

## Aplicacions

Des de la seva creació a principis dels 80s, Smalltalk ha estat extensament utilitzat tant en recerca com en el món comercial. Exemples actuals d'aplicacions Smalltalk que contribueixen a fer avançar la tecnologia del programari.

**Docència:** EToys (Squeak), SqueakBot, BotsInc, Scratch...

**Multimèdia:** Sophie, OpenCroquet, Plopp...

**Desenvolupament web:** Seaside, Aida, Komanche, Swazoo...

**Gestió de la persistència:** bases de dades orientades a objectes (Magma, GemStone), bases de dades relacionals (MySQL, PostgreSQL), correspondència entre objectes i relacions (Glorp).



Una sessió de dibuix amb Plopp

## Glossari

**Imatge:** L'entorn Smalltalk proporciona un magatzem persistent d'objectes, la imatge. Aquesta conté el codi de les aplicacions (classes i mètodes), objectes que mantenen l'estat de l'aplicació i fins i tot pot incloure eines de desenvolupament per inspeccionar i depurar un programa mentre s'executa.

**Màquina Virtual:** Una màquina virtual és un programa que és capaç d'executar altres programes. Facilita la portabilitat d'aplicacions.

**Reflexió:** Un llenguatge es diu que és reflexiu quan conté els mecanismes per inspeccionar i modificar el codi mentre s'està executant el programa corresponent.

**Tipat Dinàmic:** Alguns llenguatges forcen al desenvolupador a fer explícit el tipus de cada variable (integer, string...); això s'anomena tipat estàtic. El tipat dinàmic no imposa aquesta restricció, i per tant fa que els programes siguin més reutilitzables i fàcils de canviar.

## Llibres

- Molts llibres gratuïts:  
<http://stephane.ducasse.free.fr/FreeBooks.html>
- Smalltalk en general
  - *Smalltalk with Style* (Edward Klimas, Suzanne Skublics and David A. Thomas, gratuït)
  - *Smalltalk by Example: the Developer's Guide* – (Alec Sharp, gratuït)
- Squeak en particular
  - *Squeak by Example* – (2007, gratuït)
  - *Powerful Ideas in the Classroom* (BJ Allen-Conn and Kim Rose)

## Esdeveniments

- Les conferències de l'European Smalltalk User Group (ESUG). Des de 1993, Smalltalkers acadèmics i de sectors industrials es troben a Europa.  
<http://www.esug.org/conferences>
- Conferència anual, organitzada a Nord-Amèrica per l'STIC (<http://www.stic.st>), una associació amb presència de sectors industrials i desenvolupadors d'Smalltalk.  
<http://www.smalltalksolutions.com/>

## Internet

- Lloc web oficial d'Squeak:  
<http://www.squeak.org>
- Wiki:  
<http://wiki.squeak.org>
- Novetats:  
<http://news.squeak.org>

# Smalltalk

un llenguatge de programació  
purament orientat a objectes  
i un entorn dinàmic



## Conceptes importants d'Smalltalk

Smalltalk és un llenguatge *orientat a objectes*, amb *tipat dinàmic* i una sintaxi senzilla que es pot aprendre en *quinze minuts*. El seu principal avantatge és ser *molt consistent*:

- tot és un objecte: classes, mètodes, nombres, etc.
- un nombre petit de regles, sense excepcions!

Smalltalk s'executa sobre una *mdquina virtual*. El desenvolupament té lloc dins d'una *imatge*, en la que viuen i poden ser modificats tots els objectes.

### Sintaxi d'Smalltalk

Paraules reservades
---------------------

n1l    objecte no definit (valor per defecte)

true i false    objectes booleans

self    objecte receptor del missatge

super    objecte receptor del missatge (dins un

thisContext    pila d'execució del mètode actual

Caràcters reservats
---------------------

= (o ⇐)    assignació

~ (o ↑)    retorna un resultat des d'un mètode

| var1 var2 var3 |    declaració de tres variables temporals

\$a    caràcter a

#(abc 123)    taula (array) que conté dos literals: el

· (punt)    fi d'expressió

symbol #abc i el nombre 123

missatges en cascada

[ ]    bloc de codi (és un objecte i)

"comentari"

'cadena'

### Enviament de missatges

Un mètode és cridat enviant un missatge a un objecte, el receptor del missatge, el missatge retorna un objecte. El missatge està basat en el llenguatge natural, amb subjecte, verb i complements. Hi ha tres tipus de missatges: unari, binari i paraula clau.

**Missatges unaris.** Un missatge unari no té arguments

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

El primer exemple crea i retorna una nova instància de la classe Array, enviant el missatge new. El segon exemple demana la mida d'aquesta taula (array), i retorna 0.

**Missatges binaris.** Un missatge binari pren només un argument, el nom és un símbol i s'utilitza sovint per a expressions aritmètiques.

```
3 + 4.  
'Ho1a', ' Món'.
```

El missatge + és enviat a l'objecte 3 amb 4 de paràmetre. En el segon cas, el missatge , és enviat a la cadena 'Ho1a' amb ' Món' de paràmetre.

**Missatges de paraula clau.** Un missatge de paraula clau pot prendre un o més arguments. Els arguments s'insereixen entre cada paraula clau, després dels dos punts.

```
'Smalltalk' allButFirst: 5.
```

```
3 to: 10 by: 2.
```

El primer exemple crida el mètode allButFirst: sobre una cadena de caràcters i amb argument 5. El mètode retorna la cadena de caràcters 'talk'. El segon exemple retorna una col·lecció contenint els elements 3, 5, 7 i 9.

### Bloc

Els blocs són objectes que contenen codi que no és executat immediatament. Són la base d'estructures de control com els condicionals o les repeticions. Els blocs poden ser utilitzats per associar comportaments, ex. les opcions d'un menú.

```
#('Ho1a' ' Món')  
do: [:string | Transcript show: string].
```

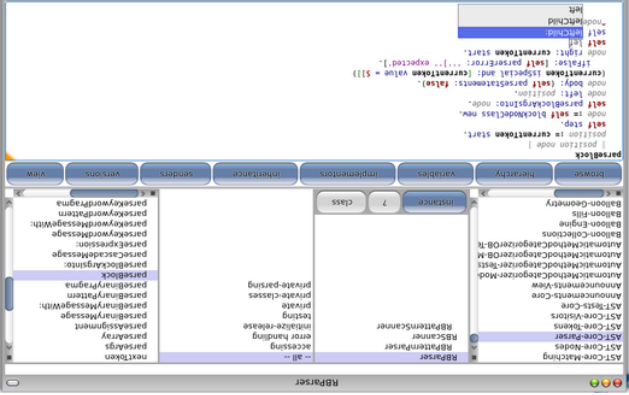
L'exemple envia el missatge do: a una taula de cadenes de caràcters amb un bloc com a paràmetre. El bloc és avaluat un cop per cada element de la taula. El paràmetre del bloc, string, conté cada element de la taula, un darrera l'altre. Com a resultat de tota l'expressió, les cadenes de caràcters 'Ho1a' i després ' Món' es mostren al Transcript.

## Entorn de desenvolupament

La majoria de les implementacions d'Smalltalk proporcionen un entorn integrat que permet explorar el codi font i interaccionar amb objectes. Aquest entorn disposa de moltes eines, totes implementades en Smalltalk gràcies a la seva API de reflexió:

- explorador de classes i mètodes;
- eines de *refactoring*;
- inspectors d'objectes;
- un depurador;
- administrador i controlador de versions;
- i molt, molt més!

El codi pot ser inspeccionat i avaluat directament dins la imatge, amb menús i senzilles combinacions de tecles.



L'explorador de codi de Pharo

## Implementacions

Existeixen diverses implementacions disponibles:

**Squeak:** gratuïta, *open-source* i multi-plataforma. Desenvolupada activament per una comunitat internacional.

**VisualWorks:** propietària i multi-plataforma, disponible gratuïtament per a ús no comercial.

**Gemstone:** propietària, inclou una base de dades d'objectes molt eficient.

**I altres:** GNU Smalltalk, Smalltalk/X, SyX, VA Smalltalk, Dolphin...