

2 Allgemeine Squeak Einführung

Markus Gälli

Universität Bern, Institut für angewandte Mathematik

Squeak⁹ ist eine auf Smalltalk basierende, freie und multimediale Spiel-, Lern-, und Entwicklungsumgebung. Dabei gibt es zwei verschiedene Entwicklergemeinschaften und Zielgruppen. Während auf squeak.org hauptsächlich eine von Smalltalk-Entwicklern für Smalltalk-Entwickler geschriebene mächtige objektorientierte Entwicklungsumgebung entsteht, verfolgen die Macher von squeakland.org das ursprüngliche Ziel von Squeak: Kindern sowohl einen neuen, spielerischen Zugang zum Programmieren zu ermöglichen, als auch in einer konstruktivistischen Sichtweise die Entwicklung von Simulationen naturwissenschaftlicher Phänomene durch die Schüler selbst als Lernmethode zu unterstützen. (Allen-Conn 2003). Dafür wurde für die Squeakland-Version eine kachelbasierte visuelle Programmierungsumgebung namens EToys entwickelt, die auch komplett auf Deutsch übersetzt wurde (www.squeak.de).

Plugins für verschiedene Webbrowser ermöglichen den Austausch von EToy-Projekten. Wichtige Eigenschaften wie Betriebssystemunabhängigkeit, Unterstützung der verschiedensten Netzwerkprotokolle, strikte Objektorientierung, umfassende multimediale Unterstützung sowie der freie Zugriff auf System und Quellcode machen Squeak aber gleichfalls als Forschungswerkzeug für die Untersuchung von Fragestellungen im Bereich der Mensch-Computer-Interaktion interessant. Besonders hilfreich ist dabei, dass die Grenzen zwischen den beiden Hauptstoßrichtungen von Squeak fließend sind: In der EToy-Version von Squeakland kann auch Smalltalk programmiert werden, das 10 Megabyte große System enthält sogar eine recht mächtige Smalltalk-Entwicklungsumgebung inklusive Klassenbrowsern.

Auf der anderen Seite findet man in der Entwicklerversion von squeak.org nicht nur eine vollständige objektorientierte Entwicklungsumgebung a la Eclipse – mit Syntaxhighlighting, Code-Completion, Refactoring-Werkzeugen und Testframeworks – sondern auch die visuelle EToy-Programmierungsumgebung.

Die Autoren des Workshops benutzen beide Ansätze, Smalltalk und EToy-Programmierung, um Studien im Bereich der Mensch-Computer-Interaktion durchzuführen: Während mittels EToys leicht und schnell Prototypen und kleinere Usability-Studien gebaut werden können, macht sich besonders bei komplexeren Simulationen von Geräten, Mikrowelten und Experimentalsteuerungen die Mächtigkeit, Wiederverwendbarkeit und dynamische Typisierung von Smalltalk bezahlt.

2.1 Motivation von Squeak-EToys

Alan Kay verfolgt seit mehr als 30 Jahren den Traum eine Maschine zu bauen, die den Möglichkeiten des persönlichen Computers, „eines Instrumentes, dessen Musik Ideen sind“, gerecht wird.

Diese Maschine nennt er „Dynabook“. Maßgeblich geprägt von den konstruktivistischen Ideen Seymour Paperts (Papert 1980), möchte Kay Kindern jeden Alters ermöglichen, mit einem tragbaren, interaktiven und vernetzbaren „Buch“ aktiv neue Inhalte zu schaffen und austauschen zu können – seien es wissenschaftliche Simulationen, multimediale Kunstwerke, interaktive Geschichten oder Spiele.

Kay ist überzeugt, dass die Nutzung des Mediums Computer auch heute noch in den Anfängen steckt. Squeak ist ein Ansatz, dies zu ändern. „Die meisten Menschen“, kritisiert Kay, „simulieren mit dem Computer zurzeit nur andere Medien“ – der Computer ist Briefträger, CD-Spieler, Buchhalter oder Zeitung. Einen Computer zu programmieren wird hingegen als trocken empfunden, ähnlich dem Lösen abstrakter mathematischer Gleichungen. Entsprechend wenige nehmen bislang diese Hürde. Doch Programmieren muss keine Angelegenheit für verschrobene Computerfreaks sein, meint Kay. Ganz im Gegenteil – Computer bieten die Möglichkeit, komplexe Sachverhalte anschaulich zu simulieren: Wenn ein Kind am Computer ein Auto nicht nur malen, sondern dieses auch steuern und festlegen kann, wie viele Pixel es sich pro Sekunde vorwärts bewegen soll, lernt es etwas über Geschwindigkeit (Abbildung 2).

⁹ Alan Kay stellte das maßgeblich durch ihn beeinflusste System im Rahmen eines eingeladenen Vortrages auf der Mensch & Computer 2002 vor (Lessmöllmann 2002).

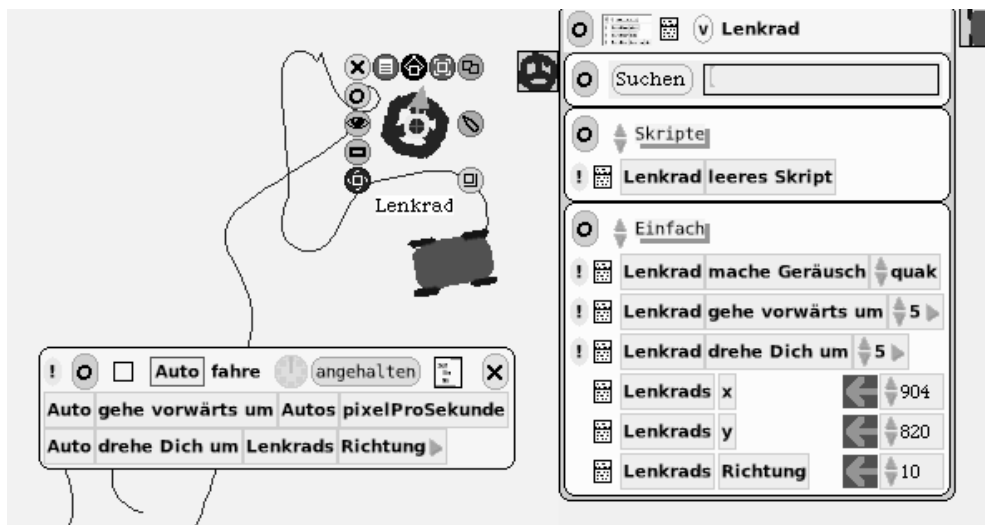


Abbildung 2: In der eingedeutschten EToy-Umgebung können einfach neue visuelle Objekte erzeugt und deren Eigenschaften verknüpft werden: Das Auto bewegt sich um pixelProSekunde vorwärts, dreht sich dabei um die Auslenkung des Lenkrads und hinterlässt eine blaue Stiftspur.

Genau über solche Mechanismen bringt Squeak Kindern nicht nur den Umgang mit Computern und ihre selbstständige Programmierung näher, sondern ist auch selbst als Lernwerkzeug einsetzbar.

2.2 Vom Dynabook zum \$100 Laptop

„Die Zukunft sagt man am besten voraus, indem man sie erfindet“ (Alan Kay). Er sollte es wissen: So erfand er Ende der 60er Jahre das Prinzip überlappender Fenster, entwarf erste Prototypen tragbarer Computer und prägte neben dem Begriff „Objektorientierung“ auch den vom „Personalcomputer“.

Nachdem er seine Dissertation über „The Reactive Machine“ (Kay 1969) beendet hatte, in der er unter anderem das Konzept überlappender Fenster entwickelte, leitete er die „Learning Research Group“ am Forschungszentrum von Xerox (PARC). Das Ergebnis war Smalltalk: Ein integriertes System aus objektorientierter Programmiersprache, Entwicklungsumgebung mit reichhaltigen Bibliotheken und grafischer Oberfläche. Es war die erste moderne grafische Benutzeroberfläche mit überlappenden Fenstern, wie man sie heute auf jedem PC findet. Steve Jobs hat später die Oberfläche für den Apple-Computer „Lisa“ komplett überarbeiten lassen, nachdem er bei Xerox eine Demonstration von Smalltalk gesehen hatte. Zu den berühmtesten Legenden der Computerbranche gehört, dass Jobs den Chefentwickler von Smalltalk, Dan Ingalls, fragte, ob man anstatt zeilenweise auch pixelweise scrollen könnte. Ingalls öffnete während der Vorführung einen Editor und änderte den entsprechenden Code im laufenden System.

Für ihre Vision, Konzeption und Entwicklung des ersten vernetzten und brauchbaren Personalcomputers wurde den ehemaligen Kollegen vom Xerox PARC (Alan Kay, Butler Lampson, Robert Taylor und Charles Thacker) von der National Academy of Engineering Ende Februar 2004 der wichtigste Ingenieurspreis der USA, der Charles-Stark-Draper-Preis, verliehen. Hinzu kamen 2005 der Kyoto-Preis (zusammen mit Habermas) und der Turing-Award, der wichtigsten Auszeichnung für Informatiker.

Anfang der 80er Jahre entwickelte sich Smalltalk zu einem immer professionelleren Programmiersystem: Das 1983 vorgestellte und später kommerzialisierte Smalltalk 80 war eines der einflussreichsten Software-Projekte der vergangenen 30 Jahre. So wird beispielsweise heute mit Hilfe eines Smalltalk-Systems der Fahrplan der Deutschen Bahn berechnet und die Chipfertigung der AMD in Dresden gesteuert.

Aber die kommerziellen Smalltalk-Dialekte wurden dem ursprünglichen Ziel eines Computers für Kinder nicht gerecht. Deshalb nahmen Mitte der 90er Jahre Kay und seine Forschungsgruppe einen neuen Anlauf, die Ziele des Dynabooks doch noch zu erreichen. Die Squeak-Central genannte Gruppe startete bei Apple das Squeak-Projekt und veröffentlichte 1996 die erste Version unter einer freien und liberalen Lizenz im Internet. Später entwickelte Squeak-Central bei Walt Disney Imagineering die mit Squeak realisierten EToys. Mit diesen basteln heute Schüler von Los Angeles bis Kyoto ihre ersten Simulationen oder Spiele. Kay gründete nach der Zeit bei Disney die Non-Profit-Organisation „Viewpoints Research Institute“ (www.viewpointsresearch.org) und arbeitete in der Forschungsabteilung von Hewlett-Packard wei-

ter an der Zukunft des Lernens. Während dieser Zeit entstand auch das Squeak-basierte Open Croquet-Projekt (www.opencroquet.org), eine freie Plattform, die verteilte Echtzeit-Programmierung in einer dreidimensionalen Mehrbenutzer-Umgebung ermöglicht. Seit 2005 ist Kay einer der Köpfe der „One Laptop per Child“-Organisation (www.laptop.org). Diese vom Gründer des MIT-Medialab Nicholas Negroponte ins Leben gerufene, gemeinnützige Organisation hat es sich zum Ziel gesetzt, möglichst vielen Kinder auf aller Welt Zugang zu Wissen und wissenschaftlicher Ausbildung zu ermöglichen, indem ihnen kostenlos ein tragbarer Rechner zur Verfügung gestellt wird. Die Idee des Dynabooks nimmt konkrete Formen an.



Abbildung 3: Mary Lou Jepsen, Alan Kay and Nicholas Negroponte stellen den \$100 Laptop auf der „World Summit on the Information Society“ in Tunis 2006 vor. Squeak wird auf dem \$100 Laptop installiert sein.

2.3 Squeak als Simulationsumgebung

Squeak steht in der Tradition von Simula, Logo, Self und Hypercard: Während es die konsequent objektorientierte Architektur Smalltalks fortgeschrittenen Entwicklern erleichtert, komplexe Simulationen zu kreieren (Abbildung 5), können auch Durchschnittsanwender mit sensorgestützten Schildkrötenrobotern interaktive Simulationen erzeugen, diese mit Animationen anreichern, in Multimedia-Präsentationen einbetten und im Netz veröffentlichen.

Obwohl der Fokus von Squeak EToys auf der konstruktivistischen Lernerfahrung liegt, ist die Mächtigkeit der aktuellen Standard-Bordmittel überraschend: So wird eine Animation einfach dadurch erzeugt, dass verschiedenen Standbilder in einen Behälter gepackt werden, der immer auf ein ausgewähltes Bild zeigt. Einem Platzhalter-Objekt wird dieses ausgewählte Bild als Form zugeordnet, anschließend wird der Zeiger des Behälters um eins erhöht, so dass er auf das nächste Bild verweist (Abbildung 4). Erwähnenswert sind auch die „Connectors“, mit welchen sich mittels weniger EToy-Zeilen beispielsweise Petri-Netz Simulatoren basteln lassen (Nierstraß & Gälli 2005).

Einen Kreis kann man zeichnen, indem man eine einzelne Schildkröte immer wieder einen Schritt nach vorne machen und sich etwas drehen lässt, aber es gibt eine noch einfachere Variante einen Kreis zu erzeugen: Man nehme tausende von Schildkröten, packe sie auf einen Punkt und lasse sie in eine zufällige Richtung schauen. Anschließend sage man allen Schildkröten, sie sollen sich um ein paar Pixel nach vorne bewegen und einen Punkt zeichnen. Diese Art des Programmierens emergenter Systeme findet auch wieder ihren Vorläufer in einer Logo-Variante namens Star-Logo, wobei

„Star“ symbolisiert, dass mit einer Vielzahl von Schildkröten gearbeitet wird. Das analoge System in Squeak hieß zunächst „Star-Squeak“, bis es von „Kedama“ (Ohshima 2004) ergänzt wurde, einer Lösung, die auch mit EToys programmierbar ist. In Abbildung 4 (links unten) sieht man eine typische Simulation von Kedama, in welcher das Verhalten von Ameisen modelliert wird, die je nachdem ob sie Futter gefunden haben oder nicht, Pheromone ausschütten und so anderen Ameisen den Weg weisen.

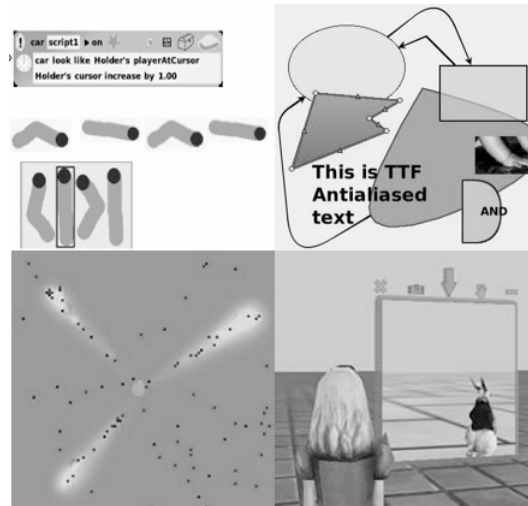


Abbildung 4: Die Bandbreite von Squeak beinhaltet Animationen, Vektorgrafiken und True Type Fonts, Ameisensimulationen oder auch verteilte kollaborative 3D-Umgebungen a la Open Croquet.

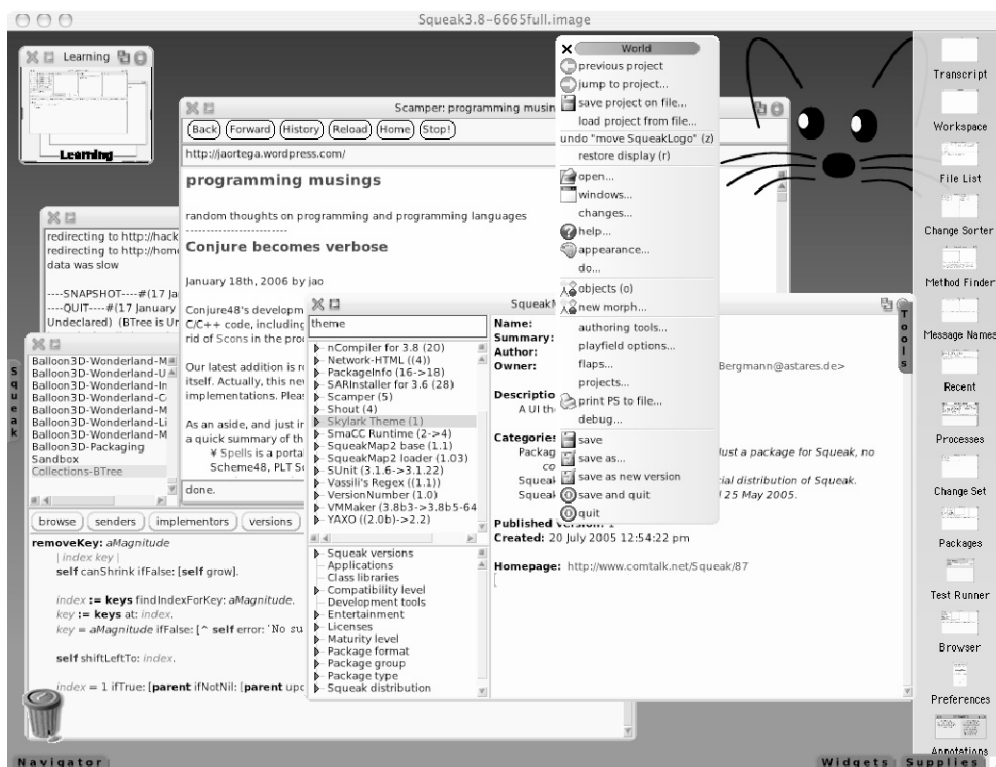


Abbildung 5: Squeak ist ein System, das dynamische Typisierung a la Ruby oder Python mit einer mächtigen Entwicklungsumgebung a la Eclipse vereinigt und somit "rapid prototyping" erlaubt. Alles in Squeak ist ein Objekt.

Kontaktinformationen

Universität Bern
Institut für angewandte Mathematik
Markus Gälli

Neubrückstr. 10
CH-3012 Bern
gaelli@iam.unibe.ch

Tel.: +41 31/631-3313
Fax.: +41 31/631-3262