

科学リテラシ習得のためのプログラミング環境「スクイーク」

阿部 和広

多摩美術大学 芸術学部 情報デザイン学科

近年、教育用のプログラミング環境として注目されているスクイークについて、開発者のアラン・ケイ博士、その基となったダイナブックやスモールトークなどの思想的背景、オブジェクト指向やタイルプログラミングなどの特徴などについて解説する。また、特に科学リテラシ習得に向いているとされる点について説明する。

Programming Environment “Squeak” for Developing Scientific Literacy

Kazuhiro Abe

Department of Information Design, Tama Art Univ.

This paper introduces a programming environment “Squeak”. Squeak was developed by Dr. Alan Kay and based on his Dynabook concept and Smalltalk. Squeak is good for developing scientific literacy of children.

1. はじめに

最近、教育の現場で Squeak(スクイーク)という言葉がよく聞かれるようになった。スクイークは、米国のコンピュータ科学者、アラン・ケイ博士を中心に開発されているパーソナルコンピューティング環境、つまり、個人がコンピュータを使うための仕組みである。ここでは、このスクイークについて概要を説明する。

2. ダイナブックとスクイーク

ケイは、「パーソナルコンピュータの父」と呼ばれ、大型コンピュータ全盛の 1968 年に今日のパソコンの基礎となる概念、Dynabook(Dynamic + Book, ダイナブック)を提唱したことで知られている。ダイナブックは、平面ディスプレイや無線ネットワークなどを備えた携帯可能なコンピュータであり、画面に表示されたウィンドウやポップアップメニューなどをペンで操作することで、誰でも簡単に使うことができるとされた。ダイナブックはダグラス・エンゲルバートの NLS、アイヴァン・サザーランドの SketchPad、シーモア・パパートの LOGO などの影響を受けている。

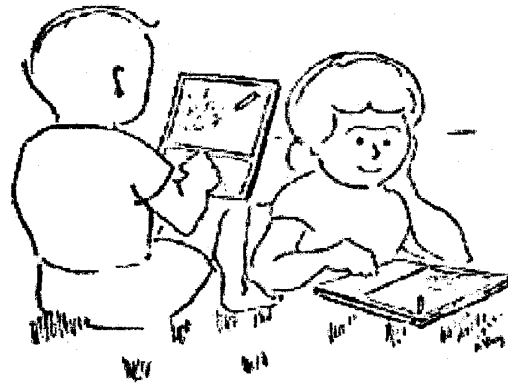


図1 ダイナブック

ダイナブックは、まずダンボールの模型が作られ、1973年に米ゼロックス社パロアルト研究所(PARC)で実際に動作する暫定版、ALTO(アルト)が完成した。アルトは、小さな机くらいの大きさで、ビットマップディスプレイとマウス、イーサネットの基になったネットワーク機能などを備えていたことから、世界初のパーソナルコンピュータとされている。

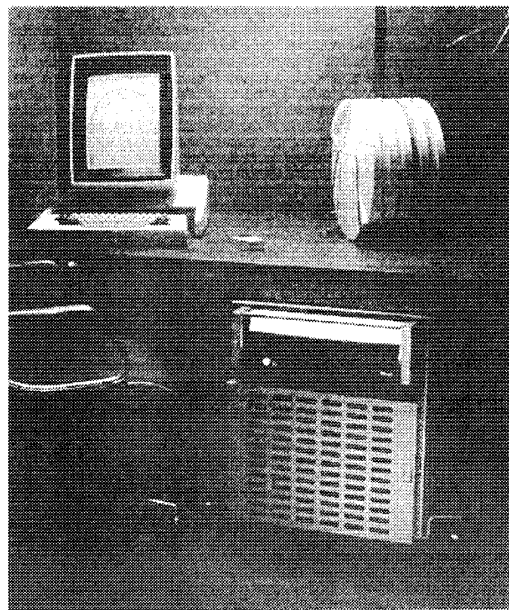


図2 アルト

3. スモールトークとスクイーク

前述のように、ケイはパパート、さらにはジャン・ピアジェの影響を受けており、ダイナブックも子供の発達段階に応じた学習に使うことを前提に設計されていた。暫定版ダイナブックであるアルトには、LOGOのタートルグラフィックスだけでなく、OSやアプリケーションを区別することなく統一的に扱うことのできる Smalltalk(スモールトーク)というプログラミング環境が組み込まれており、PARCに招かれた子供たちはアルトを使って新しい道具を作ったり、拡張したりすることができた。

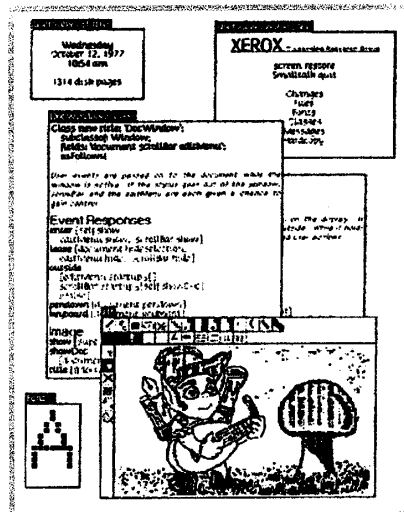


図3 スモールトーク

今では当たり前になった GUI やオブジェクト指向の考え方も、スモールトークで実現されたものである。しかし、現在のパソコンとダイナブックはかなり異なるものになっている。

一方、今日学校で行われているコンピューティングリテラシ(読み書き能力)の教育は、OS やワープロ、電子メール、Web ブラウザなどのアプリケーションの使い方の学習を主としている。当時に比べ現代のパソコンの性能は非常に高く、その潜在能力は大きいですが、既存の OS やアプリケーションを使っているだけでは、その一部しか引き出すことができない。また、これらがブラックボックス化されているため、自分でそれを作ったり、変えたりできるという発想も忘れられてしまいがちである。ケイが考えたコンピューティングリテラシには、本来プログラミングも含まれていた。

スクイークの開発は、このような状況を本来の姿に戻すべく 1996 年に始まった。そのベースになっているのがスモールトークである。スクイークは現在オープンソースのプロジェクトとして世界中で開発が進められている。

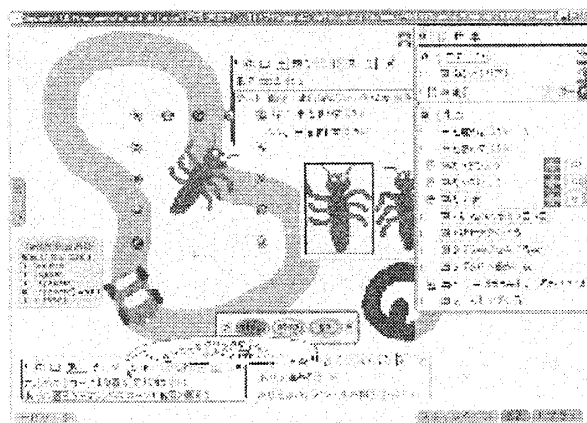


図4 スクイーク

スクイークは無償でインターネットからダウンロードすることができ、Windows, MacOS, Linux などほとんどのプラットフォームに対応している。

4. 従来のプログラミング

プログラミングの重要性が分かったとしても、子供にとって実際にプログラムを書くことは難しい。まず、プログラミング言語の文法や、エディタ、コンパイラなど開発ツールの使い方を覚えなければならないし、苦勞して入門書を読み終えても、結局は簡単なものしか作ることができず、実際にやりたいことまで届かないこともある。

このように、目的と手段のギャップが大きいことが、結果としてプログラミングへの興味を失わせる原因になっているように感じる。また、本来はなににかを実現するための手段であったプログラミングが目的になってしまっているようにも思われる。

従来、教育用の言語として、BASIC, LOGO, Pascal などが試されてきたが、これらの普及や定着がうまくいかなかった背景には、このような問題があったのではないだろうか。

5. スクイークにおけるプログラミングの特徴

5.1. タイルスクリプト

スクイークによるプログラミングの特徴のひとつに、その習得にかかる時間が短いことがある。

従来のプログラミング言語では、文字によってプログラムを記述するが、スクイークの eToys(イートイ)というプログラミング環境では、「タイル」と呼ばれる板状のアイコンをマウスで並べたものがプログラム(スクリプトと呼ぶ)になる。

これにより、キーボード操作にそれほど習熟していない子供でも、直ちにスクリプトを作ることができ、その際に文字の打ち間違いもないので、文法エラーも発生しない。また、タイルを並べた瞬間に、計算機で実行可能な形式(スモールトークのバイトコード)への翻訳が自動的に行われるので、面倒なコンパイルやリンクを行う必要もない。このようにして作られたスクリプトは直ちに実行でき、実行中も含めて、いつでも変更できる。

これにより、思いついたことを即座に試すことができ、プログラミング自体ではなく、取り組んでいる問題の解決により多くの時間を割くことができる。

5.2. オブジェクト指向

イートイはオブジェクト指向の考え方に則って作られており、すべての処理はオブジェクトにメッセージを送ることで行われる。これらのオブジェクトには、かならず目に見える形があり、これを特に「モーフ」と呼ぶ。モーフは LOGO のタートルと似ているが、グラフィックを描くだけでなく、目的に応じて多くの種類がある。あらかじめ用意されたものには四角形や楕円、多角形などがあるが、絵(スケッチ)をモーフとして使うこともできる。自分で描いたものを動かすことで親しみが増す。

モーフが持つ属性や実行可能な操作は、ビューワというツールにタイルの一覧として表示されるので、リファレンスマニュアルは不要である。タイルスクリプトを書くときは、このタイルをドラッグ&ドロップして組み立てる。

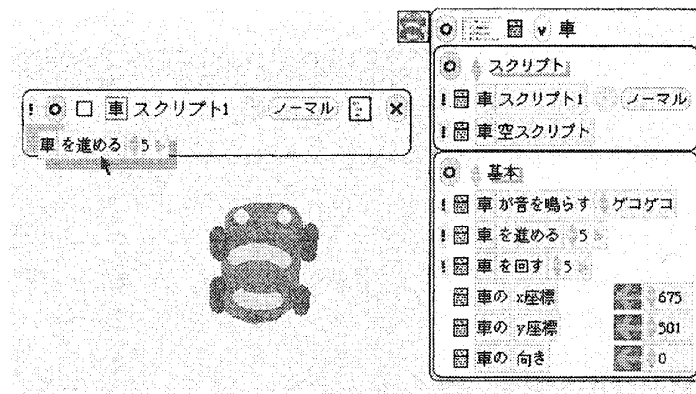


図5 モーフ、ビューワ、スクリプト

このとき、自分で作成したスクリプトもタイルとしてビューワに追加されるため、他のスクリプトで使うこともできる。

また、このようにスクリプトを追加したモーフをプロトタイプとして、その複製を作ることでもできる。その意味で、イートイはプロトタイプ(インスタンス)ベースのオブジェクト指向言語であり、クラスベースのスマールトークとは文法も環境も異なる。

5.3. 高いシミュレーション能力

個々のスクリプトは、独立に実行のタイミングを設定できる。これには、1回のみ実行、繰り返し実行、マウスイベント発生時に実行などがあり、秒あたりの繰り返し回数なども細かく決められる。

スクリプトは同時に何個でも並行して実行でき、時計と同期を取りながら動くので、物理のシミュレーションなどに向いている。また、シミュレーションの条件(モーフの属性値やスクリプトなど)は実行中でも変更できるので、試行錯誤により最適値を求めるような使い方にも適している。

5.4. 優れた拡張性

一般に、このようなビジュアルプログラミング環境に共通する問題として、最初は簡単だが、使っていくうちに、機能や拡張性の限界に達してしまうことがあげられる。通常、このようなときに環境自体を変更することは難しい。

スクイークの場合、イートイはスマールトークで記述されており、そのスマールトークもスマールトーク自体で記述されている。変更に必要なソースコードや開発ツールはすべて提供されており、自由に拡張することができる。このような柔軟性はスクイークの特徴である。

6. 科学リテラシとスクイーク

スクイークの使い方を学ぶことは、コンピューティングリテラシを習得することである。しかし、文字のつづり方を学ぶこと自体が目的ではないように、スクイークを学ぶこと自体も目的ではない。

スクイークの目的のひとつは科学リテラシの習得である。現象を観察し、仮説を立て、検証し、法則を見つけるという科学のプロセス。筆者はコンピュータ(スクイーク)を使うことで、子供たちにこのような態度を身につけさせることができると考えている。その前提として、プログラミングを自由におこなえるようになること、すなわち、コンピューティングリテラシが必要なのである。

コンピュータはどのようなメディアにもなることができるし(メタメディア)、どのような条件でも繰り返し高速にシミュレーションすることができる。これは想像力の増幅器となる。

現在、日本も含め、世界中でスクイークを用いた授業やワークショップが行われている。残念ながら、その多くはスクイーク自体を目的にしたもので、図画工作の一種として認識されている。これはスクイークのおもちゃのような外見や、最初にペイントを行うことなどから誤解されているものと思われる。

しかし、一部にはスクイークを手段として用いる動きも始まっている。たとえば、米国ロサンゼルスオープンチャータースクールのBJ コンは、落体の運動をビデオでキャプチャし、それをスクイークに取り込んで、子供たちに同じ動きをするスクリプトを作らせることで、小学生に等加速度直線運動の法則を発見させるという授業を行っている。

また、筆者らはスクイークに温度や光、電圧などのセンサをつなぐことで、現実の現象を数値化してスクイークに取り込み、理科の実験を支援する世界聴診器という装置を開発し、小学生を対象としたワークショップで用いている。たとえば、実際に果物電池を用意し、電極となる金属板と電解液となる果物の種類の組み合わせによって、どのように電圧が変化するかを調べ、その電圧をスクイーク上の車の速度に変換して競争することで、目的のある実験を行うことができる。



図5 スクイークのワークショップ

7. まとめ

パパートは難解なフランス語であっても、フランスに生まれた子供は自然に学ぶことを例に挙げ、もし数学ランドがあれば自然に数学を学べるのではないかと語っている。同様にスクイークの国、すなわち、スクイークランドがあれば、コンピュータによる科学リテラシの習得は容易になるのではないだろうか。

このスクイークランドをインターネット上に構築することで、個々の授業やワークショップで使うだけでなく、世界中の子供たちがコラボレーションできるようになるのではないかと考える。

参考文献

- 1) 阿部和広 他, 先生とパソコン, 技術評論社, 2004
- 2) Thoru Yamamoto, 阿部和広, スクイークであそぼう, 翔泳社, 2003
- 3) Squeakers, Jim Shasky, Bonnie MacBird, Ball State University, 2003
- 4) BJ Allen-Conn, Kim Rose, Powerful Ideas in the Classroom, Viewpoints Research Institute, 2003
- 5) Mark Guzdial, Kim Rose, 軋音組, Squeak 入門- 過去から来た未来のプログラミング環境, エスアイビー・アクセス, 2003
- 6) G. Korienek, T. Wrensch, D. Dechow, 菅原一孔, 鈴木元, 阿部和広, Squeak プログラミング入門- オブジェクトランドへの招待, エスアイビー・アクセス, 2004
- 7) アラン・C・ケイ, アラン・ケイ, 鶴岡雄二, 浜野保樹, アスキー, 1992
- 8) Michael Hiltzik 著, エ・ビスコム・テック・ラボ, 未来をつくった人々ゼロックス・パロアルト研究所とコンピュータエイジの黎明, 毎日コミュニケーションズ, 2001
- 9) 阿部和広, 林徹也, 世界聴診器をめぐる冒険, <http://swikis.ddo.jp/WorldStethoscope/2>, 2003
- 10) スクイークランド, <http://squeakland.jp>, 2004