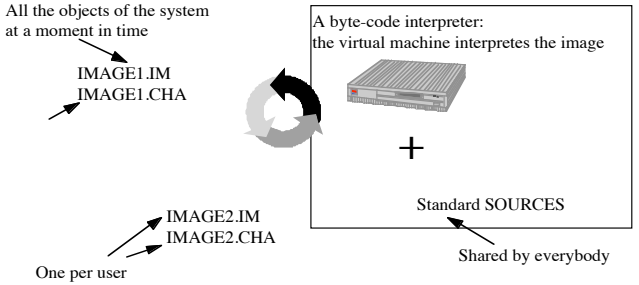
INRIA

Run-Time...

Stéphane Ducasse
stephane.ducasse@inria.fr
http://stephane.ducasse.free.fr/

Smalltalk Run-Time Architecture

- Virtual Machine + Image + Changes and Sources



Does and Dont

Do not edit the source files by hand
Do not lose your change file (it contains your code)

You cannot lose code (if you keep the change file)
Always use Smalltalk to save your code
Always use Smalltalk to rename your environment

All your changes are recorded

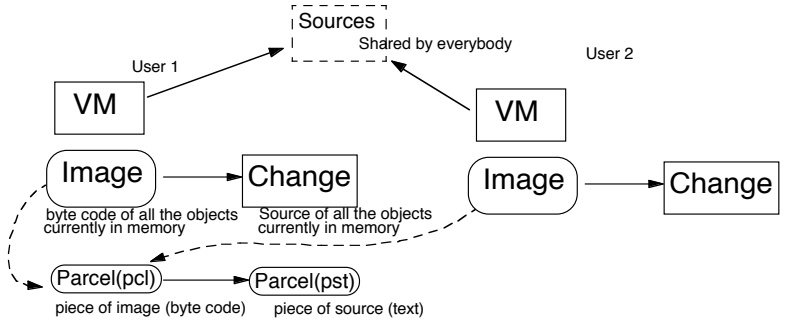
So have a lot at the changes sorter and change recovery tools suite of your dialect.

You should not lose code.

Smalltalk Run-Time Architecture

- The byte-code is in fact translated into native code by a just-in-time compiler (in VW, ST/X)
- The source and the changes are not necessary for interpreting the byte-code, this is just for the development. Normally they are removed for deployment.
- An application can be delivered as some byte-code files that will be executed with a VM. The development image is stripped to remove the unnecessary development components.

VWorks Smalltalk Run-Time Architecture



VWorks Runtime Architecture

Parcels reproduce the schema of the image and change:
*.pcl are the byte code, *.pst are the source code
Parcels allows for fast atomic loading/unloading and prerequisite parcels
Good for dynamic loading and source code management