

1 Θεμελιώδεις και μακροπρόθεσμες μετατροπές στο πρότυπο του ανοιχτού κώδικα

του [Tim O'Reilly](#)

Μάιος 2004

Το άρθρο αυτό βασίζεται σε μια ομιλία που έδωσα για πρώτη φορά στο ετήσιο συνέδριο τεχνολογίας του Warburg-Pincus τον Μάιο του 2003. Έκτοτε, έχω δημιουργήσει πάνω από είκοσι διαφορετικές εκδόσεις της ομιλίας αυτής, σε μέρη όπως το Συνέδριο περί Ανοιχτού Κώδικα του O'Reilly, ο Όμιλος Χρηστών Unix του Ηνωμένου Βασιλείου, το Τμήμα Ερευνών της Microsoft στο Ηνωμένο Βασίλειο, η IBM Hursley, η British Telecom, η εσωτερική συνάντηση «all-hands» της Red Hat, και το συνέδριο eWorld της BEA. Εν τέλει έδωσα στην εν λόγω ομιλία τη μορφή άρθρου για το βιβλίο που πρόκειται να εκδοθεί και έχει θέμα τον ανοιχτό κώδικα, «Perspectives on Free and Open Source Software,» (Προοπτικές για τα Λογισμικά Ελεύθερου και Ανοιχτού Κώδικα) που έχουν επιμεληθεί οι J. Feller, B. Fitzgerald, S. Hissam, και K. R. Lakhani και πρόκειται να εκδώσει το MIT Press το 2005.

Το 1962, ο Thomas Kuhn δημοσίευσε ένα καινοτομικό βιβλίο με τίτλο [The Structure of Scientific Revolutions](#) (Η Δομή των Επιστημονικών Επαναστάσεων). Στο βιβλίο αυτό, υποστηρίζει ότι η πρόοδος της επιστήμης δεν είναι σταδιακή αλλά (περίπου όπως αντιλαμβανόμαστε πλέον και τη βιολογική εξέλιξη), ένα είδος διακοπτόμενης ισορροπίας, με στιγμές μνημειωδών αλλαγών. Όταν ο Κοπέρνικος εξήγησε τις κινήσεις των πλανητών θέτοντας το αξίωμα ότι κινούνται γύρω από τον ήλιο και όχι γύρω από τη γη, ή όταν ο Δαρβίνος εισήγαγε τις ιδέες του περί της προέλευσης των ειδών, έκαναν κάτι περισσότερο από το να βασιστούν σε παλαιότερες ανακαλύψεις και να εξηγήσουν νέα πειραματικά στοιχεία. Όπως υποστηρίζει ο Kuhn, μια πραγματικά ριζοσπαστική επιστημονική καινοτομία, «σπάνια αποτελεί ή δεν αποτελεί ποτέ απλά μια προσθήκη στο ότι είναι ήδη γνωστό. Η αφομοίωσή της απαιτεί την ανασυγκρότηση των θεωριών του παρελθόντος και την εκ νέου αξιολόγηση προηγούμενων γεγονότων. Στην ουσία πρόκειται για μια εγγενή επαναστατική διαδικασία που σπάνια ολοκληρώνεται από έναν και μόνο άνθρωπο και ποτέ μέσα σε μια νύχτα.»^[1]

Ο Kuhn αναφέρθηκε σε αυτές τις επαναστατικές διαδικασίες στην επιστήμη ως «μετατροπές προτύπων», ένας όρος που πλέον έχει εισαχθεί στη γλώσσα για να περιγράψει κάθε ριζοσπαστική αλλαγή στα πλαίσια αναφοράς που έχουμε ορίσει.

Οι μετατροπές των προτύπων λαμβάνουν χώρα από καιρού εις καιρόν τόσο στον επιχειρηματικό τομέα όσο και στις επιστήμες. Και όπως ισχύει για τις επιστημονικές επαναστάσεις, δέχονται σκληρότατες επιθέσεις, ενώ οι βασικές ιδέες δεν γίνονται ευρέως αποδεκτές παρά μόνο πολύ καιρό μετά την πρώτη τους εμφάνιση. Επιπλέον, συχνά παρουσιάζουν επιπλοκές που ξεπερνούν κατά πολύ τη διαισθητικότητα των δημιουργών τους.

Μια τέτοια μετατροπή παραδείγματος έλαβε χώρα με την εισαγωγή της τυποποιημένης αρχιτεκτονικής του προσωπικού υπολογιστή της IBM το 1981. Βασίζομενη κατά πολύ σε προηγούμενες βιομηχανικές πρακτικές, η IBM επέλεξε να αναπτύξει τον υπολογιστή της από έτοιμα στοιχεία, και να δώσει ευκαιρίες αντιγραφής του σχεδιασμού της από άλλους κατασκευαστές. Ως αποτέλεσμα, η αρχιτεκτονική των προσωπικών υπολογιστών της IBM κατέστη το τυπικό πρότυπο, εκτοπίζοντας με την πάροδο του χρόνου όχι μόνο άλλα σχέδια προσωπικών υπολογιστών, αλλά μέσα στις δύο επόμενες δεκαετίες, και τα σχέδια μικρο-υπολογιστών και μεγάλων υπολογιστικών συστημάτων.

Ωστόσο, τα ανώτερα στελέχη της IBM δεν κατάφεραν να αντιληφθούν τις πλήρεις συνέπειες της απόφασής τους. Την περίοδο εκείνη, το μερίδιο αγοράς της IBM στους υπολογιστές υπερέβαινε κατά πολύ τη σημερινή κυριαρχία της Microsoft στην αγορά υπολογιστικών συστημάτων επιφάνειας εργασίας. Τα λογισμικά αποτελούσαν μόνο ένα μικρό τμήμα της βιομηχανίας των ηλεκτρονικών υπολογιστών, ένα απαραίτητο μέρος του ολοκληρωμένου υπολογιστή, και συνόδευαν το λειτουργικό σύστημα παρά πωλούνταν ξεχωριστά. Οι ανεξάρτητες εταιρείες ανάπτυξης λογισμικού υπήρχαν παίζοντας το ρόλο του δορυφόρου στην επιλεγμένη πλατφόρμα υλικού εξοπλισμού. Έτσι λοιπόν, όταν ήρθε η ώρα να παρέχει ένα λειτουργικό σύστημα για τα νέα μηχανήματα, η IBM αποφάσισε να το αποκτήσει μέσω μιας μικρής εταιρείας που λειτουργούσε υπό την επωνυμία Microsoft, δίδοντάς της το δικαίωμα να μεταπωλεί το λογισμικό στο μικρό εκείνο τμήμα της αγοράς το οποίο η IBM δεν είχε υπό τον έλεγχό της. Καθώς οι κλωνοποιημένοι προσωπικοί υπολογιστές αναπτύσσονταν από χιλιάδες μικρούς και μεγάλους κατασκευαστές, η IBM έχασε την κυριαρχία της στη νέα αγορά. Το λογισμικό έγινε ο νέος ήλιος γύρω από τον οποίο άρχισε να κινείται η βιομηχανία. Η Microsoft, και όχι η IBM, έγινε η πιο σημαντική εταιρεία στον κλάδο των ηλεκτρονικών υπολογιστών.

Αλλά αυτό δεν είναι το μόνο δίδαγμα της ιστορίας. Στον αρχικό ανταγωνισμό για την κυριαρχία στην αγορά των προσωπικών ηλεκτρονικών υπολογιστών, οι εταιρείες έδιναν αγώνα να «ενισχύσουν» το πρότυπο των προσωπικών υπολογιστών, προσθέτοντας υποστήριξη για νέες περιφερειακές μονάδες, ταχύτερους διαύλους, και άλλες τεχνικές καινοτομίες. Τα στελέχη τους, εκπαιδευμένα στην προηγούμενη βιομηχανία των υπολογιστών στην οποία την πρώτη θέση κατείχε ο υλικός εξοπλισμός, ενεργούσαν βασιζόμενα στις αρχές του παλαιού προτύπου.

Οι πιο αδιάλλακτοι εξ αυτών, όπως ο Ken Olson της Digital, απαξίωσαν τον Η/Υ χαρακτηρίζοντάς το παιχνιδάκι, και αρνήθηκαν να εισέλθουν στην αγορά μέχρι που πλέον ήταν πολύ αργά. Αλλά ακόμα και πρωτοπόροι όπως η Compaq, η οποία οφείλει την αρχική της επιτυχία στην εισαγωγή στην αγορά του υπολογιστή σε μορφή «βαλιτσάκι», τον πρόγονο των σύγχρονων φορητών υπολογιστών (laptop), παραπλανήθηκαν από παλαιότερες αρχές που πλέον δεν μπορούσαν να βρουν καμία εφαρμογή στο νέο πρότυπο. Χρειάστηκε να κάνει την εμφάνισή του ένα αουτσάιντερ, ο Michael Dell, ο οποίος ξεκίνησε την επιχείρησή του πωλώντας ηλεκτρονικούς υπολογιστές μέσω ταχυδρομικών παραγγελιών από το δωμάτιο του στη φοιτητική εστία του πανεπιστημίου του, για να γίνει αντιληπτό ότι οι τυποποιημένοι ηλεκτρονικοί υπολογιστές ήταν απλά εμπορικά προϊόντα, και ότι το πλεονέκτημα στην αγορά δεν βρισκόταν στην ανάπτυξη καλύτερων ηλεκτρονικών υπολογιστών, αλλά στην ανάπτυξη υπολογιστών που ήταν αρκετά καλοί, μειώνοντας το κόστος παραγωγής υιοθετώντας τα καθορισμένα πρότυπα, και αναζητώντας το πλεονέκτημα σε τομείς όπως το marketing, η διανομή, και τα logistics. Τελικά, ο Dell, και όχι η IBM ή η Compaq, ήταν αυτός που έγινε ο μεγαλύτερος πωλητής υλικού εξοπλισμού ηλεκτρονικών υπολογιστών.

Στο εντωμεταξύ, η Intel, μια άλλη εταιρεία που στοιχημάτισε στη νέα πλατφόρμα, εγκατέλειψε τη δραστηριοποίησή της στον κλάδο των τσιπ μνήμης θεωρώντας πως επρόκειτο για έναν ευπρόσβλητο κλάδο και δεσμεύθηκε να καταβάλλει κάθε δυνατή προσπάθεια για την αναβάθμιση του νέου σχεδιασμού. Το γεγονός ότι οι περισσότεροι από τους ηλεκτρονικούς υπολογιστές που κατασκευάζονται σήμερα φέρουν το λογότυπο «Intel Inside» μας υπενθυμίζει το γεγονός ότι ακόμα και μέσα σε μια αρχιτεκτονική εμπορικού προϊόντος, υπάρχουν ευκαιρίες για την επίτευξη πλεονεκτήματος.

Θα αναρωτηθείτε βέβαια τι σχέση έχουν όλα αυτά με το λογισμικό ανοιχτού κώδικα;

Η πρότασή μου είναι ότι οι προγραμματιστές ελεύθερου και ανοιχτού κώδικα βρίσκονται σήμερα κατά πολύ στην ίδια θέση που βρισκόταν η IBM το 1981 όταν άλλαξε τους κανόνες της βιομηχανίας των ηλεκτρονικών υπολογιστών, αλλά δεν κατάφερε να κατανοήσει τις συνέπειες της

αλλαγής, επιτρέποντας στους άλλους να καρπωθούν τα οφέλη. Οι περισσότεροι από τους υπάρχοντες πωλητές λογισμικού δεν βρίσκονται σε καλύτερη κατάσταση, καθώς παίζουν με τους παλαιούς κανόνες ενώ οι νέοι κανόνες αλλάζουν τη μορφή και το σχήμα της βιομηχανίας γύρω τους.

Στις ομιλίες μου κάνω ένα μικρό τεστ που μου επιτρέπει να βλέπω εάν το κοινό μου από επαγγελματίες στον κλάδο των ηλεκτρονικών υπολογιστών σκέφτεται σύμφωνα με το παλιό ή το νέο πρότυπο. «Πόσοι από εσάς χρησιμοποιείτε τα Linux;» Ρωτάω. Ανάλογα με τον τόπο συγκέντρωσης, το 20-80% του κοινού μπορεί να σηκώσει τα χέρια του. «Πόσοι από εσάς χρησιμοποιείτε το Google;» Όλα τα χέρια μέσα στην αίθουσα σηκώνονται. Και τότε όλα αρχίζουν να εμφανίζονται πιο ξεκάθαρα. Κάθε ένας από αυτούς χρησιμοποιεί το μαζικό σύμπλεγμα των 100.000 διακομιστών Linux του Google, αλλά δεν κατάφεραν να απαντήσουν σωστά στην πρώτη απάντηση καθώς στο μυαλό τους «το λογισμικό που χρησιμοποιείτε» ορίζεται ως το λογισμικό που εκτελείται στον υπολογιστή που έχετε μπροστά σας. Οι περισσότερες από τις ασυναγώνιστες εφαρμογές («killer apps») του Internet, εφαρμογές που χρησιμοποιούνται από εκατοντάδες εκατομμύρια ανθρώπων, λειτουργούν σε Linux ή FreeBSD. Αλλά το λειτουργικό σύστημα, όπως ορίστηκε πιο πάνω, είναι για τις εφαρμογές αυτές μόνο ένα συστατικό στοιχείου ενός ευρύτερου συστήματος. Η πραγματική τους πλατφόρμα είναι το Internet.

Μελετώντας ακριβώς αυτές τις εφαρμογές νέας γενιάς μπορούμε να αρχίσουμε να κατανοούμε την πραγματική μακροπρόθεσμη σημασία των μετατροπών του προτύπου ανοιχτού κώδικα.

Εάν οι πρωτοπόροι του ανοιχτού κώδικα θέλουν να επωφεληθούν από την επανάσταση που έχει σημειωθεί, πρέπει να δούν *πίσω από* τα επιφανειακά στοιχεία των κινήσεων του ελεύθερου και ανοιχτού κώδικα, και να κατανοήσουν εις βάθος τόσο τα αίτια όσο και τις συνέπειες της επανάστασης αυτής.

Ο πρωτοπόρος στον τομέα της Τεχνητής Νοημοσύνης Ray Kurzweil είπε κάποτε, «Είμαι εφευρέτης. Αρχισα να ενδιαφέρομαι για τις μακροπρόθεσμες τάσεις διότι μια εφεύρεση πρέπει να γίνεται κατανοητή από τον ευρύτερο κόσμο στον οποίο ολοκληρώνεται, και όχι τον κόσμο από τον οποίο ξεκίνησε.»^[2]

Μου είναι χρήσιμο να βλέπω τον ανοιχτό κώδικα ως μια έκφραση τριών σημαντικών, μακροπρόθεσμων τάσεων:

- [Της εμπορευματοποίησης του λογισμικού](#)
- [Της συνεργασίας μέσω δικτύου](#)
- [Της δυνατότητας προσαρμογής του λογισμικού \(λογισμικό ως υπηρεσία\)](#)

Οι μακροπρόθεσμες τάσεις σαν και αυτές, και όχι τόσο το [Μανιφέστο Ελεύθερου Λογισμικού](#) ή ο [Ορισμός του Ανοιχτού Κώδικα](#), θα πρέπει να αποτελούν τον φακό μέσα από τον οποίο μπορούμε να κατανοήσουμε τις αλλαγές που πραγματοποιούνται.

2 Το Λογισμικό ως Εμπορικό Προϊόν

Στη μελέτη του , [Some Implications of Software Commodification](#), (Ορισμένες συνέπειες της εμπορευματοποίησης του λογισμικού) ο Dave Stutz αναφέρει τα εξής:

Η λέξη εμπορικό προϊόν χρησιμοποιείται σήμερα για να δηλώσει την «τροφή» των βιομηχανικών διαδικασιών: πράγματα ή ουσίες που κρίνονται πολύτιμα ως βασικά δομικά στοιχεία για πολλούς διαφορετικούς σκοπούς. Λόγω της πολύ γενικής τους αξίας, χρησιμοποιούνται κατά κανόνα σε μεγάλες ποσότητες και με πολλούς διαφορετικούς τρόπους. Τα εμπορικά προϊόντα προέρχονται

πάντα από περισσότερους από έναν κατασκευαστές, και οι καταναλωτές μπορούν να υποκαταστήσουν το προϊόν ενός κατασκευαστή με το προϊόν κάποιου άλλου κατασκευαστή χωρίς να υποστούν καμία συνέπεια. Επειδή κατά τον τρόπο αυτό τα εμπορικά προϊόντα επιδέχονται υποκατάστασης, καθορίζονται από τα ενιαία πρότυπα ποιότητας με τα οποία πρέπει να συμμορφώνονται. Αυτά τα πρότυπα ποιότητας βοηθούν στην αποφυγή της νοθείας, ενώ παράλληλα συμβάλλουν στην ταχεία και εύκολη αξιολόγηση, η οποία με τη σειρά της αποφέρει κέρδη παραγωγικότητας.

Η εμπορευματοποίηση του λογισμικού καθοδηγείται από διάφορα πρότυπα, και ιδιαίτερα από την αύξηση των συστημάτων που προσανατολίζονται στις επικοινωνίες, όπως είναι το Internet, τα οποία εξαρτώνται από κοινά πρωτόκολλα, και καθορίζουν τις διεπαφές και τους τύπους δεδομένων που μοιράζονται μεταξύ των συνεργαζόμενων στοιχείων παρά μεταξύ των εσωτερικών των εν λόγω στοιχείων. Τα εν λόγω συστήματα συνίστανται υποχρεωτικά από μέρη που επιδέχονται αντικατάστασης. Ένας διακομιστής web όπως ο Apache ή ο IIS της Microsoft, ή οι διακομιστές όπως ο Internet Explorer, Netscape Navigator, ή Mozilla, ανταλλάσσονται με ευκολία, διότι προκειμένου να λειτουργήσουν, πρέπει να υλοποιούν το πρωτόκολλο HTTP και τη μορφή δεδομένων HTML. Το Sendmail μπορεί να αντικατασταθεί από το Exim ή Postfix ή το Microsoft Exchange καθώς όλα πρέπει να υποστηρίζουν πρωτόκολλα ανταλλαγής μηνυμάτων ηλεκτρονικού ταχυδρομείου όπως είναι το SMTP, POP και IMAP. Το Microsoft Outlook μπορεί εύκολα να αντικατασταθεί από το Eudora, ή το Pine, ή το Mozilla mail, ή έναν πελάτη web mail όπως είναι το Yahoo! Mail για τον ίδιο λόγο.

(Ως προς αυτό, αξίζει να σημειωθεί ότι το Unix, το σύστημα πάνω στο οποίο βασίζεται το Linux, διαθέτει επίσης αρχιτεκτονική που εστιάζεται στις επικοινωνίες. Στο [Περιβάλλον Προγραμματισμού Unix](#), οι Kernighan και Pike περιγράφουν εύγλωττα τον τρόπο με τον οποίο τα προγράμματα Unix θα πρέπει να γράφονται ως μικρά κομμάτια σχεδιασμένα να συνεργάζονται σε «αγωγούς», που διαβάζουν και γράφουν αρχεία ASCII παρά αποκλειστικές μορφές δεδομένων. Ο Eric Raymond δίνει μια σύγχρονη έκφραση του θέματος αυτού στο βιβλίο του, [The Art of Unix Programming](#).)

Θα πρέπει να σημειωθεί ότι σε ένα περιβάλλον με τυπικά πρωτόκολλα που εστιάζεται στις επικοινωνίες, τα λογισμικά αποκλειστικού όσο και ανοιχτού κώδικα έγιναν εμπορικά προϊόντα. Ο web browser Internet Explorer της Microsoft αποτελεί εμπορικό προϊόν όσο και ο web server ανοιχτού κώδικα Apache, καθότι και τα δύο περιορίζονται από τα ανοιχτά πρότυπα του web. (Εάν η Microsoft είχε καταφέρει να κερδίσει το κύριο μερίδιο της αγοράς και στα δύο άκρα του αγωγού πρωτοκόλλου μεταξύ του web browser και του server, τότε τα πράγματα θα είχαν αλλιώς! Ανατρέξτε στο άρθρο [How the Web was almost won](#) (Πώς το Web κατάφερε σχεδόν να κερδίσει) για να δείτε τις απόψεις μου σχετικά με το ζήτημα. Το παράδειγμα αυτό καθιστά σαφή έναν από τους σημαντικούς ρόλους που παίζει ο ανοιχτός κώδικας στη «διατήρηση της ακεραιότητας των προτύπων». Ο ρόλος αυτός αναγνωρίζεται από τους οργανισμούς όπως η W3C, που εμφανίζονται όλο και πιο διστακτικοί να υιοθετήσουν πρότυπα που έχουν μόνο αποκλειστικές υλοποιήσεις ή υλοποιήσεις που προστατεύονται από διπλώματα ευρεσιτεχνίας.)

Επιπλέον, ακόμα και λογισμικά που εμφανίζονται αρχικά ως αποκλειστικά καταλήγουν να τυποποιούνται και εν τέλει εμποριοποιούνται. Ο Dave Stutz περιγράφει με πολύ παραστατικό τρόπο τη διαδικασία αυτή στη μελέτη του που φέρει τον τίτλο [The Natural History of Software Platforms](#) (Οι Πλατφόρμες Λογισμικού και η Φυσική τους Ιστορία):

Λαμβάνει χώρα μέσα από τη σκλήρυνση του εξωτερικού κελύφους που παρουσιάζουν οι πλατφόρμες με την πάροδο του χρόνου. Καθώς μια πλατφόρμα επιτυγχάνει στην αγορά, οι διασυνδέσεις προγραμματισμού εφαρμογών (API), οι διασυνδέσεις χρήστη (UI), το σύνολο των χαρακτηριστικών, οι μορφές των αρχείων και οι διεπαφές προσαρμογής παγιώνονται και η αλλαγή

τους καθίσταται όλο και πιο δύσκολη. (Στην πραγματικότητα, παγιώνονται σε τέτοιο βαθμό που πραγματικά μετατρέπονται σε συσκευές υλικού εξοπλισμού!) Η διαδικασία της παγίωσης καθιστά τις επιτυχείς πλατφόρμες εύκολους στόχους για όσους επιχειρούν να τις αντιγράψουν, και η αντιγραφή σηματοδοτεί την αρχή του τέλους για τα περιθώρια κέρδους από τις πλατφόρμες.

Συνεπώς με την άποψη αυτή, η αντιγραφή των Windows και του Office της Microsoft έχει αποτελέσει κύριο στόχο στον κλάδο του Ελεύθερου και Ανοιχτού Κώδικα. Κατά το παρελθόν, η Microsoft κατάφερε να αντιμετωπίζει επιτυχώς τις προσπάθειες αντιγραφής, αναθεωρώντας συνεχώς τις μορφές των διασυνδέσεων προγραμματισμού εφαρμογών και των μορφών των αρχείων, αλλά πάντα υπήρχαν ενδείξεις ότι δεν θα μπορούσε να συνεχίσει έτσι για πάντα. Η απανταχού παρουσία οδηγεί στην τυποποίηση, και η μη απαιτούμενη καινοτομία προς υπεράσπιση του μονοπωλίου απορρίπτεται από τους χρήστες.

Ποιες είναι μερικές από τις συνέπειες της εμπορευματοποίησης του λογισμικού; Κανείς μπορεί να παρασυρθεί και να δει μόνο την υποτίμηση ενός πράγματος που άλλοτε αποτελούσε αντικείμενο τεράστιας αξίας. Για το λόγο αυτό, ο ιδρυτής της Red Hat, Bob Young, παρατήρησε κάποτε το εξής: «Στόχος μου είναι να συρρικνώσω το μέγεθος της αγοράς λειτουργικών συστημάτων». (Η Red Hat ωστόσο στόχευε στην κατάκτηση ενός μεγάλου μέρους αυτής της μικρότερης αγοράς!) Οι υπερασπιστές του καθεστώτος αυτού, όπως ο αντιπρόεδρος της Microsoft, Jim Allchin, έκαναν δηλώσεις του τύπου «ο ανοιχτός κώδικας είναι η καταστροφή της πνευματικής ιδιοκτησίας», και δημιούργησαν μια ψυχρή εικόνα στην οποία παρουσιάζεται η καταστροφή μιας μεγάλης βιομηχανίας, με τίποτα να μην παίρνει τη θέση της.

Εκ πρώτης όψεως, ο Allchin φαίνεται να έχει δίκιο. Το Linux έχει τώρα κέρδος δεκάδων δισεκατομμυρίων δολαρίων από έσοδα που σχετίζονται με τον υλικό εξοπλισμό διακομιστών, με τα έσοδα από τα λογισμικά να αποτελούν απλά ένα σφάλμα στρογγυλοποίησης. Παρά την ανερχόμενη κυριαρχία του Linux στην αγορά των διακομιστών, η Red Hat, η μεγαλύτερη εταιρεία διανομής του Linux, έχει ετήσια έσοδα μόνο 126 εκατομμυρίων δολαρίων, έναντι της Microsoft που πετυχαίνει έσοδα 32 δισεκατομμυρίων δολαρίων. Ένα μεγάλο μέρος της αξίας των λογισμικών φαίνεται πώς έχει εξατμιστεί.

Αλλά πρόκειται για αξία ή για γενικά έξοδα; Οι υπέρμαχοι του ανοιχτού κώδικα αρέσκονται να λένε ότι δεν καταστρέφουν την πραγματική αξία, αλλά μάλλον ότι αφαιρούν από το σύστημα τα μη αποτελεσματικά στοιχεία. Όταν ο ανταγωνισμός ρίχνει τις τιμές, τα επίπεδα αποτελεσματικότητας και μέσης κερδοφορίας αυξάνονται. Οι εταιρείες που δεν είναι σε θέση να προσαρμοστούν στα νέα επίπεδα τιμών υφίστανται αυτό που ο οικονομολόγος E.F. Schumpeter ονομάζει «δημιουργική καταστροφή», αλλά αυτό που «χάνεται» επιστρέφει πολλαπλάσιο υπό τη μορφή υψηλότερης παραγωγικότητας και νέων ευκαιριών.

Η Microsoft επωφεληθήκε, από κοινού με τους πελάτες, από τον τελευταίο γύρο «δημιουργικής καταστροφής» καθώς συντελέσθηκε η εμπορευματοποίηση του υλικού εξοπλισμού των ηλεκτρονικών υπολογιστών. Σε αυτή τη χρονική στιγμή, η Microsoft βλέπει την εμπορευματοποίηση των λειτουργικών συστημάτων, των βάσεων δεδομένων, των web servers και browsers, καθώς και του σχετικού λογισμικού ως καταστροφικά για τη βασική επιχειρηματική της δραστηριότητα. Αλλά η καταστροφή αυτή δημιούργησε την ευκαιρία για τις ασυναγώνιστες εφαρμογές του Internet όπως Yahoo!, Google, Amazon, eBay – για να αναφέρουμε μόνο ορισμένες – να επωφεληθούν.

Έτσι λοιπόν προτιμώ να κρατάω την άποψη του Clayton Christensen, του συγγραφέα του [The Innovator's Dilemma](#) (Το δίλημμα του καινοτόμου) και του [The Innovator's Solution](#) (Η λύση του καινοτόμου). Σε ένα πρόσφατο άρθρο στη *Harvard Business Review*, αναλύει τον «νόμο της διατήρησης των ελκυστικών κερδών» ως εξής:

Όταν τα ελκυστικά κέρδη παύουν να υπάρχουν σε ένα στάδιο στην αλυσίδα αξίας διότι ένα προϊόν καθίσταται αρθρωτό και εμπορευματοποιείται, η ευκαιρία επίτευξης ελκυστικών κερδών με αποκλειστικά προϊόντα κάνει κατά κανόνα την εμφάνισή της σε αμέσως μετέπειτα στάδιο.^[3]

Στις μεταβολές προτύπων για τις οποίες κάνουμε λόγο εδώ βλέπουμε καθαρά την πρόταση του Christensen.^[4] Ακριβώς όπως η εμπορευματοποίηση του βασικού σχεδιασμού των προσωπικών υπολογιστών της IBM οδήγησε σε ευκαιρίες για ελκυστικά κέρδη από τα λογισμικά, έτσι και η εμπορευματοποίηση των λογισμικών ανοιχτού κώδικα πάνω στα οποία βασίζεται το Internet, αποφέρει σημαντικά κέρδη σε μια νέα κατηγορία αποκλειστικών εφαρμογών στις οποίες σε άλλο σημείο αναφέρομαι ως [«infoware»](#). Δικτυακοί τόποι όπως το Google, Amazon, και το salesforce.com παρέχουν την πιο σοβαρή πρόκληση στην παραδοσιακή κατανόηση του λογισμικού ελεύθερου και ανοιχτού κώδικα. Εδώ υπάρχουν εφαρμογές οι οποίες έχουν αναπτυχθεί πάνω σε Linux, αλλά είναι έντονα αποκλειστικές. Επιπλέον, ακόμα και όταν χρησιμοποιείτε και τροποποιείτε λογισμικό που έχει διανεμηθεί κάτω από τις πιο περιοριστικές άδειες ελεύθερου λογισμικού, τις γενικές άδειες χρήσης (GPL), οι δικτυακοί αυτοί τόποι περιορίζονται από οποιαδήποτε από τις διατάξεις, το σύνολο των οποίων βασίζεται στο παλαιό πρότυπο. Οι προσαυτίες των GPL ενεργοποιούνται από τη διανομή του λογισμικού, και όμως οι πωλητές εφαρμογών που βασίζονται στο web δεν διανέμουν ποτέ οποιοδήποτε λογισμικό: απλά εκτελείται στη γενική κατάσταση του Internet, και παρέχεται ως υπηρεσία παρά ως πακεταρισμένη εφαρμογή λογισμικού.

Ακόμα πιο σημαντικό είναι ότι, ακόμα και αν αυτοί οι δικτυακοί τόποι έδιναν τον ανοιχτό τους κώδικα, οι χρήστες δεν θα ήταν εύκολα σε θέση να δημιουργήσουν ένα πλήρες αντίγραφο της υπό εκτέλεσης εφαρμογής! Η εφαρμογή είναι μια βάση δεδομένων που ενημερώνεται δυναμικά η πρακτικότητα της οποίας προέρχεται από την αρτιότητά της και τη σύγκλισή της, και σε πολλές περιπτώσεις, από τη δικτυακή δράση των χρηστών που συμμετέχουν.

(Για να διασφαλιστεί ότι οι χρήστες θα είχαν πολλά οφέλη, ορισμένοι από τους αλγόριθμους του Google ή του Amazon's One-Click ήταν διαθέσιμοι σε όλους, αλλά το σημείο παραμένει: ένα δείγμα του συνόλου του ανοιχτού κώδικα του Google δεν μπορεί να σας δώσει το Google, εκτός και αν είστε σε θέση να αναπτύξετε την ικανότητα να κινήστε μέσα στο web και να δημιουργείτε κατοπτρικές εικόνες, κατά τον ίδιο τρόπο που το κάνει το Google.)

Και οι ευκαιρίες δεν περιορίζονται εδώ. Υπάρχουν τεράστιες αποκλειστικές ευκαιρίες οι οποίες κρύβονται εντός του συστήματος. Ο Christensen λέει τα εξής:

Ελκυστικά κέρδη . . . μετακινηθείτε σε άλλο σημείο της αλυσίδας αξιών, συνηθέστερα σε υποσυστήματα από τα οποία συναρμολογείται το αρθρωτό προϊόν. Αυτό συμβαίνει διότι οι βελτιώσεις στα υποσυστήματα, παρά η αρθρωτή αρχιτεκτονική του προϊόντος, είναι εκείνες που καθοδηγούν την ικανότητα του συμβολομεταφραστή να μετακινείται προς τα πάνω, προς πιο ελκυστικά περιθώρια κέρδους. Συνεπώς, τα υποσυστήματα αποεμπορευματοποιούνται και παρουσιάζουν ελκυστικό περιθώριο κέρδους.

Το μοτίβο αυτό το είδαμε στην αγορά των ηλεκτρονικών υπολογιστών με τους περισσότερους ηλεκτρονικούς υπολογιστές να φέρουν την επωνυμία «Intel Inside». Το Internet θα μπορούσε άνετα κατά την ίδια λογική να φέρει την επωνυμία «Cisco Inside».

Αλλά αυτές οι επιχειρηματικές ευκαιρίες «Intel Inside» δεν είναι πάντα εμφανείς, ούτε ανήκουν απαραίτητα σε αποκλειστικό υλικό εξοπλισμό ή λογισμικό. Το πακέτο BIND (Berkeley Internet Name Daemon) ανοιχτού κώδικα που χρησιμοποιείται για την εκτέλεση του Συστήματος Ονομάτων Περιοχών (DNS) παρέχει μια σημαντική απόδειξη.

Το επιχειρηματικό μοντέλο για το μεγαλύτερο μέρος λογισμικών του Internet αποδείχθηκε ότι δεν πωλούσε το λογισμικό αυτό (παρά τις προσφορές από πωλητές όπως οι NetManage και Spry, που τώρα πια έχουν παύσει από καιρό), αλλά υπηρεσίες που βασίζονταν στο εν λόγω λογισμικό. Το μεγαλύτερο μέρος των επιχειρήσεων αυτών – οι Παροχείς Υπηρεσιών Internet (ISP), οι οποίοι κατ' ουσία μεταπωλούν την πρόσβαση στη στοίβα πρωτοκόλλων TCP/IP και στους διακομιστές email και web – αποδείχθηκαν πως ήταν επιχειρήσεις χαμηλού περιθωρίου. Υπήρχε μία σημαντική εξαίρεση.

Το BIND είναι πιθανώς το μοναδικό πρόγραμμα που είναι περισσότερο σημαντικό για το Internet, και όμως ο συντηρητής του τα βγάζει πέρα με δυσκολία τις δύο τελευταίες δεκαετίες από δωρεές και αμοιβές από παροχή συμβουλευτικών υπηρεσιών. Στο εντωμεταξύ, η εγγραφή του ονόματος περιοχών – μια υπηρεσία πληροφοριών που βασίζεται στο λογισμικό – έγινε μια επιχείρηση που αποφέρει χιλιάδες εκατομμύρια δολάρια το χρόνο, ένα πραγματικό μονοπώλιο για Λύσεις Δικτύου, και η οποία ανέλαβε δράση κατόπιν σύναψης σύμβασης με την κυβέρνηση προτού προλάβει κανείς να συνειδητοποιήσει πόσο πολύτιμη μπορούσε να είναι. Η ευκαιρία της Intel Inside για το DNS δεν αποτέλεσε καθόλου μια ευκαιρία ανάπτυξης λογισμικού, αλλά μια ευκαιρία ανάπτυξης της υπηρεσίας διαχείρισης χώρου ονομάτων που χρησιμοποιείται από το λογισμικό. Σε μια ιστορική συγκυρία, το επιχειρηματικό μοντέλο διαχωρίστηκε από το λογισμικό.

Το ότι οι υπηρεσίες που βασίζονταν σε λογισμικό θα αποτελούσαν ένα κυρίαρχο επιχειρηματικό μοντέλο για λογισμικό ανοιχτού κώδικα αναγνωρίστηκε στο [The Cathedral & the Bazaar](#), στο δημιουργικό έργο του Eric Raymond πάνω στη μετακίνηση. Ωστόσο στην πρακτική, οι πρώτοι επιχειρηματίες που ασχολήθηκαν με τον ανοιχτό κώδικα εστίασαν τις προσπάθειες και την προσοχή τους στις υπηρεσίες που συνδέονταν με τη συντήρηση και την υποστήριξη του λογισμικού, παρά στο πραγματικό λογισμικό ως υπηρεσία. (Αυτό σημαίνει, ότι το λογισμικό ως υπηρεσία δεν αποτελεί υπηρεσία για την υποστήριξη του λογισμικού, αλλά λογισμικό για την υποστήριξη υπηρεσιών που αφορούν τους χρήστες!)

Ο Dell μας δίνει ένα τελευταίο μάθημα για τη σημερινή βιομηχανία ανάπτυξης λογισμικών. Όπως η εμπορευματοποίηση του υλικού εξοπλισμού ηλεκτρονικών υπολογιστών μείωσε τα ασυνήθιστα μεγάλα περιθώρια της IBM αλλά αύξησε το μέγεθος της αγοράς, δημιουργώντας μια τεράστια αξία για τους χρήστες, και πολυάριθμες ευκαιρίες για ένα νέο οικοσύστημα κατασκευαστών ηλεκτρονικών υπολογιστών για τους οποίους τα χαμηλότερα περιθώρια των ηλεκτρονικών υπολογιστών είχαν ακόμα κάποιο επιχειρηματικό νόημα, η εμπορευματοποίηση του λογισμικού στην πραγματικότητα θα διευρύνει την αγορά λογισμικού. Όπως αναφέρει και ο Christensen, σε αυτόν τον τύπο αγοράς, οι οδηγοί της επιτυχίας «ήταν η ταχύτητα της αγοράς και η ικανότητα να δίδεται στους πελάτες ακριβώς αυτό που χρειάζονται, όταν το χρειάζονται.»^[5]

Ακολουθώντας τη λογική αυτή, πιστεύω ότι η διαδικασία της ανάπτυξης προσαρμοσμένων διανομών θα εμφανιστεί ως μία από τις κυριότερες ανταγωνιστικές διαφορές μεταξύ των πωλητών Linux. Όπως ο Dell πρέπει να έχει το ρόλο διαμεσολαβητή των διάφορων συμβατικών κατασκευαστών που συναγωνίζονται να παράγουν στοιχεία που επιδέχονται υποκατάσταση στη χαμηλότερη δυνατή τιμή, ένας πωλητής Linux θα χρειαστεί να διαχειριστεί τη συνεχώς μεταβαλλόμενη ομάδα παροχών λογισμικού των οποίων οι ασύγχρονες εκδόσεις προϊόντων παρέχουν τις πρώτες ύλες για τις διανομές Linux. Εταιρείες όπως η [Progeny Systems](#) του Ian Murdock ιδρυτή του Debian το συναντούν αυτό στην καρδιά των επιχειρήσεών τους, αλλά ακόμα και οι πωλητές παλαιάς γραμμής Linux όπως η SuSe καθώς και νέες επιχειρήσεις όπως η Sun αναπτύσσουν σταδιακά την εμπειρογνωμοσύνη τους στον τομέα της μηχανικής εκδόσεων ως ένα ανταγωνιστικό πλεονέκτημα.^[6]

Αλλά ακόμα και οι πιο επιτυχημένοι από αυτούς τους πωλητές διανομής Linux δεν θα επιτύχουν ποτέ τα έσοδα ή την κερδοφορία των σημερινών γιγάντων στον τομέα της ανάπτυξης λογισμικού

όπως είναι η Microsoft ή η Oracle, εκτός και αν επωφεληθούν από κάποια άλλα διδάγματα της ιστορίας. Όπως αποδεικνύεται τόσο από την αγορά του υλικού εξοπλισμού ηλεκτρονικών υπολογιστών όσο και από τον κλάδο ISP (που όπως αναφέρεται πιο πάνω είναι μια επιχείρηση παροχής υπηρεσιών που βασίζεται στα πρωτόκολλα προϊόντων και στις εφαρμογές του Internet), οι επιχειρήσεις εμπορικών προϊόντων έχουν στενό περιθώριο για την πλειονότητα των συμμετεχόντων. Αν οι εταιρείες δεν καταφέρουν να επιτύχουν μια υψηλή αξία μέσω της συσσώρευσης ή μέσω μιας ευκαιρίας τύπου «Intel Inside», πρέπει να ανταγωνιστούν μόνο μέσω της ταχύτητας και της ικανότητας αντίδρασης, και αυτός είναι ένας γεμάτος πρόκληση τρόπος διατήρησης ενός πλεονεκτήματος τιμής σε μια αγορά εμπορικών αγαθών.

Οι πρώτοι μελετητές της εμπορικής φύσης του Linux, όπως ο ιδρυτής της Red Hat, Bob Young, πίστευαν ότι το πλεονέκτημα μπορούσε να βρεθεί στη [δημιουργία μιας ισχυρής εμπορικής επωνυμίας](#). Είναι επίσης δυνατό, συμβατικοί κατασκευαστές, όπως η Flextronix, που εργάζονται στα παρασκήνια ως βιομηχανικοί προμηθευτές παρά ως οντότητες που λειτουργούν πίσω από μια επωνυμία και έρχονται σε άμεση επαφή με τους πελάτες, να παρέχουν μια καλύτερη αναλογία από τον Dell για ορισμένους πωλητές Linux.

Συμπερασματικά, το ίδιο το λογισμικό δεν αποτελεί πλέον το πρωταρχικό κέντρο δράσης στη βιομηχανία των ηλεκτρονικών υπολογιστών. Η εμπορευματοποίηση του λογισμικού δίνει αξία σε υπηρεσίες που ενεργοποιούνται από το λογισμικό αυτό. Απαιτούνται νέα επιχειρηματικά μοντέλα.



3 Συνεργασία μέσω δικτύου

Για να κατανοήσουμε τη φύση του ανταγωνιστικού πλεονεκτήματος στο νέο πρότυπο, θα πρέπει να στραφούμε όχι στο Linux, αλλά στο Internet, το οποίο έχει ήδη εμφανίσει δείγματα του πώς θα εξελιχθεί η ιστορία του ανοιχτού κώδικα.

Η πιο κοινή έκδοση της ιστορίας του ελεύθερου λογισμικού ξεκινάει με την επανάσταση κατά του αποκλειστικού λογισμικού που ξεκίνησε ο Richard Stallman το 1984 και η οποία είχε ηθικά κίνητρα. Είναι μια ενδιαφέρουσα ιστορία η οποία επικεντρώνεται σε μια χαρισματική μορφή, και οδηγεί σε μια εξιστόρηση στην οποία η άδεια την οποία έγραψε – το GPL – αποτελεί και το κεντρικό θέμα. Αλλά όπως και οι περισσότεροι υπέρμαχοι του ανοιχτού κώδικα, που συνηθίζουν να λένε γενικότερες ιστορίες σχετικά με την ανάπτυξη καλύτερων λογισμικών μέσα από τη διαφάνεια και την κοινή χρήση του κώδικα, προτιμώ να ξεκινήσω την ιστορία αυτή με το στυλ ανάπτυξης του λογισμικού που συνηθιζόταν στην πρώιμη βιομηχανία και ακαδημία του κλάδου των ηλεκτρονικών υπολογιστών. Επειδή το λογισμικό δεν θεωρούνταν η πρωταρχική πηγή αξίας, ο ανοιχτός κώδικας χρησιμοποιούνταν από όλους σε όλα τα επίπεδα της πρώιμης βιομηχανίας των ηλεκτρονικών υπολογιστών.

Η παράδοση του λογισμικού Unix προσφέρει ένα καλό παράδειγμα. Το Unix αναπτύχθηκε στο Bell Labs, και χρησιμοποιούνταν ελεύθερα από τους ερευνητές λογισμικού του πανεπιστημίου, οι οποίοι συνέβαλλαν στην ανάπτυξη πολλών εκ των λειτουργιών και των χαρακτηριστικών που σήμερα εκλαμβάνουμε ως δεδομένα. Το γεγονός ότι το Unix παρέχονταν με άδεια χρήσης η οποία στη συνέχεια επέτρεψε στην ATT να θέσει ένα τέλος στην ιστορία όταν αποφάσισε ότι ήθελε να εμπορευματοποιήσει το Unix, οδηγώντας τελικά στην αύξηση του BSD Unix και του Linux ως

δωρεάν εναλλακτικές λύσεις, δεν θα πρέπει να μας τυφλώνει μπροστά στο γεγονός ότι η πρώιμη, από κοινού ανάπτυξη *προηγήθηκε* της υιοθέτησης ενός μοντέλου άδειας χρήσης ανοιχτού κώδικα. Η εκχώρηση αδειών χρήσης του ανοιχτού κώδικα ξεκίνησε ως μια προσπάθεια διατήρησης μιας φιλοσοφίας από κοινού χρήσης, και μόνο αργότερα οδήγησε σε μια εκτεταμένη επίγνωση της αξίας αυτής της από κοινού χρήσης.

Για τις απαρχές του ανοιχτού κώδικα στην κοινότητα του Unix, μπορείτε να μελετήσετε τον ερευνητικό προσανατολισμό πολλών από τους αρχικούς συμμετέχοντες. Όπως σημείωσε ο Bill στο σύνθημά του στο [Συνέδριο Ανοιχτού Κώδικα του O'Reilly](#) το 1999, στην επιστήμη, πρέπει να μοιραζόμαστε τα δεδομένα που διαθέτουμε έτσι ώστε οι άλλοι να μπορέσουν να αναπαράγουν τα αποτελέσματά μας. Και στο Berkeley, ανέφερε, πιστεύαμε για τους εαυτούς μας πως ήμασταν επιστήμονες στον κλάδο των ηλεκτρονικών υπολογιστών.^[7]

Αλλά ίσως ακόμα πιο σημαντική ήταν η διαχωρισμένη φύση της πρώιμης αγοράς υλικού εξοπλισμού Unix. Με εκατοντάδες ανταγωνιστικές αρχιτεκτονικές ηλεκτρονικών υπολογιστών, ο μόνος τρόπος διανομής των λογισμικών ήταν υπό τη μορφή κώδικα! Κανείς δεν είχε πρόσβαση σε όλα τα μηχανήματα για να παράγει τα απαραίτητα δυαδικά. (Αυτό αποδεικνύει την ιδιοφυΐα ενός άλλου εκ των «νόμων» του Christensen, του νόμου της διατήρησης της αρθρωτής μορφής. Επειδή ο υλικός εξοπλισμός των ηλεκτρονικών υπολογιστών ήταν τυποποιημένος και αρθρωτός, ήταν αδύνατη η επίτευξη αξίας και μοναδικότητας στο λογισμικό. Αλλά επειδή ο υλικός εξοπλισμός του Unix ήταν μοναδικός και αποκλειστικός, το λογισμικό έπρεπε να είναι πιο ανοιχτό και αρθρωτό.)

Αυτή η φιλοσοφία ανταλλαγής του ανοιχτού κώδικα των λογισμικών αναπτύχθηκε από τις απαρχές των ερευνών, αλλά έγινε το τεκμήριο υψηλής ποιότητας ενός μεγάλου τμήματος της βιομηχανίας λογισμικών λόγω της αύξησης των δικτύων ηλεκτρονικών υπολογιστών.

Ένα μεγάλο μέρος του ρόλου του ανοιχτού κώδικα στην ανάπτυξη του Internet είναι ήδη αρκετά γνωστό: Η πιο ευρέως χρησιμοποιούμενη υλοποίηση πρωτοκόλλου TCP/IP αναπτύχθηκε ως μέρος του δικτύου Berkeley. Το Bind εκτελεί το DNS, χωρίς το οποίο δεν θα είχαμε πρόσβαση σε κανέναν από τους δικτυακούς τόπους από τους οποίους εξαρτόμαστε. Το sendmail είναι η καρδιά της κύριας δομής αποστολής ηλεκτρονικών μηνυμάτων του Internet. Το Apache είναι ο κύριος διακομιστής web. Η Perl η κύρια γλώσσα για τη δημιουργία δυναμικών δικτυακών τόπων, κτλ.

Λιγότερο γνωστός είναι ο ρόλος του Usenet στην ανάπτυξη του Δικτύου όπως το γνωρίζουμε σήμερα. Αυτό που οδήγησε στη δημόσια υιοθέτηση του Internet ήταν στην πραγματικότητα το Usenet, αυτός ο τεράστιος διανεμημένος ηλεκτρονικός πίνακας ανακοινώσεων. Εγγραφήκατε για το Usenet βρίσκοντας έναν γείτονα ο οποίος είναι πρόθυμος να σας δώσει τροφοδοσία άρθρων. Αυτό ήταν πραγματικά ένα δίκτυο συνεργασίας, όπου το ταχυδρομείο και τα νέα μεταβιβάζονταν από το έναν συνεργαζόμενο τόπο στον άλλο, ενώ συχνά απαιτούνταν ολόκληρες ημέρες για να μεταφερθούν από το ένα άκρο του Δικτύου στο άλλο. Οι τόποι ομφαλού (Hub) διαμόρφωσαν μια ad-hoc κύρια δομή, αλλά τα πάντα ήταν προαιρετικά.

Ο Rick Adams, που δημιούργησε το UUnet, το οποίο ήταν το πρώτο εμπορικό ISP, ήταν δημιουργός ελεύθερου λογισμικού (παρόλο που ποτέ δεν ενστερνίστηκε κανένα από τα ιδανικά περί του ελεύθερου λογισμικού – επρόκειτο απλά για έναν ταιριαστό τρόπο διανομής του λογισμικού που ήθελε να χρησιμοποιεί). Ήταν ο συγγραφέας του B News (ο κυρίαρχος διακομιστής νέων Usenet την εποχή εκείνη) καθώς και του SLIP (Σειριακή Γραμμή IP), της πρώτης υλοποίησης του TCP/IP για γραμμές κλήσης. Αλλά το πιο σημαντικό που έκανε ο Rick για την ιστορία του Δικτύου, ήταν να αναλάβει τον κύριο υπολογιστή υπηρεσίας του μεγαλύτερου ομφαλού Usenet του κόσμου. Συνειδητοποίησε ότι το προαιρετικό Usenet δεν ήταν πλέον λειτουργικό, και ότι οι άνθρωποι ήταν πρόθυμοι να πληρώσουν για μια αξιόπιστη πρόσβαση με

καλή σύνδεση. Το UUnet ξεκίνησε ως μια μη κερδοσκοπική δραστηριότητα, και για μερικά χρόνια, όλο και μεγαλύτερο μέρος της επιχειρηματικής δραστηριότητας βασιζόταν στο πρώιμο δίκτυο κλήσης UUCP (Πρωτόκολλο Αντιγραφής Unix-Unix) παρά στο TCP/IP. Καθώς το Internet συνέχισε να εξελίσσεται, η UUnet και άλλοι σαν και αυτή συνέβαλλαν στο να γίνει το Internet προσιτό στον ευρύτερο κόσμο. Αλλά στο τέλος, η εμπορική βιομηχανία του Internet ξεκίνησε από μια ανάγκη παροχής υποδομής για το πλήρως συνεργατικό UUCPnet και Usenet.

Το UUCPnet και Usenet χρησιμοποιήθηκαν για το email (η πρώτη ασυναγώνιστη εφαρμογή του Internet), αλλά επίσης για τη διανομή του λογισμικού και της συνεργατικής τεχνολογικής υποστήριξης. Όταν ο Larry Wall (ο οποίος έγινε αργότερα γνωστός ως ο συγγραφέας της Perl) εισήγαγε το πρόγραμμα επιδιόρθωσης το 1984, η δύσχρηστη διαδικασία αποστολής ταινιών ανοιχτού κώδικα 9-τροχιών αντικαταστάθηκε από τη μετάδοση «μπαλωμάτων» -- σεναρίων επεξεργασίας που ενημερώνουν τα υπάρχοντα αρχεία κώδικα. Προσθέστε τον μεταγλωττιστή Gnu C (gcc) του Richard Stallman, και τα πρώιμα συστήματα ελέγχου ανοιχτού κώδικα όπως το RCS (που τελικά αντικαταστάθηκε από το CVS και τώρα το Subversion), και έχετε μια διαδικασία όπου ο καθένας μπορεί να μοιραστεί και να ενημερώσει το ελεύθερο λογισμικό. Το πρώιμο Usenet ήταν κάτι σαν το «Napster» για λογισμικό στο οποίο γίνεται κοινή χρήση καθώς ήταν ένας τόπος για συζήτηση.

Οι μηχανισμοί που ανέπτυξαν οι πρώτοι προγραμματιστές για να διευρύνουν και να υποστηρίζουν την εργασία τους αποτέλεσε τη βάση για ένα πολιτισμικό φαινόμενο που ξεπέρασε τα όρια του τομέα της τεχνολογίας. Η καρδιά του φαινομένου αυτού ήταν η χρήση μιας τεχνολογίας δικτύου ευρείας περιοχής που συνέδεε τα άτομα γύρω από κοινά ενδιαφέροντα, παρά μέσω της γεωγραφικής τοποθεσίας ή της εταιρικής σχέσης. Αυτό ήταν η αρχή μιας μαζικής πολιτισμικής αλλαγής την οποία βλέπουμε ακόμα και σήμερα.

Αυτή η πολιτισμική αλλαγή μπορεί να έκανε για πρώτη φορά την εμφάνισή της με το λογισμικό ανοιχτού κώδικα, αλλά δεν συνδέεται άρρηκτα με τη χρήση των αδειών χρήσης και των φιλοσοφιών ελεύθερου και ανοιχτού κώδικα.

Το 1999, από κοινού με τον Brian Behlendorf του έργου Apache, ο O'Reilly ίδρυσε μια εταιρεία με την επωνυμία [CollabNet](#) για την εμπορευματοποίηση όχι του προϊόντος Apache αλλά της *διαδικασίας* Apache. Σε αντίθεση με πολλά άλλα έργα OSS, το Apache δεν ιδρύθηκε από έναν μόνο οραματιστή προγραμματιστή αλλά από μια ομάδα χρηστών που τους εγκατέλειψε ο αρχικός τους «πωλητής» (NCSA) και οι οποίοι συμφώνησαν να συνεργασθούν για να διατηρήσουν ένα εργαλείο πάνω στο οποίο βασίζονταν. Το Apache μας δίνει ένα καλό δίδαγμα σχετικά με την εκ συνεργασίας ανάπτυξη λογισμικού ευρείας περιοχής που μπορεί να εφαρμοστεί ακόμα και από εταιρείες που δεν έχουν υιοθετήσει πλήρως τις πρακτικές εκχώρησης αδειών για τη χρήση του ανοιχτού κώδικα. Για παράδειγμα, είναι δυνατή η εφαρμογή συνεργατικών αρχών ανοιχτού κώδικα στα πλαίσια μιας μεγάλης εταιρείας, ακόμα και χωρίς την πρόθεση έκδοσης του τελικού λογισμικού στο ευρύ κοινό.

Ενώ η CollabNet είναι περισσότερο γνωστή για την ανάληψη έργων ανοιχτού κώδικα υψηλού επιπέδου που χρηματοδοτούνται από εταιρείες, όπως το [OpenOffice.org](#), ο μεγαλύτερος πελάτης της είναι στην πραγματικότητα το τμήμα εκτυπωτών της HP, όπου η πλατφόρμα SourceCast της CollabNet χρησιμοποιείται για να βοηθηθούν περισσότεροι από 3000 εσωτερικοί προγραμματιστές στην κοινή χρήση του κώδικα μέσα στα πλαίσια του εταιρικού προστατευτικού τοίχους (firewall). Άλλοι πελάτες χρησιμοποιούν πρακτικές ανάπτυξης που βασίζονται στον ανοιχτό κώδικα για να μοιράζονται τον κώδικα με τους πελάτες τους ή τους επιχειρηματικούς τους συνεργάτες, ή για να διαχειρίζονται τις ομάδες ανάπτυξης που βρίσκονται σε όλα τα σημεία του κόσμου.

Αλλά μια ακόμα πιο ενδιαφέρουσα ιστορία έρχεται από το αρχέτυπο των αποκλειστικών λογισμικών, που δεν είναι άλλη από τη Microsoft. Πολύ λίγοι είναι εκείνοι που γνωρίζουν την ιστορία προέλευσης του ASP.NET. Όπως μου είπαν οι δημιουργοί του, Mark Anders και Scott Guthrie, οι δυο τους ήθελαν να σχεδιάσουν εκ νέου το προϊόν ASP της Microsoft προκειμένου αυτό να αναγνωρίζει την επεκτάσιμη γλώσσα σημείωσης XML. Τους είπαν ότι κάτι τέτοιο θα έσπαζε τη συμβατότητα με παλαιότερες εκδόσεις, και λήφθηκε η απόφαση να διατηρηθεί η παλαιά αρχιτεκτονική. Αλλά όταν οι Anders και Guthrie είχαν στη διάθεσή τους έναν μήνα μεταξύ των έργων, άρχισαν να εργάζονται πάνω στο όραμά τους, έτσι και αλλιώς, απλά για να δουν πού θα τους έβγαζε. Κάποια άλλα άτομα μέσα από τη Microsoft άκουσαν για το έργο τους, το θεώρησαν χρήσιμο και υιοθέτησαν κάποια τμήματα αυτού. Περίπου έξι ή εννέα μήνες αργότερα, έλαβαν ένα τηλεφώνημα από τον Bill Gates: «Θα ήθελα να δω το έργο».

Εν ολίγοις, ένα από τα πιο σημαντικά προϊόντα της Microsoft δημιουργήθηκε ως ένα εσωτερικό «σκέλος κώδικα», το αποτέλεσμα από την εργασία δύο προγραμματιστών που έκαναν το κέφι τους, και επεκτάθηκε μέσα στη Microsoft κατά τον ίδιο περίπου τρόπο που επεκτάθηκαν τα έργα ανοιχτού κώδικα στο ανοιχτό Internet. Φαίνεται ότι ο ανοιχτός κώδικας είναι η «φυσική γλώσσα» μιας δικτυωμένης κοινότητας. Εάν υπάρχει επαρκής αριθμός προγραμματιστών και ένα δίκτυο για να τους συνδέει, προκύπτει η συμπεριφορά ανάπτυξης τύπου ανοιχτού κώδικα.

Εάν θεωρήσετε ότι η εκχώρηση άδειας χρήσης του ανοιχτού κώδικα αποτελεί ένα μέσο ενθάρρυνσης της συνεργασίας μέσω Internet, και εστιάσετε στο αποτέλεσμα παρά στα μέσα που θα χρησιμοποιηθούν, θα καταφέρετε να πετύχετε μια ακόμα μεγαλύτερη διεύρυνση. Θα δείτε τους δεσμούς που ενώνουν όχι μόνο τα παραδοσιακά έργα ανοιχτού κώδικα, αλλά επίσης και τα συνεργατικά έργα «υπολογιστικού πλέγματος» όπως το [SETI@home](#), τις διεπαφές χρηστών στο [amazon.com](#), τεχνολογίες όπως το συνεργατικό φιλτράρισμα, νέες ιδέες σχετικά με την προώθηση σαν και αυτές που εκφράζονται στο [The Cluetrain Manifesto](#), weblogs, καθώς και τον τρόπο που οι πίνακες μηνυμάτων του Internet μπορούν τώρα να κινούν τα χρηματιστήρια. Αυτό που ξεκίνησε ως μεθοδολογία ανάπτυξης λογισμικού εξελίσσεται στη βασική μορφή κάθε πεδίου, καθώς οι συζητήσεις μέσω δικτύου καθίστανται το κύριο μέσο μεταφοράς των νέων ιδεών.

Μου έχει κάνει ιδιαίτερη εντύπωση πώς η συνεργασία είναι βασικής σημασίας για την επιτυχία και τη διαφοροποίηση των μεγαλύτερων εφαρμογών του Internet.

Το EBay αποτελεί ένα εμφανές παράδειγμα, και θα λέγαμε ότι είναι σχεδόν ο ορισμός μιας επιχείρησης δικτύου, όπου το ανταγωνιστικό αποτέλεσμα επιτυγχάνεται από τη σημαντικότερη μάζα των αγοραστών και των πωλητών. Οι νέες εταιρείες που εισήχθησαν στο χρηματιστήριο δυσκολεύθηκαν να αντεπεξέλθουν στον ανταγωνισμό, διότι δεν υπάρχει λόγος για τους αγοραστές και τους πωλητές να απευθυνθούν σε έναν παίκτη δεύτερης κατηγορίας.

Το Amazon αποτελεί πιθανώς μια ακόμα πιο ενδιαφέρουσα περίπτωση. Σε αντίθεση με το eBay, το σύνολο προϊόντων του οποίου παρέχεται από τους χρήστες, και αλλάζει δυναμικά από τη μια ημέρα στην άλλη, προϊόντα πανομοιότυπα με αυτά που πωλούνται στο Amazon υπάρχουν διαθέσιμα σε άλλους πωλητές. Και όμως το Amazon φαίνεται να απολαμβάνει ένα καθαρό πλεονέκτημα σε σχέση με τους άλλους πωλητές. Γιατί; Ίσως επειδή έχει να παρουσιάσει καλύτερη εκτέλεση, καλύτερες τιμές, καλύτερη εξυπηρέτηση και καλύτερες μάρκες. Αλλά αυτό που το διαφοροποιεί περισσότερο είναι ο ανώτερος τρόπος με τον οποίο το Amazon έχει αξιοποιήσει την κοινότητα των χρηστών.

Στις ομιλίες μου, συνηθίζω να κάνω μια απλή παρουσίαση. Κάνω μια αναζήτηση προϊόντων σε έναν από τους τόπους δημοσίευσης, σε JavaScript. Στο [amazon.com](#), από την αναζήτηση προκύπτει μια σύνθετη σελίδα με τέσσερις κύριες περιοχές. Στο πάνω μέρος εμφανίζεται ένα τμήμα που δείχνει τα τρία «πιο δημοφιλή» προϊόντα. Ακριβώς από κάτω υπάρχει ένας

μεγαλύτερος κατάλογος αναζήτησης που επιτρέπει στον πελάτη να καταχωρεί τα προϊόντα βάσει κριτηρίων όπως είναι οι καλύτερες πωλήσεις, η υψηλότερη βαθμολογία αξιολόγησης, η τιμή ή γίνεται απλά μια αλφαβητική καταγραφή. Στα δεξιά και στα αριστερά υπάρχουν οι κατάλογοι «ListMania» που δημιουργούνται από τον ίδιο τον χρήστη. Οι κατάλογοι αυτοί επιτρέπουν στους πελάτες να μοιράζονται τις συστάσεις τους για άλλους τίτλους σε σχέση με ένα συγκεκριμένο θέμα.

Η ενότητα που φέρει τον τίτλο «most popular» (πιο δημοφιλή) μπορεί στην αρχή να μην εμφανιστεί. Αλλά ως πωλητής που πωλεί στο *amazon.com*, γνωρίζω ότι αυτό είναι το αποτέλεσμα ενός σύνθετου, αποκλειστικού αλγόριθμου που συνδυάζει όχι μόνο τις πωλήσεις αλλά και τον αριθμό και την ποιότητα των κριτικών από την πλευρά των χρηστών, τις συστάσεις των χρηστών για εναλλακτικά προϊόντα, τις συνδέσεις από τους καταλόγους ListMania, συνδέσεις «also bought» (αγοράστηκαν επίσης), και όλα τα άλλα πράγματα στα οποία το Amazon αναφέρεται ως η «ροή» γύρω από τα προϊόντα.

Η συγκεκριμένη έρευνα που προτιμώ να παρουσιάζω εμφανίζει στην κορυφή το δικό μου έργο [JavaScript: The Definitive Guide](#). Το βιβλίο έχει 192 κριτικές, που αντιστοιχούν σε μέσο όρο 4 1/2 αστεράκια. Οι κριτικές αυτές είναι μεταξύ των *περισσότερων από δέκα εκατομμύρια* κριτικές χρηστών που έχουν δώσει οι πελάτες του *amazon.com*.

Τώρα συγκρίνετε τον παίκτη #2 στα online βιβλία, *barnesandnoble.com*. Το πρώτο αποτέλεσμα είναι ένα βιβλίο που έχει δημοσιευθεί από το ίδιο το Barnes & Noble, και δεν υπάρχει καμία ένδειξη περιεχομένου που να έχει δοθεί από τους χρήστες. Το *JavaScript: The Definitive Guide* έχει μόνο 18 σχόλια, ενώ η διαφορά στη συμμετοχή των χρηστών αντικατοπτρίζει τη διαφορά στις πωλήσεις.

Το Amazon δεν διαθέτει ένα φυσικό πλεονέκτημα ως προς το δίκτυο όπως το eBay, αλλά η ανάπτυξη του βασίστηκε σε μια αρχιτεκτονική που επιτρέπει τη συμμετοχή των χρηστών. Τα πάντα από τις κριτικές των χρηστών, τις συστάσεις εναλλακτικών προϊόντων, το ListMania, και το πρόγραμμα Εταίρων, που επιτρέπουν στους χρήστες να κερδίζουν προμήθειες με το να συστήνουν βιβλία, ενθαρρύνουν τους χρήστες να συνεργάζονται για τη βελτίωση του δικτυακού αυτού τόπου. Το Amazon Web Services, που εισήχθη το 2001, για να συνεχίσουμε ακόμα περισσότερο με την ιστορία, επιτρέπει στους χρήστες να αναπτύσσουν εναλλακτικές διεπαφές και εξειδικευμένες εμπειρίες αγορών (καθώς και άλλες μη αναμενόμενες εφαρμογές) μέσα από τη χρήση των δεδομένων του Amazon και της εμπορικής μηχανής ως οπισθοφυλακή.

Η απόσταση που χωρίζει το Amazon από τους ανταγωνιστές του, και η ασφάλεια που απολαμβάνει ως κυρίαρχος της αγοράς, οφείλεται στην αξία που προστίθεται από τους ίδιους του χρήστες. Εάν, όπως είπε ο Eric Raymond στο *The Cathedral & the Bazaar*, ένα από τα μυστικά του ανοιχτού κώδικα είναι η «μεταχείριση των χρηστών σας ως συν-προγραμματιστές», το Amazon το γνωρίζει το μυστικό αυτό. Θα πρέπει να σημειωθεί όμως ότι είναι τελείως ανεξάρτητο από πρακτικές εκχώρησης άδειας χρήσης ανοιχτού κώδικα ! Αρχίζουμε να βλέπουμε αυτό που παρουσιάζεται ως ένα αυστηρά περιορισμένο μοντέλο για τον ανοιχτό κώδικα γιατί ο ανοιχτός κώδικας μπορεί να αποτελείται από ένα σύνολο χαρακτηριστικών, τα οποία δεν είναι δυνατό να υπάρχουν πάντα όλα μαζί.



Το Google αποτελεί μια διαφορετική περίπτωση ως προς την ανάπτυξη και τη χρήση δικτύων. Η αρχική καινοτομία του Google ήταν ο αλγόριθμος PageRank, που αξιοποιεί τις συλλογικές προτιμήσεις των χρηστών του ιστού, οι οποίες εκφράζονται από τις υπερσυνδέσεις στους δικτυακούς τόπους, προκειμένου να παραχθούν καλύτερα αποτελέσματα αναζήτησης. Στην περίπτωση του Google, η συμμετοχή των χρηστών είναι άσχετη με την εταιρεία και τα προϊόντα της, και έτσι μπορεί να αντιγραφεί από τους ανταγωνιστές. Εάν η ανάλυση αυτή είναι σωστή, η μακροπρόθεσμη επιτυχία του Google θα εξαρτάται από την εξεύρεση επιπλέον τρόπων εκμετάλλευσης της αξίας που δημιουργούν οι χρήστες ως ένα βασικό μέρος της προσφοράς τους. Υπηρεσίες όπως το [orkut](#) και το [Gmail](#) φανερώνουν ότι κάτι τέτοιο έχει γίνει απόλυτα αντιληπτό και κατανοητό.

Ας δούμε ένα άλλο παράδειγμα. Η MapQuest είναι μια άλλη πρωτοποριακή εταιρεία που δημιούργησε έναν καινοτομικό τύπο εφαρμογής web στον οποίο βασίζεται σχεδόν κάθε χρήστης Internet. Και όμως η αγορά μοιράζεται δίκαια μεταξύ της MapQuest (την οποία τώρα κατέχει η AOL), [maps.yahoo.com](#), και [maps.msn.com](#) (τα οποία λειτουργούν από την MapPoint). Και οι τρεις αυτές εταιρείες παρέχουν εμπορικά προϊόντα που λειτουργούν με τυποποιημένο λογισμικό και βάσεις δεδομένων. Καμία εξ αυτών δεν αντιμετωπίζει ιδιαίτερη δυσκολία στην αξιοποίηση του περιεχομένου που παρέχεται από τους χρήστες, ή στην απασχόληση των χρηστών τους στην ανάπτυξη της εφαρμογής. (Θα πρέπει να σημειωθεί επίσης πως και οι τρεις παρέχουν μια ευκαιρία τύπου Intel-Inside για παροχές δεδομένων, όπως η NAVTEQ, που τώρα σχεδιάζουν ένα IPO πολλών δισεκατομμυρίων δολαρίων!)

4 Η Αρχιτεκτονική της Συμμετοχής

Χρησιμοποιώ τον όρο «η αρχιτεκτονική της συμμετοχής» για να περιγράψω τη φύση των συστημάτων που είναι σχεδιασμένα για τη συμβολή του χρήστη. Το βιβλίο του Larry Lessig, [Code and Other Laws of Cyberspace](#), που το χαρακτηρίζει ως μια εκτεταμένη μελέτη πάνω στο θεώρημα του Mitch Kapor, «η αρχιτεκτονική είναι συνώνυμο της πολιτικής», υπογραμμίζει ότι χρειάζεται να δώσουμε προσοχή στην αρχιτεκτονική των συστημάτων εάν θέλουμε να κατανοήσουμε τη λειτουργία τους.

Αμέσως σκέφτηκα την περιγραφή των Kernighan και Pike για τη φιλοσοφία που αφορά στα εργαλεία λογισμικού Unix [που αναφέρεται πιο πάνω](#). Ο Linus εξέφρασε επίσης την άποψη ότι η αρχιτεκτονική μπορεί να είναι πιο σημαντική από τον ανοιχτό κώδικα. «Δεν θα μπορούσα να κάνω αυτό που έκανα με τα Linux για Windows, ακόμα και αν είχα τον πηγαίο κώδικα. Η αρχιτεκτονική απλά δεν θα μπορούσε να το υποστηρίξει.» Ένα πολύ μεγάλο μέρος του πηγαίου κώδικα των windows συνίσταται σε ανεξάρτητα, καλά συνδεδεμένα επίπεδα που ένας μόνο προγραμματιστής μπορεί να ρίξει σε μια υπομονάδα αντικατάστασης.

Και φυσικά, το Internet και το World Wide Web έχουν βασιστεί επίσης σε αυτή τη συμμετοχική αρχιτεκτονική. Όπως υπογραμμίζεται πιο πάνω στην ενότητα σχετικά με την [εμπορευματοποίηση του λογισμικού](#), οποιοδήποτε σύστημα είναι σχεδιασμένο γύρω από τα πρωτόκολλα επικοινωνίας είναι σχεδιασμένο έτσι ώστε να επιτρέπει τη συμμετοχή. Οποιοσδήποτε μπορεί να δημιουργήσει ένα συμμετοχικό, πρώτης κατηγορίας στοιχείο.

Επιπλέον, η ομάδα εργασίας μελέτης Internet ([IETF](#)), για την επεξεργασία των προτύπων Internet, έχει πολλές ομοιότητες με ένα έργο λογισμικού ανοιχτού κώδικα. Η μόνη ουσιαστική διαφορά είναι ότι το αποτέλεσμα της IETF είναι ένα έγγραφο προτύπων παρά μια υπομονάδα κώδικα. Ιδιαίτερα κατά τα τελευταία χρόνια, μπορούσε να συμμετέχει ο οποιοσδήποτε, κάνοντας απλά μια εγγραφή και εκφράζοντας κάποιες απόψεις ή συμμετέχοντας σε μία από τις τρεις ετήσιες συνεδριάσεις. Τα πρότυπα αποφασίστηκαν από τα άτομα που συμμετείχαν, ανεξάρτητα από τις εταιρικές σχέσεις. Το όνομα για τα προτεινόμενα πρότυπα Internet, RFCs (Πρόσκληση υποβολής

σχολίων), αποτελεί ένα δείγμα της συμμετοχικότητας στο Δίκτυο. Παρόλο που η εμπορική συμμετοχή ενθαρρυνόταν και ήταν κάτι το επιθυμητό, οι εταιρείες, όπως και οι ιδιώτες, συναγωνίζονταν ως προς τις ιδέες τους και τις υλοποιήσεις τους, και όχι ως προς τα χρήματα ή τη δυσανάλογη παρουσίαση. Η προσέγγιση της IETF είναι το σημείο όπου συναντώνται ο ανοιχτός κώδικας και τα ανοιχτά πρότυπα.

Και ενώ υπάρχουν επιτυχημένα έργα ανοιχτού κώδικα όπως το Sendmail, που αποτελούν σε γενικές γραμμές τη δημιουργία ενός και μόνο ατόμου, και έχουν μονολιθική αρχιτεκτονική, αυτοί που δημιούργησαν μεγάλες κοινότητες ανάπτυξης το έκαναν διότι έχουν μια αρθρωτή αρχιτεκτονική που επιτρέπει την εύκολη συμμετοχή από έναν ανεξάρτητο ή χαλαρά συντονισμένους προγραμματιστές. Η χρήση της Perl, για παράδειγμα, έφτασε στα ύψη μαζί με το [CPAN](#), το Comprehensive Perl Archive Network, και το αρθρωτό σύστημα της Perl, που επέτρεπε στον καθένα να ενισχύει τη γλώσσα με ειδικευμένες λειτουργίες, και να τις κάνει διαθέσιμες σε άλλους χρήστες.

Το web, ωστόσο, είχε την ιδέα της συμμετοχής σε ένα νέο επίπεδο, διότι άνοιγε τη συμμετοχή αυτή όχι μόνο στους προγραμματιστές λογισμικού αλλά και σε όλους τους χρήστες του συστήματος.

Από πάντα με εξέπληττε και με απογοήτευε το γεγονός ότι η κοινότητα ανοιχτού κώδικα δεν έχει ανακηρύξει το web ως μία από τις μεγαλύτερες ιστορίες επιτυχίας. Εάν ρωτήσετε τους περισσότερους τελικούς χρήστες, είναι πολύ πιθανό ότι θα συνδέσουν το web με τους αποκλειστικούς πελάτες όπως ο Internet Explorer της Microsoft παρά με την επαναστατική αρχιτεκτονική ανοιχτού κώδικα που κατέστησε το web δυνατό. Αυτό είναι μια αποτυχία των δημοσίων σχέσεων! Η αρχική υλοποίηση του web από τον Tim Berners-Lee δεν ήταν απλά ανοιχτός κώδικας, αλλά κοινός τόπος. Ο web server της NCSA και ο Mosaic browser δεν ήταν από τεχνικής απόψεως ανοιχτού κώδικα, αλλά η πηγή ήταν ελεύθερα διαθέσιμη. Ενώ η ομάδα NCSA στο Netscape προσπαθούσε να περάσει βασικά τμήματα της υποδομής web στην αποκλειστική πλευρά, και οι μάχες του Microsoft-Netscape έκαναν να φαίνεται ότι το web ήταν πρωτίστως ένα αποκλειστικό λογισμικό, θα έπρεπε να δείχθει μεγαλύτερη προσοχή. Το Apache, αυτό το μοναδικό στοιχείο που αναπτύχθηκε από τον διακομιστή NCSA, διατήρησε ζωντανό το όραμα του ανοιχτού κώδικα, διατηρώντας την ακεραιότητα των προτύπων, και μην υποχωρώντας σε στρατηγικές αποκλειστικής υιοθέτησης και επέκτασης.

Ένα άλλο επίσης σημαντικό γεγονός είναι ότι η HTML, η γλώσσα των ιστοσελίδων, έγινε προσιτή προς χρήση και από τους τυπικούς χρήστες, και όχι μόνο από τους προγραμματιστές λογισμικών. Το στοιχείο μενού «View Source» (Εμφάνιση πηγής) έχει μεταφερθεί από τον αρχικό browser του Tim Berners-Lee, στο Mosaic, και στη συνέχεια στο Netscape Navigator ακόμα και στο Internet Explorer της Microsoft. Παρόλο που κανείς δεν εκλαμβάνει τη γλώσσα HTML ως μια τεχνολογία ανοιχτού κώδικα, το χαρακτηριστικό της αυτό είχε αποφασιστική σημασία για την εκρηκτική εξάπλωση του web. Τα εμπόδια για τη συμμετοχή «ερασιτεχνών» ήταν χαμηλά, διότι ο καθένας μπορούσε να «κοιτάξει πάνω από τον ώμο του άλλου» και να δει την ιστοσελίδα του και πώς τη δημιούργησε. Το δυναμικό περιεχόμενο που δημιουργείται με τις ερμηνευόμενες γλώσσες διατήρησε την τάση προς τη διαφάνεια.

Και πιο συναφές με το επιχείρημά μου στο σημείο αυτό, είναι το γεγονός ότι η βασική αρχιτεκτονική της δημιουργίας και λειτουργίας υπερσυνδέσεων διασφαλίζει ότι η αξία του web αναπτύσσεται από τους χρήστες της.

Σε αυτό το πλαίσιο, αξίζει να σημειωθεί μια παρατήρηση που έγινε αρχικά από τον Clay Shirky σε μια ομιλία στο [Συνέδριο για τις Υπηρεσίες Web και P2P](#) (τώρα ονομάζεται [Συνέδριο Ανερχόμενης Τεχνολογίας](#)), με τίτλο «Ακούγοντας Napster.» Υπάρχουν τρεις τρόποι για να αναπτύξει κανείς

μία μεγάλη βάση δεδομένων, είπε ο Clay. Ο πρώτος, που χρησιμοποιήθηκε από το Yahoo!, είναι η πρόσληψη ατόμων για την κατασκευή της βάσης. Ο δεύτερος τρόπος, ο οποίος γεννήθηκε από τα διδάγματα της κοινότητας ανοιχτού κώδικα, είναι ο διορισμός εθελοντών για την εκτέλεση της ίδιας εργασίας. Το αποτέλεσμα είναι το [Open Directory Project](#), ένας ανταγωνιστής ανοιχτού κώδικα του Yahoo!. (Η [Wikipedia](#) αποτελεί ένα ακόμα παράδειγμα.) Αλλά το [Napster](#) παρουσιάζει έναν τρίτο τρόπο. Επειδή το Napster βασίστηκε στην αυτόματη κοινή χρήση μουσικών κομματιών, κάθε χρήστης βοηθούσε αυτομάτως στην ανάπτυξη της αξίας της κοινής βάσης δεδομένων.

Το αρχιτεκτονικό στοιχείο μπορεί να παίζει πιο κεντρικό ρόλο στον ανοιχτό κώδικα από τον πιο συχνό εθελοντισμό. Η αρχιτεκτονική του Linux, του Internet, και του World Wide Web είναι τέτοια που οι χρήστες που επιζητούν να ικανοποιήσουν τα δικά τους «εγωιστικά» συμφέροντα δίνουν συλλογική αξία ως ένα αυτόματο υποπροϊόν. Με άλλα λόγια, οι τεχνολογίες αυτές εμφανίζουν ομοιότητες ως προς τα δίκτυα, όπως στην περίπτωση του eBay και του Napster, απλά μέσω του τρόπου με τον οποίο έχουν σχεδιαστεί.

Τα έργα αυτά φαίνεται πως έχουν μια φυσική αρχιτεκτονική συμμετοχής. Αλλά όπως αποδεικνύεται από την εμπειρία του Amazon, με συνεχείς προσπάθειες (καθώς και οικονομικά κίνητρα όπως το Πρόγραμμα Εταίρων), είναι δυνατή η ανάπτυξη μιας τέτοιας αρχιτεκτονικής σε ένα σύστημα που υπό άλλες συνθήκες δεν θα μπορούσε να διαθέτει.

5 Δυνατότητα Προσαρμογής και Λογισμικό ως Υπηρεσία

Το τελευταίο στοιχείο είναι η δυνατότητα προσαρμογής, που αποτελεί ένα ουσιαστικό συνεπακόλουθο στοιχείο του λογισμικού ως υπηρεσία. Είναι ιδιαίτερα σημαντικό να τονίσουμε την πλευρά αυτή διότι φανερώνει ακριβώς το λόγο γιατί δυναμικά δημιουργημένες γλώσσες όπως οι Perl, Python, και PHP, που τόσο συχνά απαξιώνονται από τους προγραμματιστές λογισμικών του παλαιού προτύπου ως απλές «γλώσσες σεναρίων», είναι τόσο σημαντικές στο γενικότερο σκηνικό των λογισμικών σήμερα.

Όπως έγραψα στη μελέτη με τίτλο [Hardware, Software and Infoware](#) (Υλικός εξοπλισμός, Λογισμικό και Infoware), του 1997:

Εάν κοιτάξετε έναν μεγάλο δικτυακό τόπο σαν το Yahoo!, θα δείτε ότι στα παρασκήνια, ένας στρατός διαχειριστών και προγραμματιστών ανασχηματίζουν συνεχώς το προϊόν. Το δυναμικό περιεχόμενο δεν δημιουργείται αυτομάτως, συχνά δημιουργείται επίσης με δυνατότητες προσαρμογής, χρησιμοποιώντας τυπικά μια σειρά από γρήγορα και ακάθαρτα εργαλεία σεναρίων.

«Δεν δημιουργούμε περιεχόμενο στο Yahoo! Αυτό που κάνουμε είναι να συναθροίζουμε περιεχόμενο», είπε ο Jeffrey Friedl, συγγραφέας του βιβλίου [Mastering Regular Expressions](#) και προγραμματιστής πλήρους απασχολήσεως Perl στο Yahoo! «Λαμβάνουμε τροφοδότηση από χιλιάδες πηγές, η κάθε μία με τη δική της μορφή. Διενεργούμε ένα τεράστιο έργο «επεξεργασίας τροφοδότησης» για να ξεκαθαρίσουμε όλο αυτό το περιεχόμενο ή για να βρούμε με ποιο τρόπο θα το εισάγουμε στο Yahoo!» Για παράδειγμα, για να εισάγει νέες ιστορίες στο *finance.yahoo.com*, ο Friedl χρειάστηκε να γράψει ένα πρόγραμμα «αναγνώρισης ονόματος» που θα είναι σε θέση να κάνει αναζήτηση σε πάνω από 15.000 ονόματα εταιρειών. Η ικανότητα της Perl να αναλύει κείμενο ελεύθερης μορφής με ισχυρές τακτικές παραστάσεις κατέστησε δυνατή την πραγματοποίηση του σχεδίου.

Η Perl αναφέρεται ως η «κολλητική ταινία του Internet», και όπως η κολλητική ταινία, οι δυναμικές γλώσσες όπως η Perl είναι σημαντικές για τους δικτυακούς τόπους όπως το Yahoo! και το Amazon για τον ίδιο λόγο που η κολλητική ταινία είναι σημαντική όχι μόνο για τους τεχνικούς

συστημάτων θέρμανσης όσο και για τον κάθε ένα που επιθυμεί να συγκρατήσει μια ταχέως μεταβαλλόμενη εγκατάσταση. Πηγαίνετε σε οποιαδήποτε διάλεξη ή θεατρικό έργο, και θα δείτε καλώδια μικροφώνου και άλλα συστήματα καλωδίωσης που συγκρατούνται με τη χρήση κολλητικής ταινίας.

Έχουμε συνηθίσει να εκλαμβάνουμε το λογισμικό ως τέχνημα παρά ως διαδικασία. Και για να σας βεβαιώσω γι' αυτό, ακόμα και στο νέο πρότυπο, υπάρχουν τεχνήματα λογισμικού, προγράμματα και εμπορικά στοιχεία τα οποία πρέπει να υποστούν την κατάλληλη επεξεργασία για να τηρούν τις προδιαγραφές διότι θα χρησιμοποιηθούν πολλές φορές ακόμα. Αλλά ο τομέας των λογισμικών *δεν* έχει εμπορευματοποιηθεί, η «κόλλα» που συνδέει τα στοιχεία μεταξύ τους, τα σενάρια για τη διαχείριση δεδομένων και μηχανημάτων, και όλοι οι τομείς που χρειάζονται συχνές αλλαγές ή ταχύ καθορισμό αρχέτυπου, που κάνουν τις δυναμικές γλώσσες να ξεχωρίζουν.

Δικτυακοί τόποι όπως το Google, το Amazon, ή το eBay – ιδιαίτερα εκείνοι που φανερώνουν τη δυναμική της συμμετοχής των χρηστών – δεν αποτελούν απλά προϊόντα, αλλά στην ουσία πρόκειται για *διαδικασίες*.

Μου αρέσει να διηγούμαι την ιστορία του Μηχανικού Τούρκου, μια απάτη του 1770 που υποτίθεται ότι ήταν μια μηχανοκίνητη συσκευή που έπαιζε σκάκι. Το μυστικό, φυσικά, ήταν ότι από κάτω βρισκόταν κρυμμένος ένας άνθρωπος. Ο Τούρκος στην πραγματικότητα έπαιξε έναν μικρό ρόλο στην ιστορία των ηλεκτρονικών υπολογιστών. Όταν ο Charles Babbage έπαιξε ενάντια στον Τούρκο το 1820 (και έχασε), είδε μέσα από το μηχανήμα, αλλά του γεννήθηκε η περιέργεια να δει κατά πόσο ένα πραγματικό υπολογιστικό μηχανήμα θα ήταν δυνατό να κατασκευαστεί.

Τώρα, σε έναν ειρωνικό κύκλο, οι εφαρμογές για άλλη μια φορά κρύβουν μέσα τους τον άνθρωπο. Πάρτε ένα αντίγραφο του Microsoft Word και έναν συμβατό υπολογιστή, και θα μπορείτε να εργαστείτε σε αυτό ακόμα και μετά από δέκα χρόνια. Αλλά χωρίς να συμπεριλάβουμε τις συνεχείς προσπάθειες ανανέωσης των μηχανών αναζήτησης, τις συνεχείς ενημερώσεις προϊόντων σε δικτυακούς τόπους όπως το Amazon ή το eBay, οι διαχειριστές που είναι υπεύθυνοι για τη λειτουργία, οι αναθεωρητές και οι σχεδιαστές που ενσωματώνουν στη διεπαφή το περιεχόμενο που παρέχεται από τους πωλητές και τους χρήστες, και στην περίπτωση ορισμένων δικτυακών τόπων, ακόμα και το προσωπικό αποθηκών που παραδίδει τα προϊόντα, η εφαρμογή της εποχής του Internet δεν εκτελεί πλέον τη λειτουργία της.

Πραγματικά δεν πρόκειται για την επιχείρηση λογισμικού που υπήρχε πριν από μια δεκαετία. Φυσικά, υπήρχαν πάντα εταιρείες λογισμικού με το χαρακτηριστικό αυτό. (Το σύστημα κρατήσεων, [Sabre](#), της American Airlines αποτελεί εμφανές παράδειγμα.) Αλλά μόλις τώρα καθίστανται κυρίαρχο πρότυπο για τις νέες επιχειρήσεις που σχετίζονται με τους ηλεκτρονικούς υπολογιστές.

Η πρώτη γενιά οποιασδήποτε νέας τεχνολογίας εκλαμβάνεται τυπικά ως μια επέκταση των προηγούμενων γενεών. Με τη λογική αυτή, κατά τη διάρκεια της δεκαετίας του 1990, η πλειονότητα των ανθρώπων βίωσε το Internet ως μια επέκταση ή μια προσθήκη των προσωπικών υπολογιστών. Είναι βέβαιο ότι το σύστημα του ηλεκτρονικού ταχυδρομείου (Email) και η φυλλομέτρηση του ιστού (web browsing) ήταν ισχυρές προσθήκες, και έδωσαν μια επιπλέον ώθηση στη βιομηχανία των ηλεκτρονικών υπολογιστών.

(Οι υπέρμαχοι του ανοιχτού κώδικα μπορούν να υπογραμμίσουν με τρόπο ειρωνικό το γεγονός ότι πολλά από τα πιο σημαντικά χαρακτηριστικά των εκδόσεων του νέου λειτουργικού συστήματος της Microsoft από την εποχή των Windows 95 έχουν σχεδιαστεί κατά τρόπο ώστε να συναγωνίζονται τη λειτουργικότητα του Internet που δημιουργήθηκε αρχικά από τους προγραμματιστές ανοιχτού κώδικα.)

Αλλά τώρα, αρχίζουμε να διακρίνουμε το σχήμα ενός πολύ διαφορετικού μέλλοντος. Το Napster μας έφερε την ομότιμη χρήση αρχείων, το Seti@home εισήγαγε εκατομμύρια άτομα στην ιδέα του διανεμημένου υπολογισμού, και τώρα οι υπηρεσίες web αρχίζουν να δημιουργούν ακόμα και δικτυακούς τόπους με τεράστιες βάσεις δεδομένων, όπως το Amazon ή το Google που φαίνεται ότι λειτουργούν ως στοιχεία ενός ακόμα μεγαλύτερου συστήματος. Οι πωλητές όπως η IBM και η HP χρησιμοποιούν όρους όπως «κατά παραγγελία υπολογισμός» και «διεισδυτικός υπολογισμός».

Τα όρια μεταξύ των κινητών τηλεφώνων, των ασύρματων φορητών υπολογιστών, και ακόμα και των συσκευών καταναλωτών όπως το iPod ή το TiVO, δεν μπορούν να καθοριστούν σαφώς. Κάθε ένα λαμβάνει τώρα ένα μεγάλο μέρος της αξίας του από το λογισμικό που βρίσκεται αλλού. Ο Dave Stutz το χαρακτηρίζει αυτό ως το [λογισμικό πάνω από το επίπεδο μιας μεμονωμένης συσκευής](#).^[8]

6 Ανάπτυξη του Λειτουργικού Συστήματος του Internet

Μου αρέσει να λέω ότι περνάμε σε ένα στάδιο όπου θα μεταχειριζόμαστε το Internet σαν να επρόκειτο για έναν μεμονωμένο πραγματικό ηλεκτρονικό υπολογιστή. Για να το κάνουμε αυτό, θα χρειαστεί να δημιουργήσουμε ένα λειτουργικό σύστημα για το Internet.

Το μεγάλο ερώτημα που τίθεται είναι το εξής: Τι είδους λειτουργικό σύστημα πρόκειται να είναι; Το δίδαγμα της Microsoft είναι ότι εάν δει κανείς βαθύτερα στο νέο πρότυπο, θα διαπιστώσει το μυστικό που θα του δώσει τον απόλυτο έλεγχο της βιομηχανίας. Οι ανταγωνιστές ο ένας μετά τον άλλον καταβάλλουν κάθε δυνατή προσπάθεια για να εκθρονίσουν τη Microsoft και να πάρουν αυτοί την πρώτη θέση, αλλά κάθε προσπάθεια οδηγεί στην αποτυχία. Το δίδαγμα του ανοιχτού κώδικα και του Internet είναι ότι μπορούμε να αναπτύξουμε ένα λειτουργικό σύστημα που είναι σχεδιασμένο εξ ολοκλήρου ως «μικρά κομμάτια χαλαρά συνδεδεμένα μεταξύ τους», με μια αρχιτεκτονική που καθιστά εύκολη τη συμμετοχή οποιουδήποτε στην ανάπτυξη της αξίας του συστήματος.

Οι αξίες της κοινότητας του ελεύθερου και ανοιχτού κώδικα αποτελούν σημαντικό μέρος του προτύπου αυτού. Όπως ακριβώς η επανάσταση του Κοπέρνικου αποτελούσε μέρος μιας ευρύτερης κοινωνικής επανάστασης που απομάκρυνε την κοινωνία από την ιεραρχία και ήταν υπέρ της γνώσης, και επίσης κατάφερε να υποδαυλίσει ένα πνεύμα συστηματικής έρευνας και εξάπλωσης των γνώσεων, ο ανοιχτός κώδικας αποτελεί μέρος μιας επανάστασης στον τομέα της επικοινωνίας καθώς είναι σχεδιασμένος να μεγιστοποιεί την ελεύθερη εξάπλωση των ιδεών που εκφράζονται υπό μορφή κώδικα.

Αλλά οι υπέρμαχοι του ελεύθερου λογισμικού προχωρούν πολύ όταν απορρίπτουν κάθε όριο στην κοινή χρήση, και ορίζουν το κίνημα προσκολλώμενοι σε ένα περιοριστικό σύνολο πρακτικών εκχώρησης άδειας χρήσης λογισμικών. Το κίνημα του ανοιχτού κώδικα έχει καταβάλλει μια προσπάθεια να είναι πιο περιεκτικό. Ο Eric Raymond περιγράφει τον [Ορισμό του Ανοιχτού Κώδικα](#) ως μια «πρόκληση της σκέψης», ένα «κοινωνικό συμβόλαιο . . . και μια πρόσκληση συμμετοχής στο δίκτυο από εκείνους εμμένουν σε αυτό.»^[9] Αλλά παρότι το κίνημα του ανοιχτού κώδικα είναι πολύ πιο φιλικό προς τις επιχειρήσεις και υποστηρίζει το δικαίωμα των προγραμματιστών να επιλέγουν μη-ελεύθερες άδειες χρήσης, εξακολουθεί να χρησιμοποιεί την παρουσία των αδειών χρήσης λογισμικού που καθιστούν την κοινή χρήση ως το βασικό κριτήριο.

Τα διδάγματα των μεταβολών του προηγούμενου προτύπου μας εμφανίζουν μια πιο περίπλοκη και ισχυρή ιστορία από εκείνη που απλά αντιπαραβάλλει μια φιλοσοφία χαρισμάτων έναντι μια οικονομικής φιλοσοφίας, και μια κοινωνία συνεργαζόμενων χρηστών έναντι εκείνων που επιλέγουν να μην συμμετέχουν. Αντίθετα, βλέπουμε μια δυναμική μετακίνηση της αξίας, στην οποία τα πράγματα εκείνα που κάποτε φυλάσσονταν προς ίδιο όφελος πλέον μοιράζονται

ελεύθερα, και τα πράγματα που κάποτε θεωρούνταν τυχαία αποτελούν πλέον αντικείμενα τεράστιας αξίας. Είναι εύκολο για τους υπέρμαχους του ελεύθερου και ανοιχτού κώδικα να δουν αυτή τη δυναμική ως μια απώλεια της υπόληψης, μια συσσώρευση αξίας η οποία πρέπει να μοιραστεί σε όλους. Αλλά μια ιστορική άποψη μιας λέει ότι η εμπορευματοποίηση των παλαιότερων τεχνολογιών και η αποκρυστάλλωση της αξίας στις νέες τεχνολογίες αποτελεί μέρος μιας διαδικασίας που προωθεί τη βιομηχανία και δημιουργεί μεγαλύτερη αξία για όλους. Αυτό που είναι σημαντικό είναι να βρεθεί μια ισορροπία, στην οποία εμείς ως βιομηχανία δημιουργούμε μεγαλύτερη αξία από αυτήν που επιτυγχάνουμε ως μεμονωμένοι συμμετέχοντες, κάτι που επιτρέπει περαιτέρω ανάπτυξη από τους άλλους.

Δεν μπορώ να πω με σιγουριά σε πιο σημείο θα καταλήξουν τα πράγματα. Αλλά όπως είπε κάποτε ο Alan Kay, «Ο καλύτερος τρόπος να προβλέψει κανείς το μέλλον είναι να το εφεύρει.»^[10] Το πιο θα είναι το επόμενο στάδιο εξαρτάται από όλους εμάς.

7 Συμπέρασμα

Ο Ορισμός του Ανοιχτού Κώδικα και έργα όπως το *The Cathedral & the Bazaar* προσπάθησαν να κωδικοποιήσουν τις βασικές αρχές του ανοιχτού κώδικα.

Αλλά όπως παρατηρεί ο Kuhn, μιλώντας για τους επιστημονικούς πρωτοπόρους που άνοιξαν νέους ορίζοντες μελέτης:

Το επίτευγμά τους ήταν χωρίς προηγούμενο σε τέτοιο βαθμό που να απομακρυνθεί μια σταθερή ομάδα οπαδών από ανταγωνιστικούς τρόπους επιστημονικής δραστηριότητας. Παράλληλα, ήταν επαρκώς ανοιχτό ώστε να καταστεί δυνατή η επίλυση των παντός είδους προβλημάτων από την επαναπροσδιορισμένη ομάδα επαγγελματιών. Τα επιτεύγματα τα οποία μοιράζονται αυτά τα δύο χαρακτηριστικά, θα τα αναφέρω ως «πρότυπα».^[11]

Εν ολίγοις, εάν μια καινοτομία είναι αρκετά ισχυρή ώστε να χαρακτηρίζεται ως νέο πρότυπο, η ιστορία του ανοιχτού κώδικα δεν έχει ακόμα φτάσει στο τέλος της, και τα διδάγματά της δεν έχουν γίνει πλήρως κατανοητά. Από το να σκεφτόμαστε τον ανοιχτό κώδικα μόνο ως ένα σύνολο αδειών χρήσης λογισμικού και σχετικών πρακτικών ανάπτυξης λογισμικού, θα ήταν προτιμότερο να το αντιμετωπίσουμε ως ένα πεδίο επιστημονικής και οικονομικής έρευνας, με πολλά ιστορικά προηγούμενα, και ως μέρος μιας ευρύτερης κοινωνικής και οικονομικής ιστορίας. Πρέπει να κατανοήσουμε την επίδραση των παραγόντων αυτών ως πρότυπα και την επίδρασή τους στην εμπορευματοποίηση, την αρχιτεκτονική συστήματος και τις συνέπειες των δικτύων, καθώς και τις πρακτικές ανάπτυξης που σχετίζονται με το λογισμικό ως υπηρεσία. Πρέπει να μελετήσουμε τους παράγοντες αυτούς όταν εμφανίζονται στα αποκλειστικά λογισμικά καθώς και όταν εμφανίζονται στα παραδοσιακά έργα ανοιχτού κώδικα. Πρέπει να κατανοήσουμε τους τρόπους με τους οποίους τα μέσα με τα οποία αναπτύσσεται το λογισμικό αλλάζουν τον τρόπο με τον οποίο δημιουργείται και χρησιμοποιείται. Πρέπει ακόμα να δούμε πώς οι ίδιες αρχές που οδήγησαν στην πρώιμη κοινή χρήση του ανοιχτού κώδικα μπορούν να επιδράσουν σε άλλα πεδία συνεργατικής δραστηριότητας. Μόνο όταν θα σταματήσουμε να υπολογίζουμε τον ανοιχτό κώδικα με τις δραστηριότητες που εξαιρούνται από τον ορισμό, και όταν ξεκινήσουμε να μελετούμε τους συνταξιδιώτες στο δρόμο για το μέλλον, θα μπορέσουμε να κατανοήσουμε την πραγματική του επίδραση και τότε πλέον θα είμαστε απολύτως έτοιμοι να ενστερνιστούμε πλήρως το νέο πρότυπο.

1. Thomas Kuhn, [The Structure of Scientific Revolutions](#), σελ. 7.
2. Ray Kurzweil, [Speech at the Foresight Senior Associates Gathering](#), Απρίλιος 2002.
3. Clayton Christensen, [Harvard Business Review, Feb 2004](#) (PDF).
4. Με το θέμα της αλλαγής προτύπου ασχολούμαι τόσο στις ομιλίες μου όσο και στα γραπτά μου εδώ και πολλά χρόνια, αλλά μέχρι που άκουσα τον Christensen να μιλάει στο [Επιχειρηματικό Συνέδριο Ανοικτού Κώδικα](#) τον Μάρτιο του 2004, δεν είχα ακούσει την εύγλωττη γενίκευση των οικονομικών αρχών στην εργασία σε αυτό που εγώ καλούσα επιχειρηματικές αλλαγές προτύπου. Νιώθω υποχρεωμένος στον Christensen και Dave Stutz, τα πρόσφατα γραπτά των οποίων πάνω στο θέμα της εμπορευματοποίησης λογισμικού εμπλούτισαν τις δικές μου απόψεις για το συγκεκριμένο θέμα.
5. Clayton Christensen, [Harvard Business Review, Feb 2004](#) (PDF).
6. Από ιδιωτικές επικοινωνίες με τον Juergen Geck της SuSe CTO και τον Γρηγόρη Παπαδόπουλο της Sun CTO.
7. Θα ήθελα να πω ότι το λογισμικό καθιστά δυνατή την επικοινωνία μεταξύ των ανθρώπων και των ηλεκτρονικών υπολογιστών. Αποτελεί επίσης τον καλύτερο τρόπο συζήτησης ορισμένων πλευρών της επιστήμης των ηλεκτρονικών υπολογιστών, όπως ακριβώς οι εξισώσεις είναι ο καλύτερος τρόπος αντιμετώπισης των προβλημάτων στη φυσική. Εάν ακολουθήσετε τη λογική αυτή, συνειδητοποιείτε ότι πολλά από τα επιχειρήματα για τον ελεύθερο διάλογο εφαρμόζονται επίσης στον ανοιχτό κώδικα. Πώς αλλιώς μπορείτε να πείτε σε κάποιον πώς να επικοινωνήσει με τον ηλεκτρονικό του υπολογιστή εκτός από το να του δώσετε τον κώδικα που χρησιμοποιήσατε; Τα οφέλη του ανοιχτού κώδικα είναι ανάλογα με τα οφέλη που προέρχονται από την ελεύθερη ροή ιδεών μέσα από άλλες μορφές διασποράς ιδεών.
8. Ο Dave Stutz παρατηρεί (σε μια απάντηση προσωπικού email σε ένα πρώτο προσχέδιο αυτού του έργου), ότι αυτό το λογισμικό «περιλαμβάνει όχι μόνο αυτό που αποκαλώ «συλλογικό λογισμικό» που λαμβάνει υπόψη τόσο τις ομάδες όσο και τα άτομα, αλλά και ένα λογισμικό που είναι προσαρμοσμένο στην τοποθεσία του στο δίκτυο, καθώς και ένα λογισμικό που είναι προσαρμοσμένο σε μια συσκευή ή ένα εικονικό περιβάλλον εξυπηρέτησης περιφερειακών συστημάτων. Αυτοί οι επιπλέον τύποι προσαρμογής οδηγούν μακριά από το λογισμικό που εκτελείται σε έναν μόνο Η/Υ ή PDA/έξυπνο τηλέφωνο και προς το προσωποποιημένο λογισμικό που εκτελείται «στο δίκτυο» και παρέχεται μέσω πολλών συσκευών ταυτόχρονα.»
9. Από μια απάντηση σε προσωπικό email από τον Eric Raymond σε ένα προηγούμενο προσχέδιο αυτής της μελέτης.
10. Ο Alan Kay, μίλησε το 1971 σε μια εσωτερική σύσκεψη περί του σχεδιασμού Xerox, όπως παρατίθεται στη διεύθυνση www.lisarein.com/alankay/tour.html.
11. Thomas Kuhn, [The Structure of Scientific Revolutions](#), σελ. 10.