

1) Raymond favorisiert die Entwicklungsmethode des Basars, aber ist es auch nicht wichtig in bestimmten Bereichen trotzdem die Methode der Kathedrale einzusetzen?

2) Raymond sagt in einer seiner Rechtlinienpunkte, dass man früh und oft veröffentlichen soll, aber in welchem Entwicklungsstadium soll es passieren? Denn ich sehe das Problem, dass dann zu häufig Programme veröffentlicht werden und man davon überladen wird. 3) Ist es wirklich so gut für das Programm und für den Programmierer selbst?

1) Der von Raymond getroffene Unterschied zwischen einer Softwareentwicklung und Nutzung unter den Aspekten eines Basars und einer Kathedrale zeigt zwar die unterschiedlichen Konzepte auf, aber was passiert wenn der Basar größer wird und die Marktordnung nicht mehr auf einem „common sense“ basiert?(Vgl. S 3,31)

2) Eine von Raymond besteht in der Auffassung, dass viele Leute mehr Fehler sehen als eine begrenzte Anzahl und somit ein Projekt besser realisiert wird. (Vgl. S. 15) Es stellt sich die Frage, was passiert wenn Ideenklau und Abgrenzung von der Projekt Kommune entsteht?

3) **Das Modell von Raymond hat sich als Erfolg erwiesen und seine Grundidee der gemeinschaftlichen Arbeit so scheint es, bestätigt diese Idee. Es stellt sich die Frage, wie eine gemeinschaftliche Arbeit im Sinne des Basars ohne Hierarchie auskommt, sind alle Teilnehmer wirklich gleichberechtigt? (Vgl. S.39)**

4) Ausgehend von der Softwareentwicklung (Thema RWE im Seminar) unter dem Sicherheitsaspekt, was würde passieren wenn zum Beispiel Terroristen in einer Softwarekommune mitarbeiten und diese für ihre Zwecke missbrauchen würden, wo liegt die Kontrolle und Kontrollinstanz?

1. Sollten Quellcodes stets offen zugänglich sein, oder sollte es auch hier Firmengeheimnisse geben?

2. Welche Geschäftsmodelle bzw. Erlösquellen können bei der Bazaarmethode generiert werden?

3. Kann frei zugängliche und frei bearbeitende Software nicht auch große Sicherheitslücken und Gefahren verursachen?

1. Wendet wirklich nur Linux dieses Bazar Model an und dies macht sie so erfolgreich? Warum wenden dies nicht auch andere an um ihren Erfolg zu steigern, irgendwie ist mir dies unklar geblieben? Die Entwicklung des Internets und der freie Zugang würde dies doch ermöglichen?

2. Durch die Beteiligung der co-developers kann dies nicht auch dazu führen, dass diese das Programm missbrauchen anstatt es weiterzuentwickeln und kann die Zahl dieser User nicht ins unermessliche führen, sodass man keine Kontrolle mehr über diese hat? Oder ist es wirklich

vorteilhaft je mehr User desto besser? Wenn jeder seinen sag ich mal "Senf" dazu abgibt würde mich die Organisation interessieren?

3. Wie sieht es mit der Kostenverteilung aus? Auf Seite 16f. wird etwas von Kosten erwähnt, aber die genaue Aufteilung würde mich interessieren, wie setzen diese sich zusammen? Möchte nicht jeder was von dem "Kuchen" abhaben?

1. Wäre das Basar-Modell ein Beispiel für eine offene Gesellschaft, die Karl Popper anstrebte, oder auch Kant als Aufklärung verstand?

2. Kann ein Basar-Modell in der heutigen Gesellschaft erfolgreicher sein als das Kathedralen-Modell, obwohl dieses sich dauerhaft bewährt?

3. Wer kontrolliert/prüft, ob die Software am Ende auch für "Nicht-Informatiker" brauchbar ist bzw. entsteht laut Raymond überhaupt ein Endprodukt für Konsumenten?

1. Raymonds letztangeführte Regel (Nr.19) spricht von einem Development Coordinator, der ein (Kommunikations-)Medium so gut wie das Internet haben muss und die Führungsrolle ohne Zwang ausüben soll. Da er sozusagen eine untergeordnete Koordinations- oder Verwaltungsposition einnimmt, während die vielen, gleichberechtigten Mitarbeiter das Projekt ja durchaus selbstbestimmt vorantreiben - wäre der also gewissermaßen wie ein referentielles Subsystem nach Hejl zu verstehen?

2. Der Markt zeichnet sich ja auch durch den Wettbewerbsgedanken aus, bzw. nehme ich an, dass dieser dort u.a. als treibende Kraft gelten kann - nach dem Motto: Wer erweist sich als der bessere/beste Baumeister, Statiker, Maurer, Bildhauer, usw. beim Kathedralenbau? Was wäre die Entsprechung im Bazaar-Style? Eine gewisse Freude an der Kooperation? Dies würde sich mit verschiedenen neueren Entwicklungen verbinden lassen, etwa in der Anthropologie, wo der Mensch ja mittlerweile nun doch wieder zunehmend als grundlegend altruistisches Wesen begriffen wird, dessen Menschwerden viel mehr mit der Fähigkeit zur Kooperation verzahnt ist als bisher angenommen...

3. Im Bazaar-Style scheint ja der individuelle, eigenverantwortliche Macher in die Reihe gleichberechtigter Macher zurückzutreten, der Demiurg zersplittert. Was hat das für Folgen, etwa für den Begriff vom Individuum, vom Künstler, vom Urheber, aber auch von Expertise oder gar Bildung? Verändert sich hier nicht das Verhältnis von Spezialisierung und Interdisziplinarität? Böse Zungen könnten hier vielleicht von Schwarmintelligenz sprechen, die umfassende Bildung und Wissen im eigenen Kopf obsolet macht... So hässlich muss man es natürlich nicht formulieren - aber weil Projekte im Bazaar-Style nicht aus einzelner (!) Kraft durchgeführt werden müssen, sondern sich durch ein gewisses Mit- und Füreinander beinahe wie von selbst

organisieren, muss der jeweils einzelne hier auch nicht alles im Blick haben und auch nicht dazu fähig sein, das gesamte Projekt theoretisch alleine durchführen zu können...

Ein Anhänger der Genie-Ästhetik wäre bestürzt, aber vielleicht ist das Ganze auch eine gute Antwort auf die mittlerweile überbordende Anreicherung von potentiell Wissen... aber das sind alles ungeordnete Gedanken... die Sache ist ja viel zu komplex.

1.) "The Linux community seems to resemble a great babbling bazaar of differing agendas and approaches... out of which a coherent and stable system could seemingly emerge only by a succession of miracles." (S.3) Frage: Mystifiziert Raymond an dieser Stelle nicht ein wenig zu sehr? Schließlich gab es gute, nachvollziehbare Gründe, warum sich diese "community" entwickelt hat, wie sie sich entwickelt hat??

2.) "Every good work of software starts by scratching a developer's personal itch." (S.6) Frage: Schließt dieses Statement Softwareentwicklung gegen Bezahlung nicht nahezu vollständig aus?

3.) "If you have the right attitude, interesting problems will find you." (S.9) Frage: Was genau ist aber diese "right attitude"?

Könnte man sich nicht einen Software Markt vorstellen, bei dem jeder Entwickler anteilig bezahlt und gewürdigt wird?

Wie und warum hat sich der Software Markt letztendlich doch zu dem entwickelt, was er heute ist?

Haben etablierte Open Source Programme weniger Bugs als andere Programme?