

Ein Stündchen mit Anita plauschen...

von Boris Schneider & Heinrich Lenhardt

Der Name Anita Sinclair hat nichts mit Sir Clive und seinem Spectrum zu tun - aber schon einiges mit dem Sinclair QL. Für diesen Computer erschien nämlich als erstes das vielgelobte Adventure »The Pawn«. Anita Sinclair gehört zur Programmierer-Truppe, die das Abenteuerspiel schrieb.

Im Trubel der PCW-Show im Spätherbst 1986 trafen wir so manchen bekannten Programmierer. Einer der Glanzpunkte war aber eine Programmiererin: Wir entdeckten Anita Sinclair (23), die die Commodore-64-Version von »[The Pawn](#)« vorstellte. Anita ist Mitglied der »drei-Mann-und-eine-Frau-starken« Programmierertruppe namens [Magnetic Scrolls](#), die sich seit knapp drei Jahren mit der Entwicklung von Adventures beschäftigt.

Gemeinsam ziehen wir drei ins Pressezentrum, in dem uns Anita bei einem heißen Kaffee viele Fragen beantwortet.

Happy: Anita, die erste Frage lautet natürlich, wie kommt eine Frau zum Programmieren? Das ist doch schon sehr ungewöhnlich.

Anita: Da habt ihr völlig unrecht! Programmierinnen gibt es ebenso viele wie Programmierer, zumindest im professionellen Bereich, wo man in Sprachen wie Cobol programmiert. Da ist das Verhältnis in etwa 50 zu 50. Aber ich gebe zu, daß im Spielmarkt die Männer klar die Oberhand haben.

Happy: Der erste Punkt ging an dich. Also, wie kommt eine Frau dazu, Spiele zu programmieren?

Anita: Ich habe mich schon immer für technische Sachen interessiert. Das äußerte sich darin, daß ich niemals Puppen besaß. Ganz im Gegenteil, ich habe sogar mal das Auto von meinem Vater total auseinandergenommen. Und bevor jemand jetzt dumme Witze reißt: Ich hab' es auch wieder richtig zusammengebaut. Zum Glück haben meine Eltern meine technische Neigung rechtzeitig erkannt und mir die entsprechende Ausbildung ermöglicht.



Boris Schneider im Gespräch mit Anita Sinclair

Happy: Wie waren denn deine ersten Schritte mit Computern?

Anita: Nun, ich habe vor einigen Jahren einen Apple II in die Finger bekommen. Durch die Basic-Phase war ich schnell hindurch. Ein Schlüsselerlebnis für mich war, glaube ich, als ich es schaffte, in Assembler

meinen Namen auf dem Bildschirm zu schreiben. Als nächstes schrieb ich dann ein Animations-Programm: Mein Name bewegte sich quer über den Schirm. Damals wußte ich sofort, daß ich noch längere Zeit mit Computern zu tun haben werde.

Happy: *Wie fing nun alles mit [Magnetic Scrolls](#) an?*

Anita: Magnetic Scrolls war eine Idee, die etwa vor dreieinhalb Jahren geboren wurde. Ihr müßt eines wissen: In der Zeit hatte ich einige Infocom-Adventures gespielt. Ich mag [Infocom](#) übrigens wirklich, sie haben einen tollen Stil in ihren Stories. Auf jeden Fall dachte ich mir, mit drei Programmier-Kollegen, daß wir auch so ein Adventure schaffen würden. Also haben wir uns die nötige Hardware besorgt und angefangen zu programmieren.

Die Hardware bestand damals aus einem Netzwerk von vier Apple-Computern, die an eine Festplatte (damals noch ein kaum erschwinglicher Luxus für diese Systeme) angeschlossen waren. Inzwischen arbeiten wir mit einem wesentlich größeren Netzwerk mit verschiedenen Computern. Wir haben unsere eigenen, selbstgeschriebenen Compiler und Assembler. Für unsere Adventures haben wir eine eigene Programmiersprache entwickelt, in der sich [Parser](#) relativ einfach programmieren lassen. Da wir alle Tools zum Programmieren selbst entwickelten, stecken etwa zwölf Jahre Arbeit in [The Pawn](#).

Eine kleine Story am Rande, die zeigt, wie wichtig es für uns war, mit unseren eigenen Programmier-Tools zu arbeiten: Als The Pawn mehr oder minder fertig war, ging es ans Debuggen, an die Fehlersuche also. Dazu gibt es ebenfalls ein Programm, einen Debugger, der ähnlich wie ein Maschinensprache-Monitor arbeitet. Unser Debugger sagte uns also laufend, daß wir an der und der Stelle einen Fehler hätten. Aber wie oft wir auch über diesem Code brüteten, wir konnten den Fehler nicht entdecken. Schließlich kam einer von uns darauf, daß der Debugger selbst einen Bug hat. Da wir den Debugger programmiert hatten, konnten wir diesen Fehler zum Glück bereinigen. Es läßt sich nie ausschließen, daß ein Programm einen Fehler hat, deswegen arbeiten wir mit Programmen, bei denen wir die Fehler selber korrigieren können.

Happy: *So, wie sich das anhört, war es sicherlich nicht billig, The Pawn zu entwickeln. Ihr wußtet ja vor drei Jahren noch nicht, ob das Projekt Erfolg haben würde und hattet auch sicherlich noch keinen Vertrag in der Tasche. Wie habt ihr das finanziell bewältigt?*

Anita: Zuallererst einmal standen wir vier völlig hinter der Idee von »The Pawn« und unserem Parser. Wir glaubten fest daran, daß dieses Programm ähnlichen Erfolg wie ein Infocom-Titel haben müßte, weil wir mindestens die gleiche Qualität erreichen wollten. Und wenn man so fest an eine Idee glaubt, dann schafft man es immer irgendwie, das nötige Geld herbeizuschaffen. Das Ganze wurde so etwa 18 Monate lang von meinem Ersparten finanziert. Aber ein wichtiger Punkt waren wohl unsere Mütter, die uns immer gut versorgt und »durchgefüttert« haben. Ohne sie wären wir wohl nicht so weit gekommen. Inzwischen verdienen wir ja etwas mit unseren Adventures, so daß die finanzielle Seite jetzt etwas sicherer steht.

Happy: *The Pawn erschien ja zuerst für den Sinclair QL als reines Textadventure. Wie kamen die Bilder in die Atari ST-Version hinein?*

Anita: The Pawn ist auch eigentlich immer ein Textadventure gewesen. Wir hatten die Atari-ST-Version fertiggestellt und gingen zu Rainbird, um einen Vertrag auszuhandeln. Da kam irgend jemand auf einmal mit der sehr dummen Idee, daraus ein Grafik-Adventure zu machen. Ich sagte nur zwei Worte, und diese sehr eindringlich: »Forget it!« (Vergiß es). Von diesem Zeitpunkt an wurden wir mit Samthandschuhen angepackt und man überredete uns, doch mal ein Bild von Geoff Quilley anzusehen. Das Bild war überwältigend. Wir dachten, laß es uns mal probieren, und sandten ihm Vorlagen für den Wald und den Palast-Garten. Er stellte die beiden Bilder fertig und die waren so gut, daß wir schlagartig unsere Meinung änderten. Die Bilder sind nicht in The Pawn drin, weil wir Bilder haben wollten, sondern weil die Bilder so gut sind, daß es geradezu dumm wäre, sie nicht zu bringen.

Eines unserer Hauptargumente gegen Bilder war der Speicherplatz-Verbrauch. Die meisten Grafik-Adventures haben sehr simple Bilder, die zur Handlung nichts beitragen und nur wertvollen Speicherplatz wegnehmen. Aber jetzt kommt wieder eine kleine Geschichte: Auf der Sommer-CES in Chicago stellte ich mal wieder The Pawn vor. Natürlich kamen auch ein paar Leute von Infocom an unseren Stand und wir hatten ein interessantes Gespräch, das sich auch um das Thema Grafik drehte. Am Ende konnte ich mich dann nicht zurückhalten und zog meinen Trumpf aus der Tasche: Eine Diskette mit Grafikbildern aus unserem zweiten Adventure, [»The Guild of Thieves«](#). Ich lud ein Bild, das ein Zimmer mit seiner Einrichtung zeigte. Das Bild ist eines der besten und detailreichsten. Ich fragte die Infocom-Leute: Wie viele Wörter braucht ihr, um dieses Bild zu beschreiben? Die Antwort war: Um die 1500 Wörter. Dann

rechnete ich ihnen vor, daß mit Hilfe unseres Grafikpackers das Bild wesentlich weniger Speicherplatz schluckt als 1500 Wörter Text in einem Infocom-Adventure.

Happy: *Kannst du uns noch etwas mehr über »The Guild of Thieves« erzählen?*

Anita: Über die Handlung werde ich nichts verraten. Da müßt ihr bis zum Frühjahr warten, wenn das Adventure erscheint. Aber ich erzähle euch gerne etwas über die Bilder. Wir hatten mit The Pawn gesehen, daß Bilder in Adventures so gut sein können, daß sich der Einsatz lohnt. Also stand fest, daß The Guild ebenfalls Bilder haben wird. Und diesmal sind die Bilder nochmals um Klassen besser als bei The Pawn! Die Bilder sind jetzt nicht nur schmückendes Beiwerk, sie erfüllen auch bestimmte Aufgaben im Spielverlauf. Wir haben deswegen die Texte für das Adventure noch mal umgeschrieben, nachdem wir die Bilder gesehen hatten.

Happy: *Wie macht denn euer Maler seine Bilder? Hat er sein eigenes Malprogramm, arbeitet er mit einem Digitalisierer?*

Anita: Digitalisierer lehnen wir alle konsequent ab. Die Qualität ist einfach zu mies! Ein Grafiker kann viel detailreicher arbeiten. Durch das Mischen von kontrastreichen Farben kann man den Effekt eines Bildes wesentlich erhöhen, und gerade das kann ein Digitizer nicht. Wir haben es einmal probiert und werden ab sofort die Finger davon lassen.

Damit Geoff mit seinen Bildern gut vorankommt, braucht er ein gutes Zeichenprogramm, das dachte ich zumindest, Also kaufte ich einige ein: »Degas«, [»Deluxe Paint«](#) und wie sie alle heißen. Über 1000 Mark gab ich für die Dinger aus. Dann ging ich zu Geoff und der meinte: Was soll ich damit? Ich brauche den ganzen Krempel nicht. Alle Bilder entstehen nämlich mit... »Neochrome«! Ich dachte zuerst, Geoff wäre verrückt und meinte: Schau mal, wie viele Funktionen Deluxe Paint hat! Dann nahm er die Maus und Neochrome und malte mir innerhalb von Sekunden alles, was ich wollte: Kreise, Rechtecke, kleine Bilder, sanfte Farbabstufungen. Geoff versteht wirklich etwas von der Malerei und braucht diese ganzen Funktionen nicht - er hat sozusagen alle Funktionen von Deluxe Paint in seiner rechten Hand.



Anita führt auf der PCW mit Begeisterung »The Pawn« vor

Happy: *Wie viele Leute sind jetzt bei Magnetic Scrolls beschäftigt?*

Anita: Wir haben inzwischen acht Programmierer, inklusive mir selbst. Die neuen Programmierer kümmern sich hauptsächlich um Umsetzungen der Spiele für andere Computer. Wir haben also einen C 64-Experten, einen Schneider-CPC-Experten und einen Atari-8-Bit-Experten. Für jede Version muß

unserer Parser-Sprache neu implementiert werden. Die Grafiktreiber sind neu zu schreiben, die Bilder müssen neu gezeichnet werden. Bis auf die Amiga-Version, die die ST-Bilder verwendet, haben alle Versionen ihre eigenen Bilder, damit diese den Computer voll ausnutzen können. Bei der C 64-Version, die als erste 8-Bit-Version fertig wurde, haben wir eine Menge Programmier-Tricks benutzen müssen. So wird der Prozessor im 1541-Laufwerk als Co-Prozessor genutzt, um Texte zu entschlüsseln. Auch das Scrolling der Bilder ist mit vielen Software-Tricks verbunden. Der Aufwand hat sich aber gelohnt, denn unseren Messungen nach ist unser C 64-Parser wesentlich schneller als der von Infocom.

Happy: *Noch eine Frage, die eure augenblickliche Arbeit betrifft. Es hieß mal, daß ihr eure Adventures in beliebige Sprachen übersetzen könntet, da euer Parser-System sehr flexibel sei. Was ist da jetzt genau dran? Kann man vielleicht schon bald deutsche Adventures von euch erwarten?*

Anita: Wir waren vor einiger Zeit noch sehr optimistisch, was das Thema Übersetzungen anging. Aber als wir uns näher mit der deutschen Grammatik befaßt haben, fanden wir heraus, daß es mit unserem jetzigen Parser-System unmöglich ist, eine deutsche Version eines Adventures zu produzieren. Noch mehr Probleme gäbe es übrigens bei französischen Versionen. Aber wir arbeiten an dem Problem, indem wir einen neuen Parser entwickeln, der wirklich an viele verschiedene Sprachen angepaßt werden kann. Ein deutsches Adventure von uns sollte es in vielleicht zwei Jahren geben.

Happy: *Du bist ja ziemlich aufgeschlossen dem ganzen Computer-Geschäft gegenüber. Aber sicherlich gibt es Dinge, die du in der Branche überhaupt nicht magst. Was gefällt dir in der Software-Industrie nicht?*

Anita: Was ich überhaupt nicht mag, ist Hying (Erklärung siehe Leserbriefseite) und Film-Spiele, die meist nichts mit dem Film zu tun haben.

Aber viel mehr Kopfzerbrechen bereiten uns Programmierern die neuen Computer wie der Atari ST und der Amiga. Die Entwicklungskosten für ein neues Programm auf einer 16-Bit-Maschine sind sehr hoch. Spiele wie [Marble Madness](#) oder [Starglider](#) brauchen nämlich viel Zeit, bis sie fertig sind. Viele Firmen können oder wollen sich aber diese Entwicklungskosten nicht leisten. Also setzen sie Spiele von 8-Bit-Computern auf 16-Bit-Computer um. Der Softwarekäufer bekommt alte Ware vorgesetzt, die den Computer in keiner Weise ausnutzt. Die Software verkauft sich nicht so gut, die Firmen erhalten nicht genug Geld und produzieren weiterhin Low-Quality-Software. Diesen Teufelskreis muß man durchbrechen. Ich kenne eigentlich nur drei Programme, die es wert sind, auf dem Atari ST zu spielen: »Starglider«, »Time Bandits« und »The Pawn«. Der Rest der Programme, die ich bisher gesehen habe, ist für einen 16-Bit-Computer viel zu simpel und auf die schnelle programmiert.

Happy: *Gehen wir noch ein bißchen auf eure Zukunft ein: Was wird Magnetic Scrolls in den nächsten Jahren machen, was wird Anita Sinclair machen?*

Anita: Wir haben jetzt einen Vertrag mit [Rainbird](#), der uns verpflichtet, sechs Adventures innerhalb der nächsten zwei Jahre zu programmieren. Im übrigen bezeichnet sich Magnetic Scrolls als Parser-Spezialist. Wir sind nicht nur auf Adventures fixiert. Um konkret zu werden: Wir haben Programme in Arbeit, die zwar einen Parser verwenden, aber keine Adventures sind. Mehr darf ich nicht sagen, aber diese Projekte sollten eigentlich hohe Wellen auf dem Unterhaltungsmarkt schlagen.

Zu Anita Sinclair: Ich glaube zwar nicht, daß ich in zwanzig Jahren immer noch an Parsern programmieren werde. Aber für die nähere Zukunft kann ich mir nichts anderes vorstellen.

Happy: *Vielen Dank für deine ausführlichen Antworten.*

Zum Schluß gab uns Anita noch bekannt, welche Spiele sie auf eine einsame Insel mitnehmen würde, wenn sie fünf zur Auswahl hätte. Anita bestand aber unbedingt darauf, außer Konkurrenz noch eine Textverarbeitung für den Macintosh mitzunehmen, die wir ihr hiermit genehmigen. Die Programme sind in einzelnen (in Klammern die dazugehörige Hardware):

- [Hack](#) (Universitäts-Mainframe)
- [Wizardry](#) (MS-DOS)
- [Zork-Trilogie](#) (Typ egal)
- [Castle Wolfenstein](#) (Apple II)
- [Elite](#) (BBC)

Eine bunte Mischung also, bei der Anita einiges an Hardware zu schleppen hat.