich werde ehrenwert Handeln und mit meiner Kompetenz entgegenkommen.“ (Hier finden die Werte „Ehrlichkeit“ und „Mut“ ihre Anwendung.)

*Wie wichtig ist Ihnen dieser Grundsatz [- 1 2 3 4 5 X + ]*

III.,,Privacy/Datenschutz: Ich werde nur auf private Informationen in Computern Zugriff nehmen, wenn es nötig ist und im Rahmen meiner technischen Pflichten. Ich werde die Vertraulichkeit aller Informationen, zu denen ich Zugriff habe, bewahren und beschützen, unabhängig davon, auf welchen Weg ich an die Information gekommen bin.“ (Hier finden die Werte „Integrität“ und „Fokus“, also sich nach SCRUM auf seine Aufgabe zu fokussieren ihre Anwendung.)

*Wie wichtig ist Ihnen dieser Grundsatz [- 1 2 3 4 5 X + ]*

IV.,,Laws and Policies / 3L: Ich möchte mich und andere über relevante Gesetze, Regeln und Prinzipien weiterbilden, die im Bezug zu meinen Pflichten stehen.“ (Hier findet sich „Engagement“ und „Kommunikation“ als Wert.)

*Wie wichtig ist Ihnen dieser Grundsatz [- 1 2 3 4 X 6 + ]*

V.,,Communication: Ich werde mit dem Management, den Usern und Kollegen über Computerthemen im beiderseitigem Interesse sprechen. Ich werde es anstreben zuzuhören und alle Ansichten versuchen zu verstehen.“ (Anwendung von Ehrlichkeit, Offenheit und Respekt vor anderen.)

*Wie wichtig ist Ihnen dieser Grundsatz [- 1 2 3 4 X 6 + ]*

## 2. Themenbereich: XA Werte

VI. „**System Integrity / Verfügbarkeit, Datensicherheit, Funktionsfähigkeit**“ : Ich werde versuchen, die notwendige Integrität, Ausfallsicherheit, und Verfügbarkeit der Systeme, für die ich verantwortlich bin, sicherzustellen.“ (Hier wird der „Fokus“ und das „Engagement“ aus den SCRUM und XP Werten beschrieben.)

*Wie wichtig ist Ihnen dieser Grundsatz [- 1 2 3 4 5 X + ]*

VII. „**Education / 3L**“: Ich werde immer weiter lernen und mein technisches Wissen und andere, mit meiner Arbeit im Zusammenhang stehenden, Fähigkeiten ausbauen. Ich werde mein Wissen und meine Erfahrung mit anderen teilen.“ (Hier findet sich ebenfalls „Engagement“ als Grundwert wieder. Darüber hinaus kommt durch die Weitergabe des Wissens, die „Kommunikation“ in den Fokus des Lernens.)

*Wie wichtig ist Ihnen dieser Grundsatz [- 1 2 3 4 5 X + ]*

VIII. „**Responsibility to Computing Community**“: Ich werde mit größeren Computer Communities zusammenarbeiten, um die Integrität des Netzwerks und anderen Computer Ressourcen instand halten zu können.“ (Hier findet dich „Kommunikation“ sowie „Feedback“ als Grundwert wieder.)

*Wie wichtig ist Ihnen dieser Grundsatz [- 1 2 3 4 X 6 + ]*

IX. „**Social Responsibility**“: Als ein informierter professioneller Systemadministrator, möchte ich zum schreiben und abstimmen von relevanten Regeln und Gesetzen in Übereinstimmung mit den ethischen Prinzipien auffordern.“ (Hier wird das „Feedback“ und die „gemeinsame“ Aufgabe explizit hervorgehoben.)

*Wie wichtig ist Ihnen dieser Grundsatz [- 1 2 3 X 5 6 + ]*

X. „**Ethical Responsibility**“: Ich werde versuchen einen sicheren, gesunden und produktiven Arbeitsplatz zu erstellen und zu pflegen. Ich werde mein Bestes geben, um Entscheidungen in Übereinstimmung mit der Sicherheit, der Privatsphäre (Datenschutz) und dem Wohlergehen meiner Gemeinschaft und der Öffentlichkeit zu treffen. Ich werde offene und ehrliche Kritik an meiner technischen Arbeit als angebracht annehmen und Anderen anbieten, sowie das Ansehen anderer sicher stellen.

Ich werde als „gutes Beispiel“ voran gehen und den hohen ethischen Standard pflegen, sowie den Grad meines Professionalismus in der Leistungsfähigkeit aller meiner Pflichten messen. Ich werde meine Kollegen und Mitarbeiter in der Befolgung des „Code of Ethics“ unterstützen.“ Im letzten Punkt findet sich eine Zusammenfassung aller Punkte wieder, die als eine Art Schwur angesehen werden kann.

*Wie wichtig ist Ihnen dieser Grundsatz [- 1 2 3 4 5 X + ]*

## 3. Themenbereich: Agile Prinzipien und Praktiken

## 1. Frage: Methoden-Abstufung

In der Methodikfamilie Crystal findet eine Abstufung der Methodik statt. Ausgehend von dem möglichen Schaden „Citality“ und der „Teamgröße“ wird die Methode angepasst. Ist eine solche Abstufung für Systemadministration (SA) auch sinnvoll?

Ja, weil für Systemadministratoren genauso Sicherheitsmaßnahmen gelten wie für Programmierer. IT Systeme, die ein Atomkraftwerk betreiben unterliegen viel strengeren Sicherheitsmaßnahmen als die Webseite von Tante Giselas Frittenbude.

Für wie gut halten Sie die Idee dieses Prinzip auf Systemadministration anzuwenden? [- 1 2 3 X 5 6 + ]

## 2. Frage: Test getriebene Systemadministration

Bei „Test getriebener Softwareentwicklung“ wird für eine neue Funktion zunächst ein Testfall (Unit-Test) geschrieben. Erst im Anschluss wird der funktionale Code implementiert. Dieser Test-First-Ansatz ist mit Monitoring-Lösungen, wie Nagios, ebenfalls möglich zu realisieren. Was für die Softwareentwicklung Klassen und Methoden sind, sind in der Welt der Systemadministration Server mit Ihren Diensten. Für einen Dienst werden positive, wie auch negative, Tests geschrieben. So wird die Funktionsfähigkeit überprüft und dauerhaft sichergestellt.

### Ist Testen / Monitoring in der Systemadministration sinnvoll?

Ja,

Für wie gut halten Sie die Idee dieses Prinzip auf Systemadministration anzuwenden? [- 1 2 3 4 5 X + ]

### Welche Vor- oder Nachteile hat es einen Testfall zu schreiben?

Vorteil : Der Sollzustand des Systems (sowohl negativ als auch positiv) ist bekannt und abweichungen werden automatisch und zeitnah erkannt.

Nachteil : Mehraufwand zum Schreiben von Testfällen, wobei dieser meist geringer ist als das Debugging im Nachhinein.

Für wie gut halten Sie die Idee dieses Prinzip auf Systemadministration anzuwenden? [- 1 2 3 4 5 X + ]

### Ist es sinnvoll den Testfall als erstes zu entwickeln (Test-First für Systemadministration)?

Ja,

Für wie gut halten Sie die Idee dieses Prinzip auf Systemadministration anzuwenden? [- 1 2 3 4 X 6 + ]

## 3. Frage: Kollektives Eigentum

Eine wichtiges Standbein von XP ist der Grundsatz des „Kollektiven Eigentums“. Dies beschreibt den erwünschten Zustand, dass jeder Softwareentwickler den Quellcode eines anderen Entwicklers ohne Rücksprache weiter entwickeln darf. Wenn der Entwickler, der eine Komponente geschrieben hat, gerade nicht erreichbar ist, kann viel Zeit verloren gehen. Die Abstimmungsarbeit nimmt eventuell mehrere Stunden oder Tage in Anspruch. Eventuell ist der Quellcode auf Grund von schlechter Dokumentation oder

### 3. Frage: Kollektives Eigentum

komplexen Programmierstil unverständlich oder gar nicht verwendbar.

Die Folgen aus „Kollektivem Eigentum“ ist der Wunsch nach einem verbindlicher Programming-Style-Guide, sowie der Einsatz von SCM-Lösungen (CVS, Subversion, Monotone, etc.).

#### **Findet sich kollektives Eigentum auch in der Systemadministration?**

Ja, wo? Configdateien, Filesystem Layouts

Nein, weil

Für wie gut halten Sie die Idee dieses Prinzip auf Systemadministration anzuwenden? [- 1 2 3 4 X 6 + ]

#### **Ist ein Style-Guide für Systemadministration sinnvoll?**

Ja, weil man so schneller relevante Einstellungen findet und Fehler vermeidet

Für wie gut halten Sie die Idee dieses Prinzip auf Systemadministration anzuwenden? [- 1 2 3 4 5 X + ]

#### **Ist der Einsatz von SCM-Lösungen für eine Systemkonfiguration denkbar und sinnvoll?**

Ja, so kann man immer nachvollziehen was nach Änderungen / Updates wirklich geändert wurde und was ein System ggf negativ beeinflusst hat

Nein,

Probleme?

Für wie gut halten Sie die Idee dieses Prinzip auf Systemadministration anzuwenden? [- 1 2 3 4 5 X + ]

### 4. Frage: Metapher

Der Aufbau einer bildlichen Vorstellung bietet bei der Softwareentwicklung entscheidende Vorteile. Planungsgespräche über Software-Design werden oft auf einer sehr abstrakten Ebene geführt. Bei diesen Gesprächen neigen Menschen dazu ungenau zu diskutieren, weil es keine „unfassbare“ Vorstellung von der Software gibt. Bei diesem Problem kann das XP Prinzip der Metapherbildung helfen.

Hier wird versucht abstrakte Sachverhalte durch bildhafte Vorstellungen zu unterstützen. Während des C3-Projekts wurde eine Lohnbuchhaltung für den Autokonzern Chrysler entwickelt. Kent Beck beschreibt eben diese Lohnbuchhaltung als „überdimensionale Tabellenkalkulation“. Dieses Gleichnis ermöglicht eine direkte Vorstellung der Aufgabe. Der Ausgangspunkt bei der Lohnbuchhaltung für einen gesamten Konzern von „einer flachen Tabelle“ auszugehen, ermöglicht eine modellhafte vorstellbare Sicht auf die Gesamtaufgabe.

#### **Ist solch eine Vereinfachung der Vorstellung auch für Systemadministration sinnvoll?**

Ja,

Anmerkungen?

Bildliche Vereinfachungen von komplexen, technischen Problemstellungen sind immer sinnvoll

#### 4. Frage: Metapher

Für wie gut halten Sie die Idee dieses Prinzip auf Systemadministration anzuwenden? [- 1 2 3 4 5 X + ]

#### 5. Frage: Pair Administration:

Beim Paar Programming von XP arbeiten zwei Entwickler an einem Arbeitsplatz. Sie teilen sich eine Tastatur und einen Bildschirm. Darüber hinaus findet eine Rollenverteilung statt. Derjenige der an der Tastatur, sitzt ist der „Driver“. Ihm obliegt die direkte Implementierung der Programmierung an der Tastatur. Er sorgt dafür, dass die Testfälle und der funktionale Quellcode richtig programmiert werden. Der „Partner“ oder „Co-Driver“ navigiert sich und den Fahrer sicher zum Ziel. Dem „Partner“ obliegt somit die Betrachtung des Ganzen. Er zeigt Lösungswege auf und sorgt dafür, dass bei der Arbeit ein kontinuierlicher Weg beibehalten wird. Nach einiger Zeit oder nach Aufgabengebiet wechseln die Rollenverteilungen. Der „Partner“ wird zum „Driver“ und umkehrt.

Die Vorteile bei Pair Programming sind: Gegenseitiges Schulen, Anerkennung, Fehlerkorrektur (4-Augen-Prinzip). Die Nachteile könnten sein: Höherer Zeitaufwand bei einfachen Tätigkeiten.

##### Ist Pair-Programming für Systemadministration auch sinnvoll?

Ja, aber nicht immer. Wenn komplexe Problemstellungen wie die Administration von VPN / Firewall Netzen gelöst werden soll, hilft ein Partner Fehler zu vermeiden und durch kollektives Eigentum wissen auch beide bescheid. Außerdem muss ein Administrator so viele technische Probleme lösen, dass er sich unmöglich in allen Bereichen optimal auskennen kann. Es sollte daher darauf geachtet werden, dass beide unterschiedliche Spezialgebiete, aber einen gemeinsamen Nenner haben.

Welche Vorteile sehen Sie?

Für wie gut halten Sie die Idee dieses Prinzip auf Systemadministration anzuwenden? [- 1 2 3 4 X 6 + ]

##### Können Sie sich Arbeitssituationen vorstellen, wo es sinnvoll ist zu zweit vor einem Terminal zu arbeiten?

Wenn ja, welche? S.o.

##### Würden Sie als Systemadministrator bevorzugt mit einem Partner zusammenarbeiten?

Ja,

#### 6. Frage: Administrationsstandard

Ein gemeinsamer Programmierstil, ist bei XP keine willkürliche Anforderung. Sie ist die Folge aus dem Grundsatz des „kollektiven Eigentums“, sowie des „Pair Programmings“. Jeder ist berechtigt den Quelltext

## 6. Frage: Administrationsstandard

des Anderen weiter zu entwickeln. Dies sollte in einer für alle verständlichen gleichen Form geschehen. Vor Beginn des Projekts einigt sich das Team bei XP auf einen solchen gemeinsamen Programmierstil. Diese Angleichung des Programmierstils wird durch „Pair Programming“ zusätzlich unterstützt.

Dieser Codierungsstandard lässt sich gut auf XA übertragen. Ein Administrator muss die Änderungen des anderen verstehen. Existiert keine klare Linie, wie Systeme administriert werden, wird eine Konfigurationsdatei schnell unübersichtlich.

Eine Art Style-Guide für die Administration, verbindlich und gemeinsam als Folge aus "Best Practise" des Administrationsteam wird erstellt.

**Was halten Sie von der Idee einen Codierungs Standard wie bei XP einzuführen, ist ein Administrationsstandard sinnvoll?**

Ja,

Comment: man findet so schneller relevante Einstellungen und vermeidet Fehler

Für wie gut halten Sie die Idee dieses Prinzip auf Systemadministration anzuwenden? [- 1 2 3 4 5 X + ]

## 7. Frage: Dokumentation

Software ist in den meisten Projekten einem fortlaufenden Verbesserungs- und Anpassungsprozess unterworfen. Ein Ziel der Softwareentwicklung ist diese Anpassungsfähigkeit. Eine Dokumentation muss sich ebenfalls den Veränderungen anpassen. Aus dem Bereich der Softwareentwicklung kommt der Ansatz, eine Dokumentation, aus dem Quelltext mit seinen Code-Kommentaren automatisiert, zu erstellen. Dieses Verfahren empfiehlt sich auch für die Dokumentationen von Servern.

**Ist die Idee Dokumentation aus den Konfiguration eines Servers zu generieren sinnvoll?**

Ja,

Comment: Dokumentation ist sehr wichtig, aber auch lästig. Alleine deshalb ist eine Automatisierung schon sinnvoll. Nebenbei unterliegen heutige IT Systeme viel zu häufigen Änderungen, als dass eine manuelle Doku sehr sinnvoll erscheint. Diese ist höchstens für Kernfunktionen der Infrastruktur tauglich, nicht aber für die spezielle Konfiguration einzelner oder mehrerer Systeme.

```
#!/**
```

```
##* @description IPSec Konfig für Kundennummer 0815
```

```
##* @author Marcel Wegermann
```

```
##* @todo Diese Konfig bitte am 2007-02-31 entfernen!
```

```
[..hier steht die Konfiguration..]
```

```
##*/ Ende der Konfiguration
```

## 7. Frage: Dokumentation

Für wie gut halten Sie die Idee dieses Prinzip auf Systemadministration anzuwenden? [- 1 2 3 4 X 6 + ]

**Wie dokumentieren Sie?**

**Wiki und inline in Configs**

## 8. Frage: Refactoring

Durch ständige Anpassung entwickelt sich die Software weiter. Die Software wird um eine Funktion erweitert, die zu Beginn gar nicht geplant war. Bei solchen Anpassungen muss sich das Software-Design mit anpassen. Wirken Weiterentwicklungen aufgesetzt, anstatt in die Architektur vollständig integriert, sollte ein „Refactoring“ durchgeführt werden. Beim Refactoring wird mehr als die neue Funktion implementiert. Alle Komponenten im Gesamtsystem, die in Beziehung stehen, kommen mit auf den Prüfstand. Durch diese Überarbeitung des Gesamtsystems wird sicher gestellt, dass auf Dauer eine wartbare und stabile Software geliefert werden kann.

Dieser Gedanke lässt sich auch auf Systemadministration übertragen. Wird auf die Wartung des gesamten Servers keine Rücksicht genommen, wird eine Wartung immer schwieriger. Das Refactoring stellt im einfachsten Fall eine Überarbeitung von Konfigurationsdateien dar oder die Verlagerung von Diensten auf andere Server (Lastverteilung/Wachstum).

**Ist die Idee Refactoring in die Systemadministration zu übernehmen sinnvoll?**

Ja,

Comment: Es ist nicht verkehrt seine Systeme von zeit zu zeit aufzuräumen, um die Übersichtlichkeit zu erhöhen, Platz zu schaffen, unnützen Code zu entfernen (weniger Sicherheitslücken)...

Für wie gut halten Sie die Idee dieses Prinzip auf Systemadministration anzuwenden? [- 1 2 3 4 5 X + ]

**Was sollte noch regelmäßig aufgeräumt/überarbeitet werden?**

**Refactoring könnte ebenso bedeuten neben Loadbalancing Dienste zu clustern oder heutzutage mehrere Systeme als virtuelle Hosts auf eine Maschine zu migrieren (e.g. XEN / Vmware)**

## 9. Frage: 40-Stunden-Woche?

Bei XP gilt der Grundsatz das Überstunden ein Tabu darstellen. In Ausnahmefällen sind Überstunden für einen Zeitraum von 1-2 Wochen im Rahmen. Sollten, über einen längeren Zeitraum, Überstunden notwendig sein, liegt ein Planungsfehler vor.

Sind Überstunden auf Dauer die Regel, brennen die Mitarbeiter aus und die Arbeitsleistung sinkt anhaltend. Diese Leistungsgrenze des Einzelnen kann bei 35 Stunden oder bei 45 Stunden liegen, wichtig ist, dass auf

## 9. Frage: 40-Stunden-Woche?

Dauer die Belastungsgrenze nicht überschritten wird.

Dieser Grundsatz ist nicht nur auf XA, sondern auf jeden Berufszweig zu übertragen.

**Ist es sinnvoll diesen Grundsatz in die Welt der Systemadministration zu übernehmen?**

Ja

Für wie gut halten Sie die Idee dieses Prinzip auf Systemadministration anzuwenden? [- 1 2 3 4 5 X + ]

## 10. Permanente Integration

Bei XP wird täglich der geänderten Programmcode in das Gesamtsystem integriert. Auf diesem Weg wird sicher gestellt, dass zu jedem Zeitpunkt ein funktionierendes Gesamtsystem existiert. Für diese Arbeit werden z.B. im Bereich der Java-Entwicklung Build-Tools, wie ANT oder Maven, eingesetzt.

Diese „Permanente Integration“ lässt sich mit dem „Reload eines Dienstes“ und damit mit der Aktivierung einer Konfigurationsänderung vergleichen.

**Ist es sinnvoll dieses Prinzip in die Welt der Systemadministration zu übernehmen?**

Nein, weil Änderungen an der Config sollten immer getestet werden bzw nach einem Update muss die Funktionsfähigkeit des Systems sichergestellt werden. Hierzu bedarf es IMHO keiner Automatisierung

Für wie gut halten Sie die Idee dieses Prinzip auf Systemadministration anzuwenden? [- X 2 3 4 5 6 + ]

## 11. Frage: Stand-Up-Meeting

Bei Scrum wird täglich zu Beginn der Arbeit ein Stand-Up Meeting abgehalten. Bei Scrum dient das Meeting zum schnellen Einholen von Feedback um den aktuellen Entwicklungsstand im Projekt abzufragen. Bei XA bietet sich ein Stand-Up Meeting ebenfalls an. Hier lässt sich innerhalb von Projekten der aktuelle Fortschritt abfragen. Im täglichen Umfeld lässt sich mit einem Stand-Up-Meeting eine Job-Übergabe realisieren.

**Ist es sinnvoll dieses Prinzip in die Welt der Systemadministration zu übernehmen?**

Ja, weil Probleme schnell ans Tageslicht kommen

Für wie gut halten Sie die Idee dieses Prinzip auf Systemadministration anzuwenden? [- 1 2 3 X 5 6 + ]

## 12. Frage: Customer-On-Side

Bei XP wird im engen Kontakt mit dem Kunden Software entwickelt. Der Kunde beteiligt sich durch seine Arbeit aktiv an der Entwicklung. Der Kunde hilft vor Ort bei den Entwicklern mit, die Anforderungen an die Software zu erschließen. Er arbeitet Beschreibungen der Anforderungen, so genannte „User-Stories“, aus.

Die Arbeit vor Ort mit dem Kunden ist zentraler Bestandteil der Systemadministration. Bei vielen Aufgaben geht die Administration nur bei und mit dem Kunden gemeinsam.

„Customer-On-Side“-Ansatz ist somit auf Systemadministration zu übertragen.

### **Ist es sinnvoll dieses Prinzip in die Welt der Systemadministration zu übernehmen?**

Nein, weil der Kunde meist kein Verständnis von der Technik hat

Für wie gut halten Sie die Idee dieses Prinzip auf Systemadministration anzuwenden? [- 1 X 3 4 5 6 + ]

### **In wie fern ist eine Beteiligung des Kunden im Projekt vorstellbar?**

## 13. Frage: Iteratives Vorgehen

Bei XP wird iterative vorgegangen. Kurzen Release-Zyklen werden angestrebt. Es ist wesentlich einfacher 1-2 Monate zu planen, als dies für eine Zeitspanne von 4-6 Monaten möglich ist. Jede Iteration sollte spezielle Geschäftsanforderungen erfüllen und als Ganzes sinnvoll sein. Das Ziel ist es dem Kunden möglichst schnell ein Eindruck der späteren Software zu verschaffen.

Das Grundprinzip von kurzen Release-Zyklen und einem Iterativen Vorgehen lässt sich auf die Welt der Systemadministration übertragen. Noch nicht komplett aufgebaute Dienste werden als Dummy-Dienste online gestellt. Erst nach und nach wird das gesamte Netzwerk-Konzept ausgebaut und umgesetzt.

### **Ist es sinnvoll dieses Prinzip in die Welt der Systemadministration zu übernehmen?**

Nein, weil bei der Administration entwickelt man meist nichts von grund auf neu, sondern konfiguriert bestehende Systeme und schaltet diese ggf zusammen.

Für wie gut halten Sie die Idee dieses Prinzip auf Systemadministration anzuwenden? [- X 2 3 4 5 6 + ]

## 14. Frage: User-Stories

Zur Anforderungsanalyse werden in XP Projekt User-Stories erstellt. Eine User-Story beschreibt mit einfachen Worten eine bestimmte Funktion der Software, aus Sicht des Benutzer.

Der Kunde wird in der Erstellung der User-Stories vom XP Team geschult und erhält Feedback zu seinen Anforderungen. Es dient darüber hinaus als Diskussionsgrundlage für Planungsgespräche. Oft werden diese User-Stories auf einzelnen Karteikarten verfasst.

## 14. Frage: User-Stories

Der Vorteil ist die Austauschbarkeit der einzelnen „Story-Cards“ untereinander. Auf diesem Wege lässt sich eine Iteration des Software-Release in der Planung gut zusammensetzen.

Dieses Mittel der Anforderungsanalyse ist ebenfalls auf die Systemadministration zu übertragen. Untereinander austauschbare Aufgaben lassen sich auf diesem Wege gut nach Prioritäten strukturieren.

**Ist es sinnvoll dieses Prinzip in die Welt der Systemadministration zu übernehmen?**

Ja, weil so der Kunde bildlich ohne technische Kenntnisse seine Anforderungen erläutern kann

Für wie gut halten Sie die Idee dieses Prinzip auf Systemadministration anzuwenden? [- 1 2 3 4 X 6 + ]