

Squeak 2

Squeak Grundlagen

Heiko Schröder

22. Februar 2006



Inhaltsverzeichnis

1	Einleitung	3
2	Die ersten Schritte	4
2.1	Zusätze unter Linux*	5
2.2	Start	6
2.3	Start über ein Icon	6
2.4	Das World-Menü	7
2.5	Einstellung der Sprache	7
2.6	Einstellung der Schrift	7
2.7	Vollbildschirm	8
2.8	Stiftspuren löschen	8
2.9	Speichern, Kopieren und Verlassen des Images	8
3	Maustasten	8
4	Fenstersteuerung	9
5	Flaps (Laschen)	10
6	Projekte	10
6.1	Anlegen eines neuen Projektes	11
6.2	Speichern des Projektes	11
6.3	Verlassen und gegebenenfalls Löschen des Projektes	12
6.4	Laden eines Projektes	13
7	Kleine Kontrollfragen	13

Abbildungsverzeichnis

1	Ein Squeak-Verzeichnis unter Linux	4
2	Squeak nach dem Start mit einer offenen Lasche	7
3	Testbild für Maustasten	9
4	Vorbereitung eines Projektfensters	11
5	Squeak mit mehreren geladenen Projekten	12

1 Einleitung



Dieses Kapitel ist die zweite vorbereitende Reise. Die Erfahrungen, die Du hier machst, sind für die folgenden Kapitel recht wichtig. Du solltest auf jeden Fall die Kontrollfragen des letzten Abschnitts beantworten können, bevor Du weiterliest.

Natürlich kann und soll – wie auch in allen folgenden Kapiteln – nicht *alles* erklärt werden. Jede Reise soll Dich so weit bringen, dass Du selbst weiterforschen kannst. Mit anderen Worten: Wenn Du die Squeak-Maus am Ende des Kapitels siehst, so ist das *Tutorial* für diese spezielle Reise beendet. Wann die Reise aber selbst zu Ende ist, entscheidest nur Du allein.

An dieser Stelle seien einige wichtige Internetadressen genannt:

www.squeak.org ist die Hauptseite von Squeak. Hier kannst Du stets die aktuellsten Downloads tätigen. Außerdem gibt es hier auch Mailinglisten, in die Du Dich eintragen kannst. Ferner enthält die Seite weitere Links zu anderen Informationsquellen.

www.squeak.de ist die deutsche Seite des Squeak-Projektes

www.squeakland.org Hier findest Du eine Unzahl aktueller Projekte, vor allem zu EToys. Squeakland gibt ein eigenes Squeak-Image heraus, das auf EToys optimiert ist.

Das soll erst einmal reichen. Wie auch in Squeak selbst, wirst Du feststellen, dass die Menge an Informationen eher zu groß als zu klein ist. Und nun viel Spaß beim nächsten Ausflug.

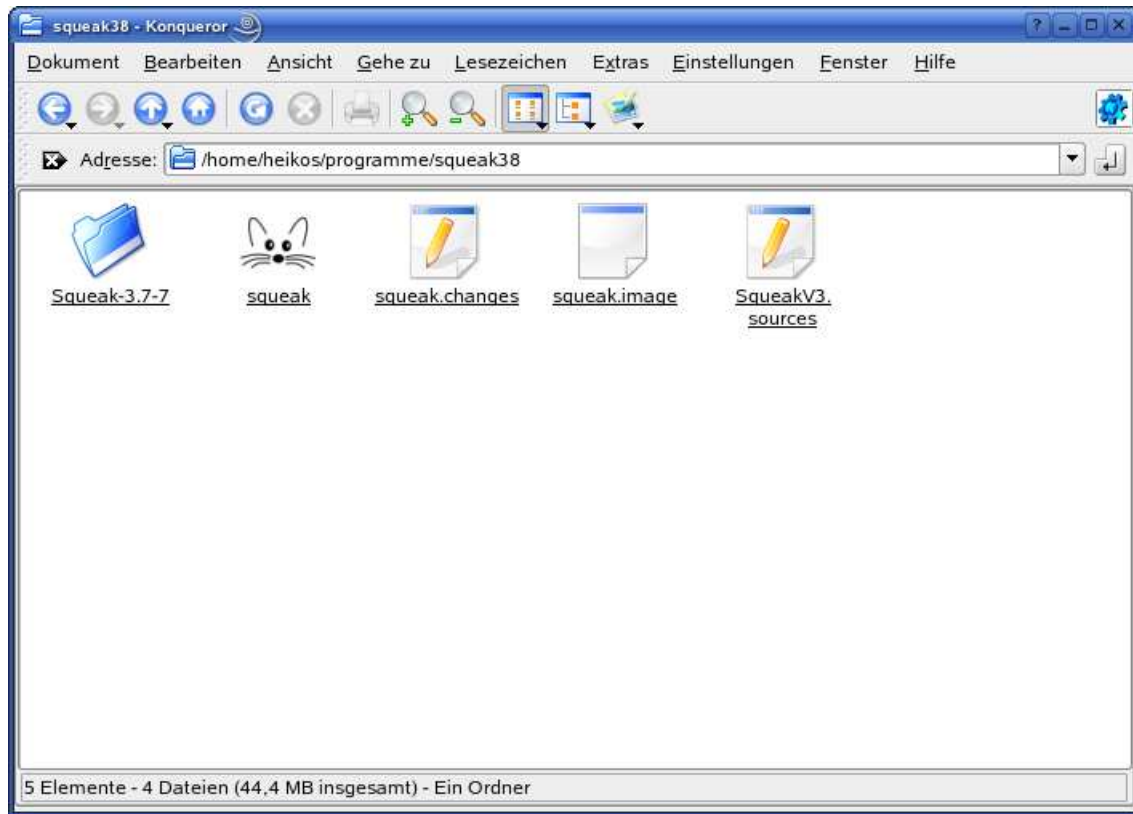


Abbildung 1: Ein Squeak-Verzeichnis unter Linux

2 Die ersten Schritte

Bevor wir uns aber auf die nächste Reise begeben, musst Du natürlich erst einmal Squeak installieren. Ich empfehle Dir, zunächst ausschließlich das Originalpaket zu verwenden, dass Du bei www.squeak.org downloaden kannst. Squeak gibt es für jede Plattform. Und es läuft auch auf jeder Plattform. Die mit Abstand beste Wahl ist zweifellos Linux, da die Belegung der Maustasten der Smalltalk Originalmaus entspricht¹. Squeak ist – um es nicht zu vergessen – Open Source. Es ist zwar keine GPL-Software, aber die Lizenz ist mit dieser durchaus vergleichbar. Am besten informierst Du Dich auf der Webseite von Squeak über den genauen Wortlaut.

Squeak findest Du, zum Beispiel auf dem FTP-Server [ftp.squeak.org](ftp://ftp.squeak.org), als ZIP-Archiv. Zur Zeit, in der ich dieses Tutorial schreibe, lautet die stabile Version zum Beispiel *Squeak3.8-6665-full.zip*. Kopiere dieses Archiv auf Deinem Rechner in ein besonderes Squeak-Verzeichnis Deiner Wahl und entpacke es. Für die meisten Betriebssysteme (außer Linux) war das schon alles. Du brauchst keinen Systemadministrator für diese Installation.

In Deinem Squeak-Verzeichnis solltest Du nun vier wichtige Dateien vorfinden. Unter Linux sind es zunächst nur drei. Die vierte ist etwas versteckt. Die Abbildung 1 zeigt Dir mein Squeak Verzeichnis unter Linux. Bei Windows sehen lediglich die Icons etwas anders aus.

.image: Eine Datei trägt die Namenserweiterung *.image*. Der Name dieser Datei ist oft sehr lang. Zum Beispiel *Squeak3.8-6665.image*. Diese Datei enthält die eigentliche

¹ Die erste Maussteuerung wurde mit Smalltalk verbreitet. Der Erfinder der Computermaus ist Douglas ENGELBART.

Squeak-Umgebung. Sie wird allgemein das *Image* genannt. Dieses Image liegt in dem nicht lesbaren Binärcode vor. Dieser Binärcode ist *hardwareunabhängig*. Er läuft also auf jeder Maschine, auf der Squeak läuft. Die Hardwareunabhängigkeit bedeutet aber auch, dass die Datei in dieser Form noch nicht ausführbar ist. Das Image ist – um mit Java zu reden – sozusagen ein riesiges Applet, das von einer Virtual Machine (VM) *interpretiert* werden muss. Du kannst verschiedene Images zur Verfügung haben.

squeak: Diese VM ist nun jene Datei mit dem Namen *squeak*. Unter Windows ist es diejenige Datei, die durch die Squeak-Maus gekennzeichnet ist. Unter Linux scheint sie nach dem Entpacken zu fehlen. Doch das ist nicht so. Genauer dazu wird gleich in den Installationszusätzen zu Linux erklärt 2.1. Diese VM ist also das Bindeglied, das *Interface*, zwischen dem Image und dem Prozessor des Rechners. Sie ist – allgemein gesprochen – der *Interpreter*. Im Gegensatz zum Image muss die VM zu dem entsprechenden Betriebssystem *passen*. Das ist auch bei der in den Browsern eingebauten VM von Java nicht anders. Hardwareunabhängig ist also nur das Applet, nicht der Interpreter. Die Squeak-VM ist allerdings selbst in Squeak-Smalltalk geschrieben, so dass Du sie im Prinzip beliebig umprogrammieren kannst, wenn Du Deine eigene Programmiersprache erfinden möchtest.

sources: Die Datei mit der Bezeichnung *sources*, meistens lautet sie *SqueakV3.sources*, enthält den lesbaren *Quellcode* des ausgelieferten Images. Du solltest als Anfänger an diesem Code nichts verändern! Während der ganzen Arbeit mit Squeak bleibt diese Datei unangetastet.

.changes: Anders sieht es mit der vierten Datei aus. Programmieren in Smalltalk bedeutet *immer* eine Veränderung der Umgebung, des Images. Der Quelltext, mit dem diese Veränderungen erfolgen, wird in der Changes Datei mitprotokolliert. Und auch der Name des Urhebers dieser Veränderungen. Wenn Du an irgendeiner Stelle bei der Arbeit mit Squeak dazu aufgefordert wirst, Deinen Namen oder Deine Initialen einzugeben, so habe keine Bedenken bezüglich des Datenschutzes. Squeak will diesen Namen nur in die Changes-Datei schreiben. Dadurch wird bei einem Projekt klar, welcher Entwickler welche Veränderungen vorgenommen hat. Wichtig ist, dass die Changes-Datei grundsätzlich – mit Ausnahme der Endung *.changes* – exakt denselben Namen wie das Image trägt. Wenn das Image *Squeak3.8-6665.image* heißt, so *muss* die Changes-Datei auch *Squeak3.8-6665.changes* lauten. Benennst Du die Image-Datei in *squeak.image* um, so musst Du entsprechend den Namen der Changes-Datei auch in *squeak.changes* verwandeln. Falls Du mehrere Images in Deinem Verzeichnis verwaltest, gehört natürlich zu jedem eine Changes-Datei.

Nun sind wir fertig für den ersten Start. Lediglich die Verwender von Linux sollten noch den nächsten Abschnitt lesen. Die anderen können zum übernächsten springen.

2.1 Zusätze unter Linux*

Bei Linux ist die VM im Squeak-Verzeichnis noch nicht sichtbar. Bei Linux gehen die Entwickler davon aus, dass Du die Multiuser-Fähigkeiten ausnutzen möchtest. Daher ist eine ergänzende Installation notwendig. Falls Du diese Installation aber nicht vornehmen möchtest, kannst Du die VM auch in das oberste Verzeichnis Deines Squeak-Verzeichnisses kopieren. In diesem obersten Verzeichnis findest Du einen Unterordner. Zum Beispiel *Squeak3.7.7* in der Abbildung. Die abweichende Versionsnummer bezieht sich auf die VM, nicht auf das Squeak-Image. Jetzt gilt es, den Verzeichnisbaum einmal hinabzusteigen. Nach der ersten

Stufe siehst Du bereits zwei Skript-Dateien *config.guess* und *install*. Die letztere, *install*, nimmt die angesprochene Installation vor.

Installation: Falls Du dies möchtest, kannst Du diese Datei in einem Editor betrachten. Dort findest Du die einfache Anleitung, allerdings auf Englisch. Im Wesentlichen steht dort nur das, was ich Dir jetzt sage. Allerdings ist den Autoren bei der Beschreibung ein kleiner, aber ärgerlicher Fehler unterlaufen. Den vermeiden wir natürlich. Verlasse den Editor wieder und logge Dich mit *su* oder *su root* als Systemadministrator ein (falls Du das darfst). Rufe dann */bin/sh install* auf. Und hier steht leider in der Anleitung von *install* der Aufruf */sh/bin INSTALL*, was falsch ist, denn die Datei *INSTALL* existiert nicht. Verlasse danach den Administratormodus. Der Aufruf von *squeak* über ein Terminal liefert aber immer noch eine Fehlermeldung. In jedem Falle sollte das Squeak-Verzeichnis in der Variable *PATH* eingetragen sein. Dann weiß die VM noch nicht, welches Image aufgerufen werden soll. Entweder mußt Du den Pfad der Image-Datei beim Aufruf direkt angeben. Das hat den Vorteil, beliebige Images aufrufen zu können. Oder Du trägst in deiner *.bashrc* die Zeile *export SQUEAK_IMAGE=<absoluter Pfad zum Image>* ein. Dann startet Squeak (neuen Shellstar nicht vergessen!) beim Aufruf *squeak* über ein Terminal immer dieses in der Variablen *SQUEAK_IMAGE* eingetragene Image. Der Start von Squeak über ein Icon wird weiter unten beschrieben 2.3.

Kopie: Falls Du keine Installation möchtest, aber auch sonst, empfehle ich Dir die Kopie der VM in das oberste Verzeichnis, in dem sich auch die anderen drei wichtigen Dateien befinden. Dazu mußt Du weiter den Verzeichnisbaum hinabsteigen. Gehe so weit hinunter, bis Du in ein Verzeichnis mit den Unterverzeichnissen *bin*, *doc*, *lib*, *man* gelangst. Die Abfahrt geht dann über *lib* weiter. Nach drei weiteren Stufen bist Du am Ziel und findest eine Datei mit dem Namen *squeak*. Das ist die VM. Warum die Entwickler sie so versteckt haben gehört zu den ungelösten Rätseln dieser Welt. Kopiere die Datei (vielleicht reicht auch eine Verknüpfung) in das oberste Verzeichnis, in dem sich die anderen Dateien befinden.

2.2 Start

Unter allen Betriebssystemen – auch bei Linux, wenn die VM entsprechend kopiert wurde – ziehst Du mit der Maus das Image auf die Datei mit dem Namen *squeak* und läßt sie fallen. Wenn alles reibungslos abläuft, siehst Du die bunte Umgebung von Squeak so ähnlich wie sie die Abbildung 2 auf der nächsten Seite zeigt. Unter Linux gibt es auch den Start über die Konsole. Wenn die Installation korrekt vorgenommen wurde, genügt entweder der Aufruf *squeak*, wenn die Variable *SQUEAK_IMAGE* wie beschrieben gesetzt wurde, oder es funktioniert der Aufruf *squeak <absoluter Pfad zum Image>*.

2.3 Start über ein Icon

Vielleicht möchtest Du gerne Squeak über ein Icon auf dem Desktop starten. Das geht grundsätzlich. Es muß aber für jedes Image ein eigenes Icon erstellt werden. Denn woher soll die VM sonst wissen, welches Image gestartet werden soll. Unter Linux richtest Du Dir ein Icon ein und verbindest es mit dem Befehl *squeak <absoluter Pfad zum Image>*. Bei mir heißt das zum Beispiel *squeak /home/heikos/squeak/squeak38/squeak.image*.

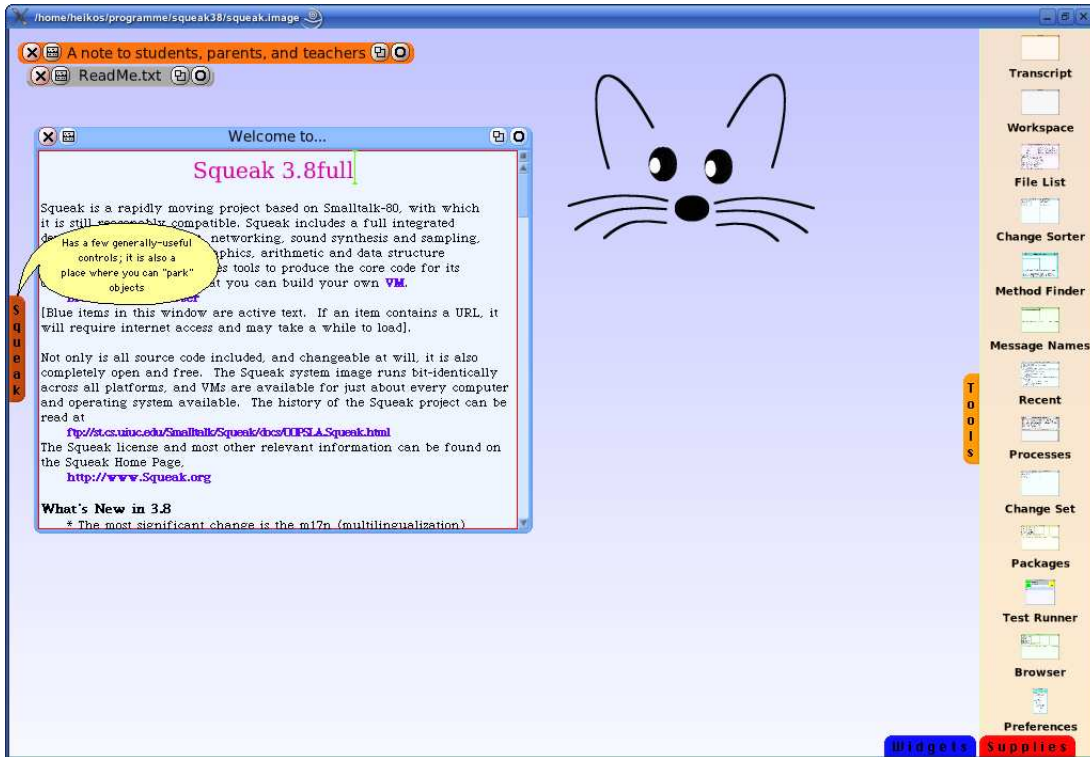


Abbildung 2: Squeak nach dem Start mit einer offenen Lasche

2.4 Das World-Menü

In der Squeak-Umgebung wird Dir vieles von grafischen Betriebssystemen her vertraut vorkommen. Kein Wunder. Die grafischen Betriebssysteme gehen alle auf Smalltalk-Ideen zurück. Es gibt nun ein zentrales Menü, sozusagen die »Schaltzentrale« von Squeak. Dieses erhältst Du durch einen Klick mit der Maus auf den freien Squeak-Desktop. Über dieses World-Menü kannst Du jede Aktion vornehmen, die Squeak versteht. Im Tutorial werde ich Dir in der Regel ausschließlich Handgriffe zeigen, die schneller als über das World-Menü gestartet werden können. Grundsätzlich gilt: Wenn irgendeine Aktion nicht so funktioniert wie sie soll oder Du irgendeinen Handgriff nicht findest, so kannst Du dies auch immer über das World-Menü ebenfalls erledigen. Wir betonen das nicht jedesmal neu.

2.5 Einstellung der Sprache

Üblicherweise ist die voreingestellte Sprache Englisch. Über *World-Menü* \triangleright *help...* \triangleright *set language...* kannst Du das ändern. Ein anderer, vielleicht schnellerer, Weg geht über die Navigator-Lasche (siehe Abschnitt 5 auf Seite 10). Wir gehen im Folgenden davon aus, dass Du alles auf Deutsch umgestellt hast.

2.6 Einstellung der Schrift

Über *World-Menü* \triangleright *Einstellungen...* \triangleright *Systemschriften* kannst Du jede beliebige Schriftgröße anpassen. Sehr wahrscheinlich erscheinen Dir die voreingestellten Schriften (zum Beispiel die Schrift der Menüs) zu klein.

2.7 Vollbildschirm

Über die *Einstellungen* kannst Du auch den Squeak-Desktop auf den ganzen Bildschirm ausdehnen. Damit befindest Du Dich völlig in der Welt von Squeak. Wenn Du allerdings einen Laptop im Batteriebetrieb am Laufen hast, bedenke bitte, dass Du dann die Batterieanzeige nicht mehr siehst. Über dasselbe Menü kannst Du den Vollbildschirm auch wieder beenden.

2.8 Stiftspuren löschen

Du kannst irgendwelche Turtle-Zeichnungen direkt auf dem Desktop vornehmen. Nur: wie bekommst Du sie wieder weg? Über die *Einstellungen* siehst Du einen Eintrag »Stiftspuren löschen«. Das ist der richtige Ort.

2.9 Speichern, Kopieren und Verlassen des Images

Im World-Menü hast Du sicher die Einträge im unteren Abschnitt bemerkt. Vor dem Verlassen von Squeak solltest Du unbedingt den aktuellen Stand des Images speichern. Du kannst auch eine neue Kopie des Images mit verändertem Namen vornehmen. Gleichzeitig wird eine entsprechende Changes-Datei in Deinem Squeak-Verzeichnis erstellt. Erstelle Kopien des Images immer aus Squeak selbst heraus.

3 Maustasten

Ein kleines, aber lösbares Problem, sind die Tastenbelegungen der Maus bei den unterschiedlichen Betriebssystemen. Die ersten Smalltalk Mäuse besaßen von Anbeginn drei farbige Tasten. Von links nach rechts in der Reihenfolge rot, gelb und blau. Nur unter Linux / Unix entsprechen diese Tasten derselben Reihenfolge (Drei-Tasten-Maus vorausgesetzt).

Schaue Dir die Abbildung 3 auf der nächsten Seite an. Bei einem Klick mit der Maus auf den freien Desktop sollte bei der roten Taste das linke World-Menü erscheinen. Beim Klick mit der gelben Taste erscheint das mittlere, kleine Kontextmenü. Ein Klick mit der blauen Taste auf ein Auge der Squeak-Maus läßt den bunten Halo erscheinen, den wir beim Umgang mit Grafiken (EToys) kennenlernen werden.

Grundsätzlich empfehle ich Dir dringend die Anschaffung einer Drei-Tasten-Maus. Du mußt allerdings austesten, welche Taste bei Deinem Betriebssystem die rote, gelbe und blaue ist. In der Regel ist unter Windows die rote Taste wie gewohnt auch links, aber die gelbe Taste ist die rechte Maustaste. Die blaue Taste ist die mittlere. Bei Linux ist die Reihenfolge – wie schon gesagt – links, mitte, rechts.

Die rote Taste dient zum Aktivieren irgendwelcher Aktionen. Lediglich der Klick auf den freien Desktop liefert ein Menü, das Du schon bereits als World-Menü kennst. In unserem Text kennzeichnen wir einen roten Mausklick mit dem Symbol @r. Wenn nur von einem Mausklick ohne Angabe gesprochen wird, ist *immer* der rote Mausklick gemeint. Jedes Ziehen (Drag) eines Objektes wird immer mit festgehaltener roter Taste vorgenommen.

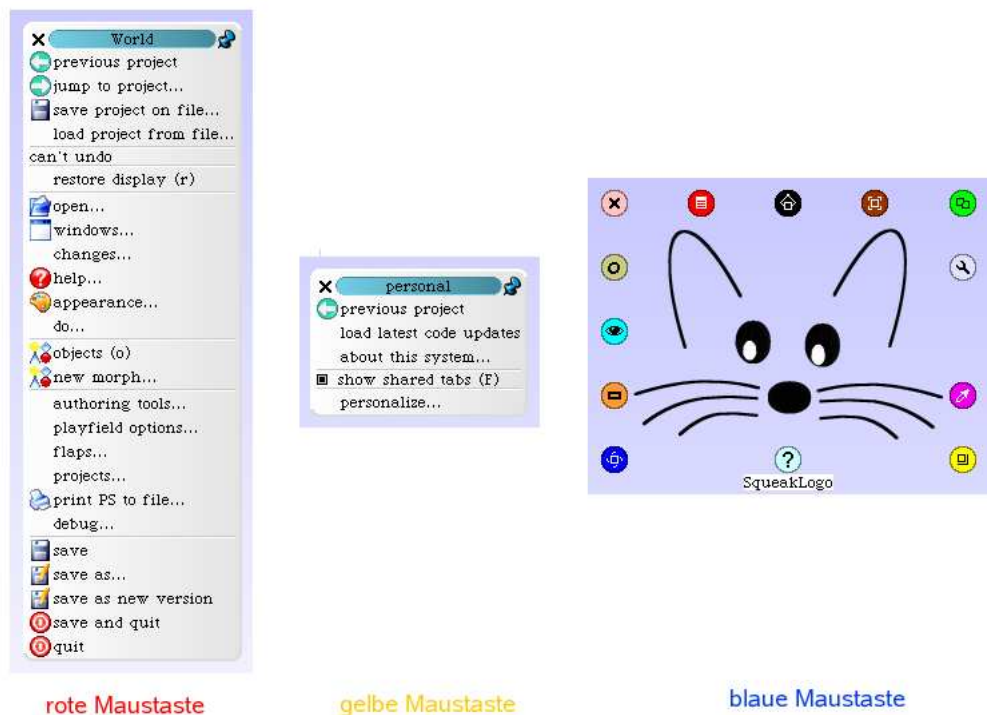


Abbildung 3: Testbild für Maustasten

Die gelbe Taste holt zu dem angeklickten Objekt ein Kontext-Menü. Das ist Dir sicherlich auch von den Betriebssystemen her bekannt. Ein gelber Mausklick bezeichnen wir in unserem Text mit dem Symbol @g. Wenn von dem Aktivieren eines Kontextmenüs die Rede ist, bedeutet das *immer* den gelben Mausklick auf das Objekt.

Die blaue Taste aktiviert jenen bunten Halo beim Anklicken einer Grafik. Genauer gesagt, eines Morphs. Ein Morph ist eine Grafik, die als Objekt selbstständig ist. Eine Turtle-Grafik dagegen ist *kein* Morph. Die bunten »Smarties« sind so genannte *Handles*, die ganz bestimmte Aktionen ermöglichen. Die Handles werden stets mit der roten Maustaste aktiviert. Einen blauen Mausklick bezeichnen wir natürlich mit @b. Oder wir sagen »Aktiviere den Halo«. Dann ist *immer* der blaue Mausklick gemeint.

4 Fenstersteuerung

Fenster sind Dir sicherlich aus der Welt der grafischen Betriebssysteme bekannt. Beim Start von Squeak siehst Du sicherlich wenigstens ein Fenster, das geöffnet ist. Wenn nicht, gehe zu den beiden nächsten Abschnitten über und kehre danach hierher zurück. Was die einzelnen Symbole des Fensterrahmens bedeuten, erklärt Dir die Balloon-Erklärung, sobald Du mit der Maus über das Symbol gefahren bist. Bekannt ist sicherlich das Kreuz X, mit dem ein Fenster geschlossen wird. Hoppla: In der korrekten objektorientierten Sprache sendest Du einem Fenster damit eine Nachricht, sich selbst zu schließen. Ein Fenster kannst Du auch durch Ziehen so verschieben, wie es Dir von Betriebssystemen bekannt ist. Fährst Du mit der Maus über ein solches Fenster, erscheint ein Schatten, der besagt, dass das Fenster jetzt für Mausektionen aktiviert ist.

Verkleinerte Fenster zeigen nur noch den farbigen Fensterrahmen:



Ein Fenster ist aber auch ein Morph, der einen Halo besitzt. Führe auf einem geöffneten Fenster @b aus. Mit dem grünen Handle kopierst Du das Fenster. Dazu musst Du das Handle ziehen. Mit dem blauen Handle kannst Du es drehen. Wenn Du willst, probiere alle Handles aus und stelle fest, was sie bedeuten.

5 Flaps (Laschen)

Am auffälligsten sind wahrscheinlich die bunten Laschen. Du solltest nach dem Start und der Sprachumstellung wenigstens die Laschen Squeak (braun), Navigator (orange), Geräte (blau), Lager (rot) und Werkzeugleiste (orange) sehen. Wenn irgendeine Lasche nicht sichtbar sein sollte, kannst Du sie über *World-Menü* ▷ *Klappenmanager* aktivieren. Eine Lasche öffnest Du mit @r. Aber für das Herausziehen eines Objekts genügt ein Klick nicht. Dazu musst Du das Objekt auf den Desktop ziehen.

Werkzeugleiste (Tools): Hier findest Du alle Werkzeuge, die Du für die Smalltalk-Programmierung benötigst. Und wahrscheinlich sehr viel mehr.

Lager: Hier findest Du vor allem Prototypen für die Programmierung von Morphem.

Geräte: Diese Lasche enthält fertige »Programme«, die eine ganz bestimmte Aufgabe erfüllen sollen. Hier befindet sich zum Beispiel eine Lupe, die die Umgebung unter dem Mauscursor vergrößert (probiere es aus). Ferner siehst Du hier den Malkasten zum Entwerfen eigener Morphe, sowie die Skriptkontrolle für Arbeiten mit EToys. Sehr witzig, aber in ihrer Funktion wohl noch nicht durchschaubar ist die »Teilchengrippe«. Lass Dich einmal überraschen. Wie Du sie wieder loswirst, weißt Du ja. Wenn nicht, probiere mal @b aus.

Navigator: Die Knöpfe in dieser Lasche dienen zur Navigation zwischen Projekten. Da wir noch nicht wissen, was das ist, können wir den Sinn dieser Lasche an dieser Stelle noch nicht verstehen.

Squeak: Hier nun finden sich die Einträge, die die ganze Squeak-Umgebung betreffen. Unter anderem auch der wichtige Mülleimer. Was »gelöscht« ist, ist nicht *wirklich* weg. Du kannst es über den Mülleimer wieder aktivieren. Wie öffnest Du den Mülleimer? Beachte die ausführliche Balloon-Erklärung.

Natürlich kannst Du Dir später auch eigene Laschen generieren. Bei Squeak gibt es fast nichts, was es nicht gibt.

6 Projekte

Sehr wahrscheinlich ist Dir schon im World-Menü das vielsagende Wort *Projekt* begegnet. Ein Projekt ist – grob gesagt – eine »Spielwiese« für fertige Objekte. In der Regel handelt

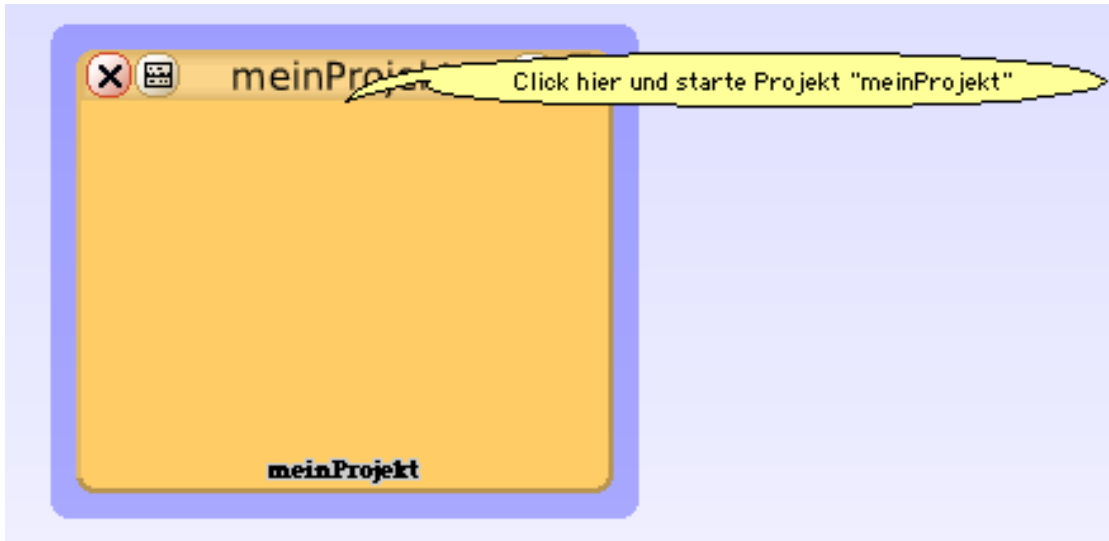


Abbildung 4: Vorbereitung eines Projektfensters

es sich dabei um grafische Objekte, also Turtles oder Morphe. Während die Smalltalk-Programmierung tiefer lotet, indem sie das Image, also die »Natur«, verändert, stellt ein Projekt eine eigene Lebensgemeinschaft dar, die zwar von der »Natur« abhängig ist, diese aber nicht verändert.

6.1 Anlegen eines neuen Projektes

Ein neues Projekt öffnest Du über den Navigator durch @r auf den Button »Neu«. Es entsteht ein kleines Projektfenster, in der Regel mit farbigem Hintergrund (siehe Abbildung 4). Das Projekt trägt vorerst den Namen »Ohne Namen«. Dieser Name ist an zwei Stellen sichtbar: Oben im Rahmen, wo Du ihn nicht verändern kannst und unten in kleiner Schrift als Teil des Fensters. Klicke auf diesen unteren Teil. Die Schrift wird grün unterlegt und ändert selbst ihre Farbe in rot. Gib jetzt den Namen *meinProjekt* ein und achte darauf, was am Fensterrahmen passiert. Du stellst fest, dass sich dort *gar nichts* verändert. Der neue Name wird erst akzeptiert, wenn Du es ausdrücklich mitteilst. Das kann durch die Return-Taste geschehen oder aber auch durch Eingabe von ALT-s. Allgemein wird durch ALT-s das System gebeten, eine bestimmte Aktion anzunehmen.

Wenn Du alles in dieser Weise vorgenommen hast, sollte die kleine Schrift am unteren Rand wieder schwarz erscheinen. Vor allem hat sich der Text des Fensterrahmens in *meinProjekt* verändert. Klicke nun in das Projektfenster hinein². Das Fenster breitet sich über den ganzen Desktop aus. Ziehe nun zum Beispiel die Teilchengrippe (bouncing atoms) aus der blauen Lasche »Geräte« in das Projektfenster hinein. Und vielleicht noch etwas Anderes. Bist Du ganz mutig, so zeichnest Du Dir vielleicht Deinen ersten Morph mit dem Malkasten selbst.

6.2 Speichern des Projektes

Klicke im Navigator auf »Veröffentlichen³«. Es erscheint ein kleines Karteifenster, in dem Du einige Einträge vornehmen solltest. Du musst für diesen Test nicht alle Zeilen ausfüllen.

² Hältst Du @r über dem Fenster fest, erscheint ein Kontextmenü, falls Dich weitere Informationen interessieren.

³ Hältst Du @r länger über dem Button fest, erscheint ein Kontextmenü.

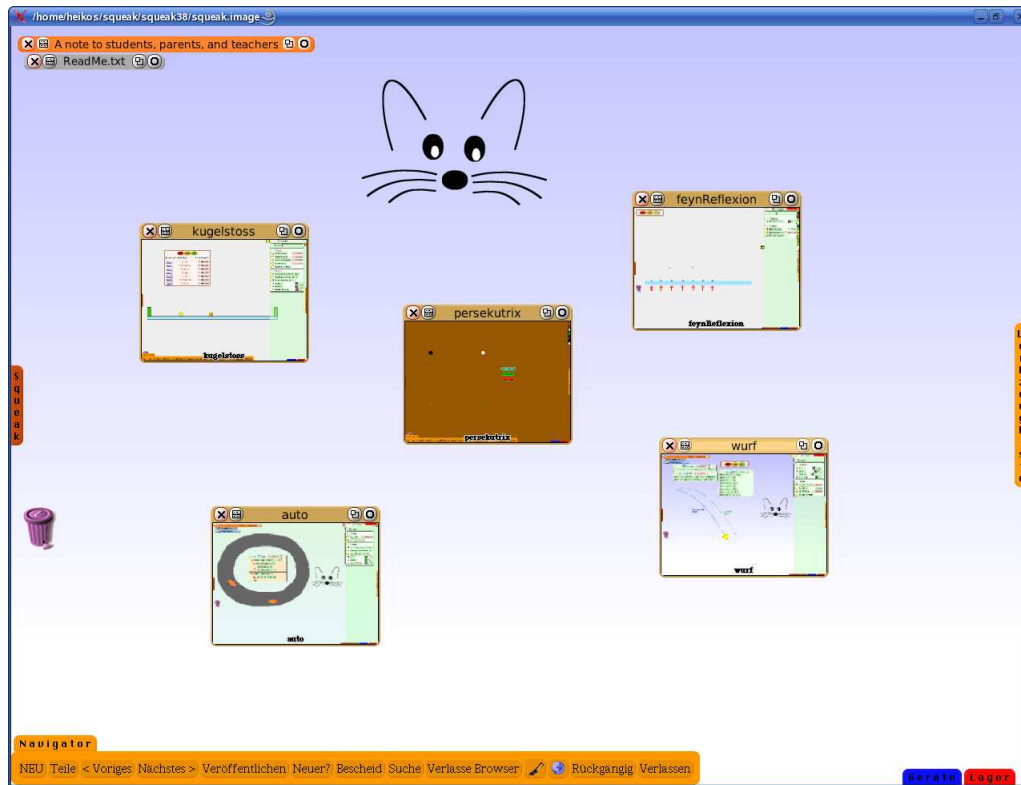


Abbildung 5: Squeak mit mehreren geladenen Projekten

len. Danach erscheint ein blaues Fenster mit dem Verzeichnisbaum auf Deiner Festplatte. Ein Projekt kannst Du auf dem öffentlichen Squeak-Server speichern (das tue bitte jetzt *nicht*) oder aber auf Deiner Festplatte. Wähle in dem Fenster den Knopf »Nur auf lokaler Festplatte speichern« aus. Dann sollte alles mit dem Hinweis »Publishing« automatisch ablaufen.

Schaue nun in Dein Squeak-Verzeichnis. Ja, wo ist denn nun das Projekt abgeblieben? Achte immer auf die Balloon-Erklärungen bei jeder neuen Tätigkeit. Der Button »Nur auf lokaler Festplatte speichern« sichert ein Projekt in dem Unterverzeichnis *Squeaklets*. Schaue einmal dort nach. Du findest dort hoffentlich die Datei *meinProjekt.001.pr*. Willst Du das Projekt an anderer Stelle speichern, so benutze nicht den Knopf »Nur auf lokaler Festplatte speichern«, sondern »Speichern«. Voraussetzung ist natürlich, dass Du in dem Verzeichnisbaum ein bestimmtes Verzeichnis ausgewählt hast. Vielleicht fällt Dir auf, dass zusätzlich noch eine GIF-Datei mit dem Abbild Deines Projektes angelegt wird. Dies entspricht einer Momentaufnahme kurz vor dem Speichern.

6.3 Verlassen und gegebenenfalls Löschen des Projektes

Hier gibt es beim Navigator eine kleine »Falle«: Bitte klicke nicht auf den Knopf »Verlassen«, sondern auf den Knopf »Voriges«. »Verlassen« bedeutet »Squeak verlassen«, was Du jetzt sicherlich nicht willst. Nach dem Druck auf »Voriges« verkleinert sich das Projektfenster wieder auf seine ursprüngliche Größe. Alle Bewegungen *in* dem Projekt werden eingefroren. Das kleine Bild, was Du siehst, ist also eine Momentaufnahme des letzten Zustands, wie wir ihn im vorigen Abschnitt beschrieben haben.

In der Abbildung 5 siehst Du eine große Anzahl geladener Projekte. Willst Du ein Projekt

lowerden, so schlieÙe einfach das Fenster. Du wirst noch einmal vorher über die Richtigkeit Deines Vorhabens befragt. Achtung: ein gelöschttes Projekt ist aus der Squeak-Welt verschwunden. Es ist nicht im Mülleimer zu finden, noch kann es über den Knopf »Rückgängig« wiederhergestellt werden. Es muß von der Festplatte neu geladen werden.

6.4 Laden eines Projektes

Klicke im Navigator auf den Knopf »Suche«. Es öffnet sich wieder ein blaues Fenster mit einem Verzeichnisbaum. Suche in Deinem Squeak-Verzeichnis das Unterverzeichnis »Squeaklets«. Dort solltest Du Dein eben angelegtes Projekt wieder finden. Wähle es aus und klicke auf OK. Achtung: Falls Du das Projekt mehrfach abgespeichert hast, wird in dem Fenster immer die neueste Version zu sehen sein. Über @g im Auswahlfenster rechts erhältst Du ein umfangreiches Kontextmenü. Natürlich kannst du auch über das World-Menü Projekte laden. Dort siehst Du dann auch ohne weitere Einstellungen die älteren Versionen. Hast Du eine Verbindung zum Internet, solltest Du auch in der Lage sein, auf diesem Wege in den Squeakland Verzeichnissen zu suchen.

7 Kleine Kontrollfragen

Frage 1 *Wie startest Du Squeak?*

Frage 2 *Welche Bedeutung hat die Datei mit der Endung .changes? Wozu wirst Du manchmal während der Arbeit mit Squeak nach Deinem Namen gefragt?*

Frage 3 *Was bedeuten in unserem Tutorial die Bezeichnungen @r, @g und @b?*

Frage 4 *Was ist ein Halo? Besitzen alle Objekte einen Halo?*

Frage 5 *Wie kannst Du Turtle-Strichspuren auf dem Desktop wieder löschen? Wie solltest Du besser vorgehen, als direkt auf den Desktop zu zeichnen?*

Frage 6 *Finde heraus, wie Du einen Eintrag aus dem Mülleimer herausholst.*



Index

Engelbart, Douglas, 4

Projekt, 10