

アプリケーション

Smalltalk は学術研究と商用のアプリケーションの両方において広く使われてきました。ここでは、いくつかの Smalltalk アプリケーションを紹介します。

教育 : EToys (Squeak), SqueakBot, BotsInc, Scratch...

マルチメディア : Sophie, OpenCroquet, Plopp,  Jun...

Web 開発 : Seaside, Aida, Komanche, Swazoo...

永続性の管理 : オブジェクト指向データベース (Magma, GemStone), 関連データベース (MySQL, PostgreSQL), オブジェクト関連マップ (Glorp).



OLPC で動作している Etoys と DrGeo

用語集

イメージ: Smalltalk 環境にはイメージと呼ばれる、永続的なオブジェクトの保存場所があります。イメージは、(クラスやメソッドなどの) オブジェクトとして抽象化したアプリケーションのコードと、状態を含みます。さらに、プログラムの調査やデバッグを行うために使われる開発環境も含まれています。

仮想マシン: 仮想マシンは別のプログラムを実行できるプログラムです。仮想マシンを使うことで、アプリケーションの移植性を高めるのが容易となります。

リフレクション: 言語がリフレクション能力を持つというのは、プログラムの実行中にコードを調べたり、変更したりする機構があることをいいます。

動的な型付け: 言語の中には、開発者が (整数型や文字列型などの) 変数の型を指定するものがあります。このような言語は、静的に型付けされているといわれます。動的な型付けでは、開発者は変数にどんな型の値を入れても構いません。



書籍

- たくさんのフリーの本一覧:
<http://stephane.ducasse.free.fr/Books.html>
- Smalltalk 全般
 - Smalltalk with Style
(Edward Klimas, Suzanne Skublics and David A. Thomas, free)
 - Smalltalk by Example: the Developer's Guide -
(Alec Sharp, free)
- Squeak
 - Squeak by Example - (2007, free)
 - Powerful Ideas in the Classroom
(BJ Allen-Conn and Kim Rose)
 - ¹ 自由自在 Squeak プログラミング
(梅澤真史, ISBN:4-88373-203-7, 2004)
 -  スクイークであそぼう
(Thoru Yamamoto, ISBN:4-7981-0480-9, 2003)

活動

- European Smalltalk User Group conferences (ESUG). 1993 年から活動を開始しました。企業や教育組織の Smalltalker (Smalltalk 利用者) がヨーロッパの国で会合を行なっています。
<http://www.esug.org/conferences>
- 年に1度のカンファレンスが、企業や Smalltalk 開発者の協会である STIC (<http://www.stic.st>) の主催で、北米で開かれます。
<http://www.smalltalksolutions.com/>

Internet

- Squeak 公式 website: <http://www.squeak.org>
-  日本の Smalltalk ユーザのためのハブサイト:
<http://smalltalk-users.jp>
-  日本の Squeak の拠点:
<http://www.squeakland.jp>
- Wiki: <http://wiki.squeak.org>
- ニュース: <http://news.squeak.org>

based on English Rev.82

¹  は日本語版追加情報です。

Smalltalk

純粹にオブジェクト指向のプログラミング言語に、動的な環境



Smalltalk の重要な概念

Smalltalk は、オブジェクト指向であり、動的に型付けされた言語であり、文法の主な部分は 15 分で学習可能 です。Smalltalk の強みは以下のような優れた一貫性に あります。

- 全てはオブジェクト: クラスもメソッドも数も、なに ともかも
- 規則の数はとても少なく、規則の例外はありません
- Smalltalk は、仮想マシンを使って実行します。開発は、 全ての生きたオブジェクトが含まれ、変更可能なメー シンに対して行われます。

Smalltalk の文法

予約語

nil 未定義オブジェクト (変数のデフォ

ルト値)

true と false

self

スーパークラスでの現

在のオブジェクト

thisContext 現在のメソッドの実行時ス tack

予約された文字

← (または →) 代入

↑ (または ↓) メソッドからの結果を返す

| var1 var2 var3 | 3 つの一時変数の定義

\$a 文字 a.

#(abc 123) リテラル#abc と 123 を含む配列

. (ピリオド) 式の終了

; メッセージのカスタム

[] コードブロック (モジュール)

' 文字列'

メッセージの送信

メソッド呼び出しは、メッセージを送信することによっ て行います。メッセージ送信の記法は自然言語に基づい ており、主語、動詞、補語から成ります。全てのメッセー ジは、メッセージレシーバオブジェクトに送られ、オブ ジェクトを返します。メッセージには、単項、二項、キー ワードの 3 種があります。

単項メッセージ 単項メッセージは引数を取りません。

```
array := Array new.  
array size.
```

最初の例は、new メッセージを送信し、配列 (Array) クラスの新しいインスタスを作成し、返します。2番 目の例では、この array の大きさを尋ね、0 が返ってき ます。

二項メッセージ 二項メッセージは、引数をとってだけ取 ります。メッセージ名にはシンボルが使われ、算術式な どに使われます。

```
3 + 4.  
'Hello', 'World'.
```

+メッセージがオブジェクト 3 に引数 4 で送られます。 後の例では、メッセージが文字列'Hello' に引数、 World' で送られます。

キーワードメッセージ キーワードメッセージは、引数 を 1 つ以上取ります。引数は、コロンで終わるキーワー ドの後に書きます。

```
'Smalltalk' allButFirst: 5.  
3 to: 10 by: 2.
```

最初の例は、文字列の allButFirst: メソッドを引数 5 で呼び出します。メソッドは、文字列'talk'を返しま す。2番目の例は、要素が 3, 5, 7, 9 のコレクションを 返します。

ブロック

ブロックは、直ぐには実行されないコードを含んだオブ ジェクトです。ブロックは、条件やループなどの制御構 造の基礎となります。また、メニューアイテムなどの振 る舞いを与えるためにも使うことができます。

```
#('Hello', 'World')  
do: [:string | Transcript show: string].
```

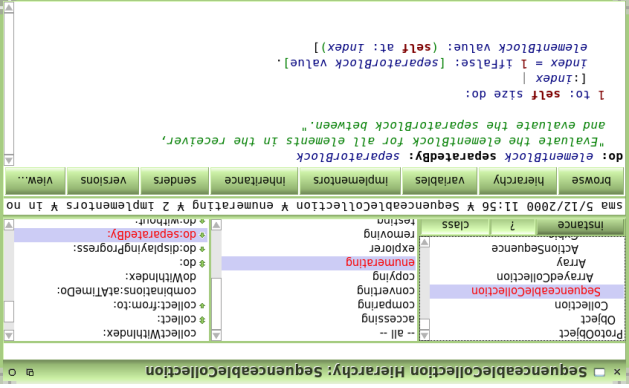
この例では、ブロックを引数として do: メッセージを文 字配列に送ります。ブロックは、配列の各要素で一度 ずつ評価されます。ブロック引数 string には各配列要 素が、次々に渡されます。結果、'Hello' と 'World' という文字列がトランスクリプトに表示されます。

開発環境

Smalltalk の多くは、ソースコードをブラウザしたり、 オブジェクトを扱ったりできる統合開発環境 (IDE) と 一緒に提供されます。Smalltalk のリフレクシヨン API を使って実装された以下のようなたくさんのツールが利 用できます。

- クラスやメソッドブラウザ
- リフレクシヨンツール
- オブジェクトインスペクタ
- リソース管理ツール
- リリース管理ツールとパーシジョン管理ツール
- テイカ
- などなど

このような環境は簡単なキーボード操作でコードの評価が でき、すぐにその結果を見ることができます。



Squeak コードブラウザ

実装

様々な Smalltalk 実装があり、利用可能です。

Squeak: フリーで、オープンソース、マルチプラットフォームな実装です。国際的なコミュニティによっ て、活発に開発されています。

VisualWorks: プロプライエタリで、マルチプラットフォーム、商用利用しない個人は自由に入手可能です。

Gemstone: プロプライエタリ実装で、とても高性能な オブジェクトデータベースを含みます。

その他: GNU Smalltalk, Smalltalk/X, SyX, VA Smalltalk, Dolphin...