



Croquet: Verteilte Benutzerwelten mit Smalltalk

Mehr als ein Spiel

Frank Müller

Trotz anerkannter Vorzüge führt die Programmiersprache Smalltalk ein Nischendasein. Das hindert Entwickler nicht, sie zur Grundlage neuer Konzepte zu machen. Das auf Squeak basierende Croquet ist ein offenes System, in dem Benutzer weltweit in einer 3D-Umgebung zusammenarbeiten können.

Neuigkeiten gibt es viele, Zukunftsvisionen nur wenige. Eine davon heißt Croquet. Am 11. Oktober 2004 erfolgte die Freigabe der Version 0.1 unter dem Namen Jasmine, seit kurzem gibt es ein Update auf 0.3. Diese frühe Release ist noch weit von der Fertigstellung entfernt, vermittelt aber dennoch einen guten Eindruck von den Zielen der Entwickler. Und diese resultieren in nichts Geringerem als in einem neuen Paradigma für die Zusammenarbeit von Computern und deren Nutzern in einer vernetzten Welt.

Die Vision der Croquet-Entwickler, die in der Regel von Computern mit einer relativ hohen, dafür aber nahezu ausschließlich vom Anwender genutzten Rechenleistung ausgeht, sprengt den heute bekannten Rahmen der IT. Client/Server-Anwendungen nutzen zusätzlich die Leistung zentraler Server, zum Beispiel für Datenbanken oder zunehmend für Teile der Applikationslogik. Aktuelle, serviceorientierte Anwendungen integrieren zudem externe Dienste in ihre Geschäftslogik, sodass hier der Grad der Vernetzung steigt. Doch im Gegensatz zu Croquet ist dies noch weit von der in den Neunzigerjahren durch Sun formulierten Idee „Das Netz ist der Computer“ entfernt.

Sechs Köpfe – Alan Kay, David A. Smith, David P. Reed, Andreas Raab, Julian Lombardi und Mark McCahill – steuern als Principle Architects das Croquet-Projekt, dessen Ziel die Entwicklung eines neuen Betriebssystems inklusive einer innovativen Benutzerschnittstelle ist. Als konzeptionelle Basis dienen die konsequente Nutzung verteilter Ressourcen im Internet, die Zusammenarbeit von mehreren Benutzern im Netz, die Ausnutzung schneller Hardware und 3D-Beschleuniger sowie dynamischer Sprachen mit später Bindung. Als Implementierungssprache dient Smalltalk in der Form von Squeak.

Smalltalk war seiner Zeit voraus

Smalltalk, 1971 von Alan Kay entwickelt und von Simula und Lisp inspiriert, war damals seiner Zeit weit voraus – dynamisch typisiert, objektorientiert und von Anfang an mit einer grafischen Oberfläche ausgerüstet. Gleichzeitig ist Smalltalk plattformunabhängig. Einzig die virtuelle Maschine erfordert ähnlich wie bei Java eine für die Zielumgebung spezifische Implementierung. Alles Weitere wird direkt in Smalltalk entwickelt und

befindet sich im so genannten Image, einer Datei, die alle zu einem System gehörenden Bibliotheken sowie die Entwicklungsumgebung umfasst. Das Image ist plattformneutral und somit leicht zwischen unterschiedlichen Rechnern austauschbar. Gleichzeitig hat der Entwickler Zugriff auf den gesamten Code und kann diesen bei Bedarf anpassen. Der Austausch von Codefragmenten hingegen erfolgt in speziellen Smalltalk-Dateien, Change Sets oder proprietären Formaten für ein Versionsmanagement.

Die Entwicklung von Smalltalk-Applikationen findet in unterschiedlichen Browsern statt. Diese erlauben ein schnelles Blättern durch die Pakete, Klassen, Methodenpakete und Methoden. Änderungen im Methodenrumpf werden mit dem Übernehmen umgehend in maschinenunabhängigen Bytecode kompiliert und stehen sofort zur Verfügung. Ebenso können Inspektoren den Inhalt von Objekten durchsuchen und verändern. Da alles gemeinsam mit der Applikation in der VM läuft, ergibt sich ein schneller Entwicklungs- und Testzyklus. Im Fehlerfall erlaubt ein Dialog den sofortigen Sprung in den Debugger, in dem der Entwickler im Call Stack den Programmablauf nachvollziehen und bei Bedarf korrigierend

eingreifen kann. Unit Testing und Refactoring haben über eigene Browser ebenfalls ihren Ursprung in Smalltalk. All dies trägt zusammen mit der einfachen Sprachstruktur, der automatischen Speicherverwaltung und der konsequenten Objektorientierung zu einer produktiven Entwicklung bei.

Squeak basiert auf dem heute gültigen Sprachstandard, der 1980 festgelegt wurde und daher den Namen Smalltalk-80 trägt. 1996 hat Alan Kay mit seinem Team zunächst bei Apple, später bei Disney mit der Entwicklung von Squeak begonnen. Sogar die Virtual Machine von Squeak ist in Smalltalk geschrieben. Ein Übersetzer generiert C, sodass die VM mit kommerziellen Smalltalks mithalten kann. Kays Intention war, eine innovative Umgebung speziell für den Einsatz in Schulen zu schaffen. Dabei bildete Smalltalk nur den Kern, die Oberfläche für die Kinder trägt den Namen eToys. In ihr können sie Objekte zeichnen und über grafische Programmierung steuern. Inzwischen basieren zahlreiche Schulprojekte auf Squeak. Mit der Freigabe als Open Source hat sich die Sprache jedoch auch für weitere Programme erschlossen: Wissenschaftliche Projekte, Tools für Mail und das Web, Entwicklungswerkzeuge, Datenbanken und Webserver sowie verschiedene hilfreiche Bibliotheken für die Entwicklung. Croquet ist eines dieser Projekte.

Für die Installation stellt das Croquet-Team Pakete für Mac OS X, Linux und Windows zum Download zur Verfügung. Voraussetzung ist, dass das Betriebssystem OpenGL unterstützt. Nach einem Entpacken sollen die mitgelieferten Startprogramme und Scripts einen schnellen Einstieg ermöglichen. Leider ist etwas händische Nacharbeit zur An-



Neben seinem eigenen Avatar sieht ein Benutzer in einer Croquet-Welt so weit anwesend weitere Avatare, Portale und Fenster (Abb. 1).

passung der Pfade notwendig. Hier zeigt sich der frühe Status. Da Squeak nicht das unterliegende Fenstersystem für die eigene Darstellung nutzt, sind Optik und Bedienung zunächst zwar etwas gewöhnungsbedürftig, aber man findet sich schnell zurecht.

Aus der Anwendung heraus aktualisieren

Vor dem Einstieg in die verteilten Welten von Croquet sollte der Anwender der Anweisung des Programms folgen und sein Image mit den neuesten Änderungen aktualisieren. Dahinter verbirgt sich kein separates Herunterladen und Installieren weiterer Werkzeuge oder Bibliotheken, die Croquet extern erweitern. Vielmehr bietet Squeak – und damit Croquet – eine Aktualisierung direkt aus der Applikation über das Internet an. Leider sieht sich der Anwender stellenweise mit Fragen konfrontiert, die ein Nichttechniker nicht zwingend ohne Hilfe beantworten kann. Lohn der Mühe sind andererseits nicht nur stets fehlerkorrigierter Code, sondern auch Erweiterungen der aktuellen Version.

Nach dem Start zeigt sich der Croquet-Desktop mit einer Information zum System sowie dem Einstieg in die Entwicklung und eine Spielwiese. Mit dem Namen „First Steps“ betitelt, dient diese einer schnellen Demonstration der Croquet-Welt. Ein Klick ändert den Desktop der Anwendung. Die linke Fensterhälfte dominiert eine Toolbox, die eine Übersicht der Oberflächenelemente von Croquet und Squeak zur Verfügung stellt. Ferner ist eine Suche in den Komponenten möglich. Standard ist die Darstellung der Croquet-Kategorie, die unterschiedliche Welten zur Erforschung zur Verfügung stellt. Der Workspace auf der rechten Seite enthält verschiedene Hinweise für den Nutzer.

Als Einstiegstor in die Welt einer Croquet-Applikation dienen das Ziehen und Fallenlassen des gewählten Symbols in die Mitte des Hauptfensters. Abhängig von der Geschwindigkeit des Rechners erscheint mehr oder weniger schnell ein Fenster mit einer dreidimensionalen virtuellen Welt an der Stelle des fallen gelassenen Symbols. Allerdings ist die Applikation zu diesem Zeitpunkt noch nicht fertig geladen. Je nach Komplexität lädt Croquet weiterhin noch Bestandteile, die nach und nach in der Applikationswelt erscheinen.

In diesem Moment betritt der Benutzer einen virtuellen Raum, in der Croquet-Welt Space genannt. Ein Space beherbergt als Container eine Menge von Objekten, darunter den oder die Nutzer. Die Objekte stellt das System in der Regel dreidimensional dar. Sie orientieren sich entweder an Gegenständen aus der realen Welt oder an virtuellen Dingen für die Applikationen. Halbtransparente Avatare vertreten die Benutzer. Sich selbst sieht man als Anwender von hinten, die Navigation erfolgt über die Maus. Ein Klick auf ein Objekt führt dazu, dass eine halbtransparente Linie von den Augen des Avatars zu diesem Objekt erscheint. So können andere Nutzer im Raum sehen, was sich der jeweilige Nutzer gerade anschaut. Für virtuelle Konferenz- oder Schulungsräume ist dies ideal.

Die sich dem Croquet-Neuling darstellenden Welten sind in Jasmine weite Flächen, gepflastert oder mit Rasen, und wattigen, weichen Wolken. Kein nüchterner Desktop, sondern eine stark vereinfachte Welt. Darüber hinaus sind frei schwebende Fenster sichtbar, um die man sich als Anwender herum bewegen kann und die Portale in weitere Räume darstellen. Statische Bilder in den Fenstern repräsentieren diese Welten. Ein Klick auf den senkrechten Pfeil oberhalb eines solchen Fensters bewegt den Avatar direkt davor.

X-TRACT

- Croquet ist ein auf Smalltalk basierendes System, das die weltweite Zusammenarbeit vieler Benutzer in einer offenen 3D-Umgebung gestattet.
- Während der Interaktion sehen alle in einem Raum befindlichen Benutzer das, was die jeweils anderen Personen wahrnehmen.
- Für den Eintritt in die 3D-Welt muss der Anwender kein Plug-in für seinen Webbrowser installieren, sondern er bewegt sich direkt auf dem Rechner, auf dem die OpenGL-Anwendung installiert ist.



In einer alternativen Welt findet hier die Entwicklung einer Croquet-Anwendung unter Wasser statt (Abb. 2).

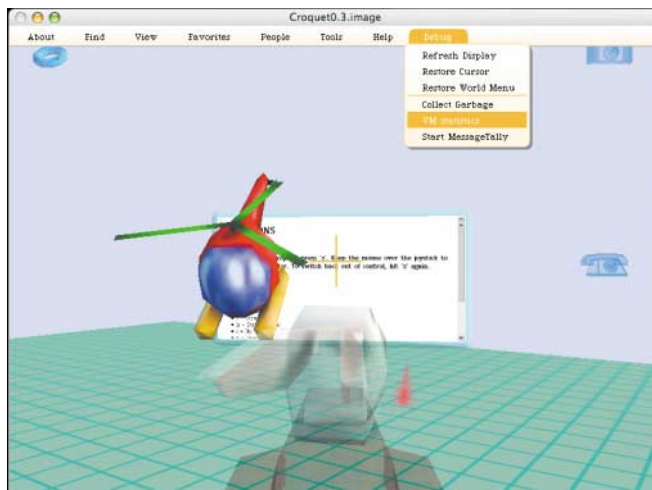
An dieser Stelle zeigt sich das wahre Croquet. Es basiert auf TeaTime, das unter anderem die transparente Replikation versionierter Objekte zwischen den verbundenen Räumen umfasst. Eine einheitliche Zeitbasis im Protokoll sorgt für die korrekte Synchronisation der Objekte zwischen den beteiligten Systemen, auch nach Verzögerungen im Netz. Im Falle einer Zeitüberschreitung erfolgt eine definierte Ausnahmebehandlung, sodass der Zustand des Systems immer sauber definiert bleibt. Das Protokoll und das realisierende Framework erlauben die Entwicklung von Croquet-Objekten so, dass diese sich in der Welt der vernetzten Systeme frei bewegen können. Damit geht Croquet weit über reine 3D-Ergänzungen – etwa Looking Glass von Sun – hinaus. So wesentlich die Visualisierung ist, die Vernetzung von Croquet-Welten bildet den Kern des Systems.

Bewegt sich ein Anwender mit seinem Avatar durch seine Welt, trifft er auf Portale in Form frei schwebender Bilder. Um ein neues Portal zu betreten, klickt er auf ein solches Bild, was bewirkt, dass sich das Portal öffnet und der Benutzer einen ersten Blick in den dahinter liegenden Raum werfen kann. Hierbei handelt es sich nicht mehr nur um eine Abbildung, vielmehr erhält der Benutzer eine echte perspektivische Ansicht von der dynamischen, sich bewegenden Welt hinter dem Portal. Veränderungen des Blickwinkels passen diese Ansicht an. Der Nutzer schaut also nicht auf ein Fenster, sondern durch es hindurch. Allein dieses Spielen mit der oberflächlichen Erkundung der anderen Räume macht schon Spaß. Aber die Möglichkeiten gehen weiter.

Sobald der Benutzer seinen Avatar durch ein Portal führt, betritt er diesen Raum. Andere Nutzer können ihn sehen und alles, was er von nun an tut, findet auf dem Rechner statt, der den Raum beherbergt – also die komplette Berechnung und jede Applikation, die der Anwender dort anstößt. So können mehrere Personen gemeinsam auf einem Rechner eine Simulation starten und beeinflussen sowie sich parallel auf einem zweiten Rechner treffen, um dort die ersten Ergebnisse statistisch zu verarbeiten oder für eine Präsentation aufzubereiten. Eine Rückkehr zur Simulation ist jederzeit möglich. Die Interaktion geht sogar so weit, dass ein Benutzer Objekte aus einem Raum mit in einen anderen Raum nimmt. Sind diese ausführbare Objekte, führt das System im neuen Raum sie weiterhin aus. Croquet stellt sich seinen Nutzern als eine große, aus diversen Einzelteilen zusammengesetzte Welt dar, in der sie sich gemeinsam bewegen und interagieren.

Komplexe Sachverhalte intuitiv erfahren

Die Menge denkbarer Applikationen ist riesig. Individuelle Rechnerwelten werden über das Netz freigegeben, ähnlich heutiger Webserver. Nur dass sich diese Welten nicht nur auf das individuelle Betrachten von stark textorientierten Informationen beschränken. Als interaktive, grafisch orientierte und kollaborative Räume bieten sie dem Besucher mehr. So können Geoinformationssysteme räumliche Informationen dreidimensional darstellen und der oder die



Hubschrauber-Simulation mit Sound: Der Besuch dieser Welt lohnt wegen der plattformübergreifenden 3D-Audio-API OpenAL (Abb. 3).

Besucher können diese Landschaft durchschreiten oder überfliegen. Oder Organisationen stellen sich selbst als virtuelle Gebäude dar, in denen Besucher in direkten Kontakt mit Ansprechpartnern oder anderen Besuchern treten können. Andere Varianten sind zwei- oder dreidimensionale Graphen für die Visualisierung komplexer Zusammenhänge. Der Besucher kann diesen Graphen folgen und durch die Darstellung schnell Zusammenhänge erfassen. Ihn interessierende Knoten kann er als statische Bilder oder Texte darstellen lassen. Gleichzeitig kann man unterschiedliche Multimediaanwendungen oder externe Anwendungen integrieren. Deren Ausgabe erscheint in den Croquet-Fenstern.

Wesentlich hierbei ist, dass der Anwender kein Plug-in in einen Browser installieren muss. Er besucht ja den Anbieter, auf dessen Plattform die notwendigen Applikationen inklusive aller Elemente zur Interaktion laufen. Welchen Standard der Anbieter einsetzt, eine Eigenentwicklung oder das Produkt eines Drittherstellers, ist dem Nutzer gleich.

Neben diesen Anwendungen, die von der vielfältigen Darstellung profitieren, bieten sich Communities mit weiteren Räumen und Interaktionen zwischen den Besuchern an. Gleichzeitig mischen sich Kommunikationsformen wie E-Mail, Chat und Messaging in das Betriebssystem. Die interaktive Kollaboration zwischen Benutzern verteilter Rechnerwelten werden zum Alltag. Den Gang durch die Büroräume ergänzt der Gang durch virtuelle Räume.

Inzwischen hat Croquet die Versionsnummer 0.3. Augen-, oder besser ohrenschneidend ist die Integration von

OpenAL, das für dreidimensionalen Sound die gleiche Rolle übernimmt, wie OpenGL für 3D-Grafik. In einer Demo zu OpenAL kann der Anwender einen Hubschrauber fernsteuern. Und dank der Audiobibliothek erhält er die passende Klanguntermalung inklusive Dopplereffekte dazu.

Eine weitere Änderung ist die zunehmende Nutzung von Tweak als Ersatz von Morphic für grafische Oberflächen. Tweak basiert auf einer asynchronen Kommunikation und Ereignissen, die nahtlos in die Sprache integriert sind. Dies erleichtert den Einstieg ebenso wie

Online-Quellen

Homepage des Croquet-Projekts

www.opencroquet.org

Wiki zu Croquet

minnow.cc.gatech.edu/croquet/

Blog über Croquet und Tweak

croquetweak.blogspot.com

Croquet-Tutorial

www.dmu.com/croquet/

Infos zu Tweak

tweak.impara.de/

der angepasste Systembrowser. Insgesamt erweist sich Tweak als ebenso einfach wie Morphic, zudem flexibel wie das klassische Model-View-Controller-Pattern. Letztendlich wurden alle aktuellen Fehlerkorrekturen und Erweiterungen sowie neue Demos in die beiden Versionen aufgenommen.

Fazit

Noch zeigt sich Croquet in einer jungen Form. Insbesondere die Optik ist im Vergleich zu nativen Oberflächen oder den visuellen Eindrücken moderner Spiele rudimentär. Mit der Versionsnummer 0.3 präsentiert sich das Paket jedoch bereits leistungsfähig. Die 3D-Darstellung sowie die komplexen Szenarien, die über eine reine Simulation hinausgehen, kosten zwar Rechenzeit, doch sowohl CPU-Leistung als auch die Performanz von Grafikkarten steigen schnell. Und so ist es nur eine Frage der Zeit, bis die richtige Hardwarebasis für eine vernetzte 3D-Welt flächendeckend zur Verfügung steht.

Die nächsten Versionen dürften dies ausnutzen. TeaTime soll die Be-

kanntmachung und das Entdecken von Places sowie eine automatische Replikation von Objekten unterstützen. Hintergrundberechnungen und Performanzverbesserungen sind ebenfalls angekündigt. Die Nutzung von Tweak soll durchgängig werden, und die Integration eines Hilfesystems ist ebenso geplant wie Skripting-Unterstützung. Da nicht alles in 3D erscheinen muss, will man außerdem die zweidimensionale Darstellung verbessern.

Mögen viele Anwender sich noch fragen, wozu sie ein System wie Croquet benötigen, erkennen die Early Adopters sicherlich schnell, welches Potenzial darin steckt. Vielleicht wird Croquet nicht zwingend die Basis derartiger Plattformen, doch es wird sie ebenso beeinflussen, wie sich Smalltalk in objektorientierte Sprachen, Entwicklungsprozessen und grafischen Oberflächen wiederfinden lässt. (ka)

FRANK MÜLLER

arbeitet als Systemspezialist bei der Thales Defence Deutschland GmbH in Wilhelmshaven.

