

Die Geburt von MacroMind

Marc Canter

Am Beginn von Macromind stand unsere Vision einer Welt, in der PCs und andere Geräte der Unterhaltungselektronik mit höchst interaktiven Grafik- und Musikinterfaces ausgestattet waren. Das war unsere Version jener Szene aus *Blade Runner*, in der der Held ins Foto eintaucht – und mit 1.000 Sachen durch die Pixel jagt. Wir wussten, die digitale Zukunft braucht Werkzeuge; und wir waren Werkzeugmacher mit Erfahrung in der Herstellung von Entwicklungswerkzeugen für Videospiele.

Allerdings hatten wir beim Start von Macromind keine Ahnung von GUIs oder Mäusen. Wir hatten keinerlei Kenntnisse von der PC-Industrie oder vom Softwareverkauf. Wir wussten nur, dass der Macintosh unglaubliche 128 kB RAM und eingebaute Audio- und Videokarten hatte – und mehr brauchten wir nicht zu wissen.

Wir kamen vom Videospielehersteller Bally-Midway in Chicago. Unsere Firma hieß zuerst Chicago Software, wir taufte sie aber drei Tage später nach einer Figur in einem Videospiel meines Kompagnons Jay Fenton – GORF. Wir arbeiteten gerade an der Fortsetzung *Ms. GORF*, als Jay, ich und Mark Pierce als drittes Gründungsmitglied Macromind im April 1984 gründeten. Eines war uns klar – die Welt brauchte benutzerfreundliche „Tools“, die es Künstlern, Musikern und Grafikern ermöglichten, jenes „Zeugs“ – diese Mischung aus Grafik, Text, Musik und Interaktivität – zu kreieren, das uns vorschwebte. Digitales Video war noch in weiter Ferne, aber wir wussten um das Potenzial, das in Computermusik und -grafik steckte – hatten wir doch in Chicago in den späten 1970er Jahren ständig damit zu tun.

1983 konnte man allerdings einen Computer nur mit C, FORTH oder einem ähnlichen Programmierwerkzeug dazu bringen, musikalische oder visuelle Leistungen zu vollbringen.

Deshalb machten wir uns daran, Entwicklungstools für Videospiele zu schaffen, die auch Künstler oder Musiker einsetzen konnten. Wir bezeichneten uns selbst als „Software-Rock-&-Roll-Band“ und wurden von der Agentur Wm. Morris vertreten. Wir sahen uns als moderne Poeten, Rockstars oder Maler, die wie ehemals die großen Büchsen- oder Uhrmacher ihre Werkzeuge selbst herstellten.

Wir hatten diese Idee eines Endanwender-Werkzeugs, eines modernen Violin-Pinsels, für unser erstes Produkt *SoundVision*. Doch unser Verleger bestand darauf, das Konzept zu teilen und zwei Produkte zu kreieren: *MusicWorks* für die Computermusik (wurde später zu MIDI und Sampling-Software) und *VideoWorks* für Grafik und Animation (entwickelte sich zu Multimedia-Animation und Autorensystemen).

Im Oktober 1984 erschien *MusicWorks*, das über die erste Echtzeit-Timeline für Musiknotation verfügte. Es gab auch eine Klaviatur-Timeline und ein Überblicksfenster. Während der Musikwiedergabe konnte der Benutzer Viertel- oder Achtelnoten direkt oder über die Klaviertastatur einfügen. 1984 war das eine Sensation und erweckte die Aufmerksamkeit von Alan Kay, Steve Jobs und der Macintosh-Gemeinde. Und das lange vor MIDI.

Doch innerhalb von zwölf Monaten gab es bereits zehn weitere Musikprodukte für den Macintosh, von dem noch nicht einmal 100.000 verkauft waren. Da ungeheuer viele Programmierer auch Musiker waren, wirkte das Macintosh-GUI wie ein Magnet auf jeden, der schon IMMER von einem interaktiven Echtzeit-Musikinterface geträumt hatte.

VideoWorks erschien im Juni 1985 – und wurde zu unserem Rettungsanker. Bereits Weihnachten 1985 hatten wir eine simple Scriptsprache in die Timeline von *VideoWorks* integriert

und an Apple lizenziert, die damit animierte Lernprogramme auf Disketten produzierten, die mit jedem Macintosh ausgeliefert wurden.

Die Timeline von *VideoWorks* synchronisierte mehrere Kanäle verschiebbarer Sprites mit einem Audiokanal. Jeder Sprite konnte zu jedem Frame verschoben werden und plötzlich fanden sich die Grafiker – mit Tausenden vorbeiziehenden Frames – in einer dynamischen Welt, in der sie Grafik- und Navigationselemente beliebig zu einer Animation kombinieren konnten. In diesen Synchronisierungsmechanismus wurde ein Scriptsystem eingebunden. Damit konnten kreative Köpfe Hot-Spots gezielt setzen und festlegen, was passiert, wenn man sie anklickt, und eine völlige Neudefinition von Menüs vornehmen.

Jedem Multimedia-Menü wurde eine eigene Funktion zugewiesen und schon war das „Multimedia-Authoring“ geboren. Zur selben Zeit gab es auch schon Hypercard, mit dem aber die dynamische Natur von Multimedia nicht machbar war (dafür lag seine Stärke in der Datenbank!), weil es auf einem nicht-zeitbasierten Kartensystem beruhte.

Dank der Neudefinition, was alles als Menü fungieren konnte, und der Schaffung völlig neuer Bildungs-, Unterhaltungs- und Visualisierungswelten hatten Multimedia-Designer ein extrem leistungsfähiges Werkzeug an der Hand, das sie in jede Sphäre vordringen ließ.

Eher zufällig ermöglichten wir es normalen Nicht-Technikern, Multimedia-Inhalte zu kreieren. Wir entdeckten das Multimedia-Authoring, weil der Markt und die Welt dies damals (Mitte bis Ende der 1980er Jahre) brauchten. Wir schauten uns um, erkannten die Möglichkeiten und setzten uns mit den Leuten an einen Tisch, die diesen Multimedia-Kram machten. Und gaben ihnen dann genau das, was sie wollten und brauchten.

Das ist das Geheimnis hinter *Director*. Wir waren Multimedia-Entwickler. Mit Aufträgen von Microsoft, Ashton-Tate und Borland lief unsere Firma prächtig. Wir sind quasi die Erfinder der Marketing- und Schulungsdiskette (bzw. später CD-ROM). 1985 absolvierten wir unsere ersten Projekte – und mussten sie 1988 einstellen, als die Videokonsolen auf der Bildfläche auftauchten und uns die Konkurrenz mit unseren eigenen Kunden untersagt wurde.

Weil wir einen direkten Draht zu den Animatoren, Grafikern und Musikern hatten, die unsere Werkzeuge einsetzten, wussten wir ganz genau, welche Funktionen und Fähigkeiten sie benötigten. Außerdem ließen wir auf den Fachmessen unsere Programmierer die Software den Interessenten vorführen. So lernten sie die Sicht- und Herangehensweise der User verstehen. Das sind einige der Gründe, warum sich *Director* letztlich so entwickelt hat.

Der glücklichste Umstand war jedoch die Tatsache, dass wir uns in der ersten goldenen Ära der Softwareentwicklung befanden, die mit der Veröffentlichung von Lotus 1-2-3, Pagemaker, Photoshop, Painter, Filemaker, Quicken, Sidekick, MORE, Quark Express, Word und Word Perfect begann. Mit der Produktivitätssoftware verlagerte sich die Macht der Mikrocomputer hin zu den Nicht-Programmierern. 1986 war das eine große Sache!

2003 blicken wir zurück und lachen. Wir sind aber noch nicht fertig.

Werkzeuge und ihr neues Paradigma

Allerdings sind wir seit diesen Zeiten der Textverarbeitungen und Tabellenkalkulationen kaum vom Fleck gekommen. Die einzig wirklich neuen Softwarekategorien seit 1988 sind E-Mail, IM (Instant Messages), Browser und Blogging-Werkzeuge. Trotz allen Tamtams und der vermeintlichen Entwicklung hat sich die ursprüngliche Idee, was ein Werkzeug ausmacht, kaum erneuert.

Und wir haben einige Möglichkeiten digitaler Medien aus den Augen verloren.

Doug Engelbart spricht von einer intensivierten Computererfahrung, die Interaktion und Kooperation der Menschen steigert. Don Norman sieht einen unsichtbaren Computer und setzt auf aktivitätsbasierte Usererfahrung. Nicholas Negroponte erörtert Bits und die digitale Vereinigung. Weltweit werden derzeit nur drei Prozent der Glasfaserinfrastruktur genützt.

Ich habe (seit meinem Weggang von Macromedia vor elf Jahren) angewandte Forschung im Bereich skalierbarer Inhalte betrieben und verschiedene Implementierungen einer Multimedia-Personalisierung entwickelt. Wie immer man es auch nennen mag, wir stehen an der Schwelle eines neuen Werkzeug-Paradigmas, das uns genau so weit nach vorne katapultieren wird wie damals in den 1980ern, als der Wechsel vom Programmieren in Assembler zu den Produktivitätswerkzeugen erfolgte.

Software-Tools werden auch heute noch vornehmlich fertig verpackt angeboten. Per Lkw kommen sie in ein Lager, von wo aus sie über ein zweistufiges Distributionssystem in die Läden kommen, zu denen die Kunden hinfahren und wo sie für das Produkt bezahlen müssen – im Voraus. Sie nehmen es mit nach Hause, lesen das Handbuch und versuchen die Software einzusetzen. Das Produkt liefert traditionelle Ausgabeformate für traditionelle Verteilernetze – der Kunde muss weitere Upgrades, Verbesserungen oder Implementierungen von neuen Techniken *kaufen*.

Das alles erscheint mir ziemlich altmodisch.

1997 hatte Dave Winer das Content-Management-System *Frontier* entwickelt, das aus einer Basis-Website und einer Reihe von Tools bestand. Dave wollte auch verschiedene Möglichkeiten der Kommunikations- und Datenflusskontrolle, weshalb er die Fähigkeiten und Funktionen für Browser und Online-Server weiter verfeinerte und eine einfache Scriptsprache integrierte. Er erfand ein Protokoll zum Senden und Aufspüren von allen möglichen XML-Daten (XML-RPC). Er war in einer Zone.

Während ich neben Dave saß und zusah, wie diese Objektdatenbank die Daten herumschob, filterte und sortierte, war ich von der Direktheit und Leichtigkeit, mit der die (wie ich dachte) komplizierten Datenstrukturen und Prozeduren manipuliert wurden, begeistert. Zu meiner Freude hatte ich ein einfachstes System für Maßlösungen und zur Erstellung von Vorlagen entdeckt, das sich leicht an alle meine Vorhaben anpassen ließ!

Plötzlich war mir klar, dass die riesigen Webapplikationssysteme (wie sie Vignette, Broadvision, Interwoven oder ATG anboten) de facto „erwachsene“ Varianten von Daves System *Frontier* waren. Allerdings blieben diese Riesensysteme auf die Entwicklung von traditionellen Websites und E-Commerce-Lösungen beschränkt (zum Teil wohl auch wegen der Kosten und Komplexität) und hinkten weit hinter den Möglichkeiten her.

Doch Dave (und die Firma Pyra) stolperte über etwas, das ALLE Möglichkeiten ausschöpfte: Bloggen. 1999 hatte Dave ungefähr zur selben Zeit, als Pyra ihren *Blogger* veröffentlichte, seinen moderierten Blog-Service *EditThisPage* eingerichtet. Diese frühen Blogging-Tools waren das erste – wie ich es nenne – Paradigma von Werkzeugen, die alles, was der Endanwender benötigt, direkt vor seinen Augen auf der Webseite bereitstellen.

Im Lauf der Entwicklung der „Blogosphäre“ veröffentlichte Dave *Radio Userland*, das das Blog-Werkzeug mit integriertem Aggregator auf dem Rechner des Anwenders installierte. Die nächste Stufe dieses Paradigmas war geboren und knüpfte dort an, wo Napster (und andere P2P-Clients) aufgehört hatte.

News-Aggregatoren wurden möglich, weil Dave und Netscape ein Standardaustauschformat mit der Bezeichnung RSS entwickelt hatten. Mit einem Mal konnten Blogs von jedem als RSS-Channel abonniert werden – aufgelistet neben den Channels der New York Times oder der BBC. Weitere Standards wie OPML oder das MetaWeblogAPI kamen dazu und förderten das, was man heute als Blogosphäre bezeichnet. Diese Standards führten zusammen mit XML-RPC zu News-Aggregatoren und Hunderttausenden von RSS-Feeds und Blogs.

Diese Blogosphäre hat neue Formen von Gemeinschaften und Interaktionen geschaffen, die alle auf der Prämisse privater Veröffentlichung fußen. Das war die Zeit des „Micro-Content“, wie Anil Dash es bezeichnete, in der intelligente, verteilte Content-Brocken eine Ära maßgeschneiderter und widerstandsfähiger Ökosysteme einläuteten. Es sollten neue „Micro-Content-Browser“ entstehen, die völlig neue Navigations- und Surfmöglichkeiten eröffneten.

Ich hatte oft davon geträumt, über die einfachen Links, die Tim Berner-Lee uns bescherte, hinauszugehen. Ich wollte Zweiweg-, Video-, Audio- oder selbsterklärende Links. Ich stellte (dank der Blogosphäre) fest, dass mein Traum vom semantischen Web Wirklichkeit geworden war.

Die Selbstdarstellung als eine neue Form von Amateurjournalismus, mit Berichten über den Kampf um Bagdad oder den eigenen Hinterhof, bilden das Grundprinzip der Blogosphäre. Mir wurde auch bewusst, wie man Technologie, Geist, Ethik und Prozesse der Blogosphäre auf größere Einflussbereiche anwenden konnte. Die Austausch- und alternativen Distributionssysteme hinter der Blogosphäre werden viel mehr als bloß Amateurjournalismus entstehen lassen. Jonathan Peterson schreibt in seinem Corante.com-Blog unter *The Me in Media* über die neuen Trends im Personal-Publishing und wie eines Tages Einzelpersonen den Großteil der Online-Medien stellen werden. Der Glaube ans „Narrowcasting“¹ bewegt mich seit Jahren, und ein Narrowcasting-Button zielt auch unser neuestes Tool *WebOutliner* – obwohl wir nicht genau wissen, was Narrowcasting eigentlich ist und was es leisten kann.

Wir sind aber zuversichtlich, dass uns die Endanwender ihre Wünsche mitteilen werden. Wie schon in den Anfängen von *VideoWorks* planen wir, uns mit dem Markt zu entwickeln – sobald wir herausgefunden haben, was es ist! Dieser Vorgang des iterativen Designs war das Geheimnis von *Director* und kann nicht im Voraus entwickelt, geplant oder in einem Gant-Diagramm dargestellt werden.

Offene Standards

Als junger Unternehmer empfahl mir Dave Winer ein Buch, das zu meiner Marketing-Bibel wurde: *Marketing Warfare* von Reiss und Trout. Man soll einen neuen Hügel errichten (was einem neuen Produkt entspricht), auf dessen Gipfel Sperren aufstellen und die Konkurrenz mit Maschinengewehren und Handgranaten in Schach halten. So nahmen wir den entstehenden Markt für Multimedia-Authoring-Werkzeuge ein und verteidigten unsere Stellung. Nun, für Macromedia war diese Strategie des Sich-Verschanzens recht erfolgreich. 15 Jahre später läuft auf 97 Prozent aller Browser ihr Flash-Format.

1994 meinte ein gewisser Doc Searls: „*Ein Dialog wäre besser als Krieg*“, womit sich ein neuer Trend entwickelte, der im Buch *The ClueTrain Manifesto* seinen Ausdruck fand. Doc und seine Co-Cluetrainer versuchten zu erläutern, dass man auf verlorenem Posten steht, wenn man nicht auf seine Klientel hört und mit ihr zusammenarbeitet. Sie sind auch nicht mehr bloß Kunden, sondern Partner. Denn ohne sie ist man nichts. Dieser erfrischend neue Ansatz, wie man Software kauft und verkauft, hat eine völlig offene Welt von Standards und Quellcode erschaffen.

In den letzten acht Jahren haben sich Open Source und offene Standards herausgebildet, die LAMP (Linux, Apache, MySQL, Python/Perl/php) und eine NAJ-Umgebung zur allgemeinen Verwendung schufen: Niemand hat die Kontrolle. Alle können es nützen. Jeder kann es weiter entwickeln. Das Internet ist zu einem blühenden Garten der Innovation geworden, aus dem ein neues Paradigma für Werkzeuge entspringt. Es ist eine „Welt der Enden“, in der die auf kooperierende Menschen verteilte Macht die sich verschanzenden Mächte und Infrastrukturen besiegen kann.

Tatsächlich bauen wir eine neue Infrastruktur auf.

Ich werde von vielen gefragt, wie man in der Open-Source-Welt Geld verdienen kann. Und ich verweise immer auf *Jabber*, das sowohl ein .org als auch ein .com ist. Wir wollen unseren *WebOutliner* als Open Source anbieten und die Nutzer des Codes nur zu einem verpflichten: einen Link zu unserer Site einzurichten und uns als Urheber anzugeben.

Wir wollen unsere Software nach dem Prinzip „Bezahlung bei Nutzung“ anbieten. Man wird den *WebOutliner* gratis verwenden können – so lange man seine Dateien auf unseren Servern

speichert. Will man die Dateien jedoch auf eigenen Servern oder lokalen Laufwerken ablegen, werden 40 USD in Rechnung gestellt. Man wird gar nicht in der Lage sein, seine Medien auf Server hochzuladen. Man soll mit dem Tool nach Herzenslust spielen können; sobald man aber das Erschaffene gezielt einsetzen will, ist der Zeitpunkt gekommen, wo wir glauben, Geld verlangen zu dürfen.

Alles Notwendige wurde erfunden, nun ist gemeinsames Funktionieren gefragt

Gut – alle diese Trends fügen sich ineinander, es gibt offene Standards und eine Menge von Open-Source-Codes (und dank Mitch und Andy werden es noch mehr werden).

Als eine der spannendsten Entwicklungen sehe ich wohl die zukünftige weltweite Verbreitung der technischen und sozialen Normen an, die sich in der Blogosphäre herausgebildet haben – und wenn dies nur geschieht, damit die ISPs weiter Bandbreite verkaufen können, die fast unbegrenzt vorhanden ist. Denn sie werden Software einsetzen, um sich in ihren Angeboten zu unterscheiden. Dasselbe gilt übrigens auch für Anbieter von Unterhaltungselektronik und Systembetreiber. Eigentlich können unendlich viele Dinge von diesen neuen Werkzeugen und den Funktionen und Inhalten, die sie ermöglichen, profitieren.

Die besten Beiträge werden (natürlich) von den Digitalkünstlern und -musikern (und anderen) kommen, die einfach damit spielen wollen. Es bieten sich so viele Möglichkeiten! Gemeinschaftliche und offene Galerien, virtuelle Forschungszentren, Wikis, Foren, reine Nachrichten, soziale Netzwerke und Systeme werden entstehen, die neue Formen der Kreativität eröffnen. Stellen Sie sich eine neue Art interaktiver TV-Sendungen vor, die auf einem x-beliebigen PC entstand und per Narrowcasting nur einer begrenzten Anzahl von Zusehern zur Verfügung steht. Und das ist nur *ein* mögliches Szenario!

Auf virtuellen Flohmärkten wird man seine Sachen verkaufen können – und auf eBay herumspazieren. Neue Werkzeuge ermöglichen die Erstellung von neuem Micro-Content, die außerdem eine eingebaute Online-Gemeinschaft Gleichgesinnter bieten.

Ein Weg zur Realisierung könnte über das von Seb Paquet so genannte „strukturierte Blogging“ führen. Dieses Micro-Content-Konzept kann auf viele andere Medien- und Datentypen angewendet werden. So kann man sich ein unveränderliches digitales Identitätsprofil einer Person als Micro-Content vorstellen. Eine Medienbibliothek oder eine Fotosammlung (Alben) sowie Musik oder Videos (Titellisten) sind ebenfalls Arten von Micro-Content. Dies alles sind offensichtliche Beispiele für diese neuen Formen.

Es gibt aber auch revolutionär neue Formen. So kann eine Unterhaltung (die von einer IM- oder Chat-Sitzung, von einem E-Mail-Verkehr, einem Message-Board oder einem Forum stammen kann) einen Micro-Content darstellen, ebenso eine Film-, Platten- oder Restaurantkritik. Sogar eine ganze Online-Gemeinschaft kann als Micro-Content abgelegt werden.

Tim O'Reilly spricht über neue Distributionssysteme. Dazu müssen wir uns vorstellen, was passiert, wenn neue Tools Unmengen neuer Formen von Micro-Content kreieren, die völlig anders geartete Online-Gemeinschaften entstehen lassen.

Genau das ist es, was unsere Firma Broadband Mechanics macht. Wir hoffen nur, dass so viele neue Tools und Technologien wie möglich die offene Arena betreten. Die Absicht dahinter ist, dass wir alle gemeinsam es Microsofts Projekt Longhorn gleichtun können.

Eine untereinander vernetzte Welt von Micro-Content-Servern und RSS-fähigen Werkzeugen kann ein verteiltes, auf Open Source und Webdiensten basierendes „Volksnetz“ erschaffen. Longhorn und iLife von Apple sind die Latte, an der wir unser Volksnetz messen werden. Das erklärte Ziel dabei ist, die gleiche Funktionalität wie diese, aber kostenlos und für alle frei zugänglich, zu bieten.

Aus dem Amerikanischen von Michael Kaufmann

1 Anm. d. Ü.: Antonymische Wortbildung zu *broadcasting* (senden, übertragen).