



A graphical programming environment

Project assignment, DB330 PMI, 1989

Anna Bång D86
Magnus Hulthén D86
Per Lindberger D85
Kristina Nedlich D86

Johan Persson D86
Jan Sundberg D85
Claes Svensson D85

Idea and supervision

Johan Appelgren
Ambjörn Naeve
NADA-Matematik/KTH

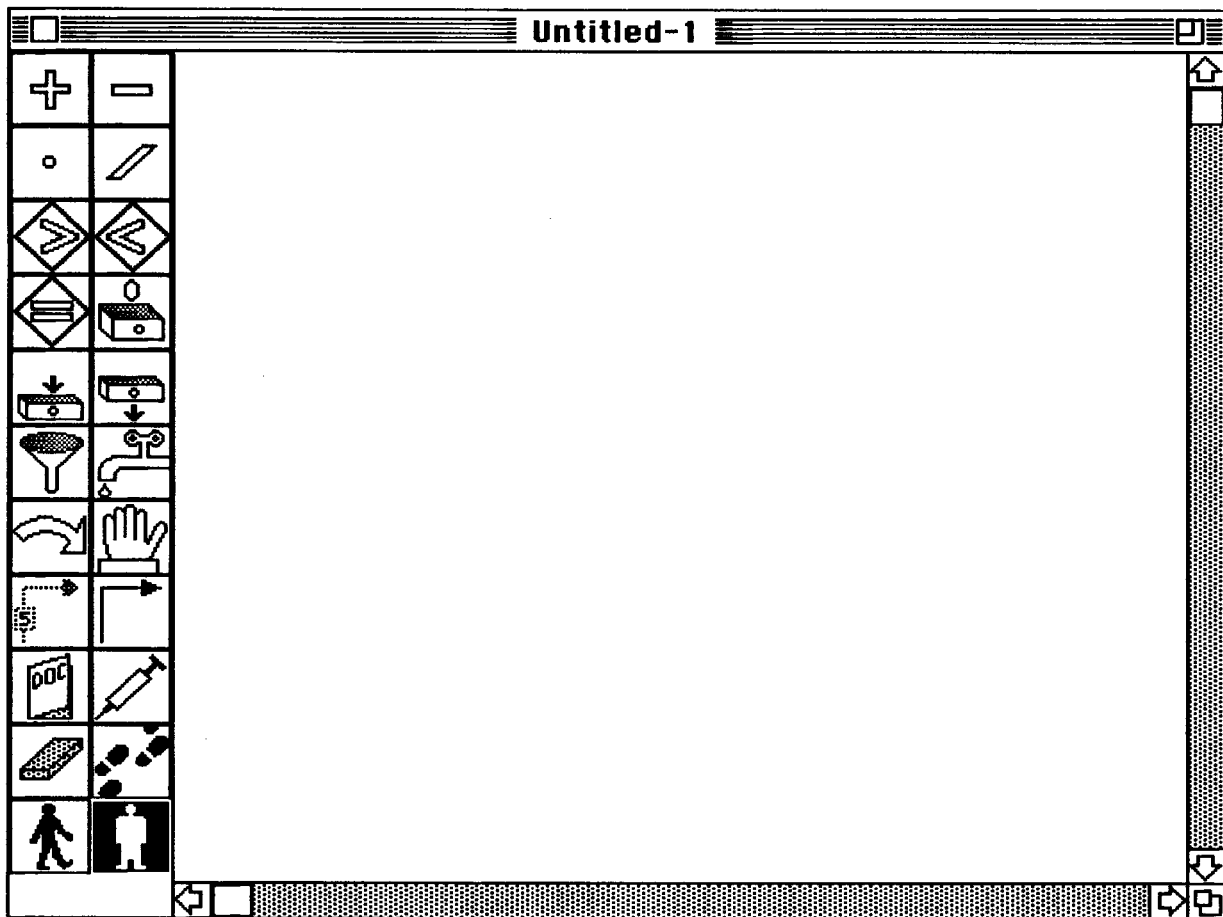
Introduction

Hyperflow is a graphical programming environment designed mainly for educational purposes. The user creates flowcharts with interconnected operators and variables. These are called procedures. The interconnections are either controlflow or dataflow. The flowcharts can then be executed.

There are a few simple operators at hand for use. These are the arithmetic operators, plus, minus, multiply, and divide. The relational operators greater, less, equal. There are two interaction operators, Input, and Output. These operators are connected via control arrows.

To provide data space there are graphically shown variables, parameters, and results. These are connected via data arrows. The variables and parameters are essentially the same, except a parameter can not be set, a result can not be read, where a variable can be both set and read.

A Procedure can be used as a "Macro" from another procedure. In this case the procedures are basically an extension of the control operators. The definition of a procedure resides in its own document



The display

The palette

arithmetic operators

which is opened by the callers document. The call creates a stack frame for the procedure which then executes as a main procedure would.

The control of a program is exerted by use of special icons in the main menu. These icons represent Start, Stop, and Single Step. There is also a possibility to set Breakpoints within a procedure.

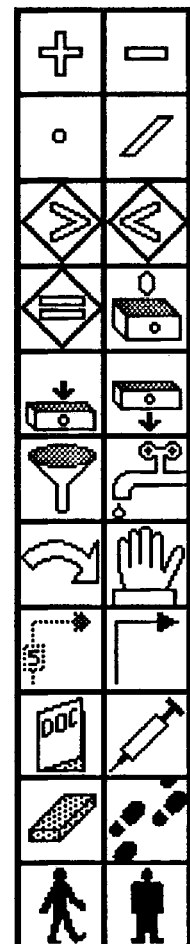
There are two kinds of documents used by the application. The Procedure document, and the Library document. The screen upon startup displays a menu-bar and an empty procedure document. (see fig 1).

The Procedure document is shown as a window with a palette on the lefthand side. This window is used to input operators, variables, and flow arrows. It is also within this window that execution is shown.

The Library document is shown with the Library window. This is an ordinary window showing procedure objects with the procedure name below the procedure object.

The palette is the "control panel" of the Hyperflow software. It is divided into two main parts. The operational part, and the control part. The operational part contains operators, variables, and parameters. The Control part contains icons for execution and control of the procedures.

The arithmetic operators have two incoming dataflows and one outgoing dataflow. The incoming dataflows are on the upper side of the operator box, and the outputting is on the lower side. The operator box has two control flows. The incoming control flow is on the center of the upper side, whereas the outputting control flow is on the lower. The operations of the operators is only executed when control has been passed to the operator.



Plus



Plus adds two incoming dataflows and outputs it on the third output dataflow. The incoming dataflows are connected to the two hollow squares on the upper sides of the operator box. The output is at the bottom right hand side of the box.

Minus



Minus subtracts one incoming dataflow from another. The dataflow that is to be subtracted from is on the upper lefthand side of the databox. The dataflow which subtracts is on the right hand side of the box. The output is on the lower right hand side.

Times



Times multiplies two incoming dataflows. The incoming dataflows are on the upper side and the output is at the bottom.

Divides



Divides divides two values from the incoming dataflows. The value which is to be divided comes on the left side of the operator box, and the value which divides is on the right side.

relational operators

The relational operators compare two values and depending on the specific operator will pass control to either of two control flows. The relational operator has the shape of a rhombus. The input values to the operator are on the upper left side, and the upper right side of the square. The control flow comes in on the top peak of the square and exits on either the bottom peak, or on the right side peak.~

Greater



Greater compares values. If the value on the right side is greater than the value on the left side control will be passed to the right peak. If not control will be passed to the bottom of the operator rhombus.

Less



Less compares values. If the value on the right side is less than the value on the left side control will be passed to the right peak. If not control will be passed to the bottom of the operator rhombus.

Equal



Equal compares values. If the values are equal control will be passed to the right peak of the operator square. If the values are different control will be passed to the bottom.

variables and parameters

Variables



Parameters



Procedure results



Input/Output

Input



Output



Break point



Variables and parameters are purely data specific. They have no control flows. There are three different kinds; Variables, parameters, and procedure results. The shape of the variable box is a square.

Variables have an incoming dataflow on the top of the square, and output on the bottom. All data arriving on the input will be immediately stored internally and sent to the bottom.

Parameters are set by the calling procedure and can only be read by the procedure in which they exist. The outputting dataflow is on the bottom of the parameter box.

Procedure results can only be set by the procedure in which they exist. The data will be sent to the calling procedure. The connection for the incoming dataflow is on the top side of the result box.

Input and Output are operators which are used for interaction by the procedures. They have a control input and output on the bottom and the top of the operator box.

The input box will open a dialog when control has been passed to it. The dialog will request a number. The number will be outputted on the dataflow on the bottom right hand of the operator box.

The Output box has a dataflow input on the upper left side of the operator box. When control has been passed to the output box a dialog will open showing the value on the input of the box.

The break point is a normal operator, but has no dataflow connections. It is placed in a program where a temporary stop of a procedure execution is wanted. Normal controlflow arrows should be connected to the break point.

ARROWS

Control arrow



Data arrow



Command icons

Start



Stop



Single Step



Injector



Eraser



The arrow are used to interconnect the variables and operators into procedures. There are two kind of arrows; The control arrow, and the data arrow. The arrow are drawn by pressing the mouse button on the start point and the dragged with the button still pressed to the destination position. The arrow can be segmented with several vertices.

The control arrow is the solidly drawn arrow. It is connected at the filled squares on the outline of the boxes.

The data arrow is the gray dithered arrow. It is connected to the hollow squares on the outline of the boxes.

The command icons are not operators which can be used in a procedures but are operators to control execution and data in the procedures.

Starts execution of a procedure. The execution will stop when the end of the procedure, or a breakpoint is encountered. The execution can also be stopped by the Stop command.

Stops the execution. This requires that the procedure is executing.

Single steps through a program. The executed operator is highlighted. When the mouse is clicked at that operator the control is passed to the next operator.

Injector is used to set values in variables. When the injector cursor is clicked over a variable a dialog will open. The dialog will request a value for the variable.

This command is used to erase operators, variables, and arrows. When the eraser icon is chosen an eraser cursor will be shown. This cursor is used to point out the items to be erased. When an operator or variable has been chosen to be erased, all connecting arrows will be erased with it.

Open



The command icon open is used to open documents. When this icon is selected the cursor will be changed to the open document cursor. This cursor is then used to click on procedureboxes within the procedure. This will open the procedure document and show a window containing the procedure.

Menubar

There are three menus in the menubar of the application. They are the File menu, the Edit menu, and the Qwe menu. They work as normal Macintosh application menus work.

File menu

The file menu is used for operations on documents. You can create, save, and open documents. You can also print documents. The menu is also used to exit the application.

File	Edit	Qwe	Debug
New CodeDocument			⌘N
New Library			
Open CodeDocument			⌘O
Open Library			⌘L

Close			⌘W
Save			⌘S
Save As...			
Save a Copy In...			
Add to Library...			⌘R
Revert			⌘R

Page Setup...			
Print One			⌘P
Print...			

Quit			⌘Q

New CodeDocument ⌘N

This menu choice is used to create a new and empty code document, or procedure. It will open an empty window with the title **Untitled-1**. The number increases for every time a new code document is generated.

New Library

This menu choice is used to create a new and empty library. A window with the name **Untitled** will open and place itself on top of the desktop.

Open CodeDocument ⌘O

This menu choice is used to open an existing code document. When the choice is executed a dialog requesting for a file will be shown. When a file has been selected a new window will present itself on the top of the desktop. Inside the window a procedure is shown.

Open Library ⌘L

This menu choice will open an existing library document. A dialog will present itself asking for a library document. When a library document has been given a new window will show itself with procedures inside the window.

Close **⌘W**

This used to close documents. Both codedocuments and libraries are closed by this menu choice. When a document is closed the window of the document will disappear from the desktop. If the document has been changed while it was opened without being saved a dialog will open asking whether it should be saved or not.

Save **⌘S**

This menu choice is used to save documents. When this command is executed the document will be stored with the changes from where it was taken. If the document has not been changed the menu choice will be gray and can't be selected.

Save As...

This menu choice is used to save a document under a new name. When it is chosen a dialog, asking for a new name of the document, is opened. When a name has been given the dialog will close and the document will be saved under the new name.

Save a Copy In...

This is a menu choice of which I know nothing whatsoever. Therefore I will just type some text to fill up thisd area until I know what it does and give a real explanation and thereby removing this junk which is just reserving space for the real text.

Add to Library **⌘A**

This menu choice is used to place a reference to a procedure in a library. When it is chosen a dialog, asking for a library to place the reference in, is opened. When the library has been given a reverence is saved into a library. If the library is opened and visible the reference will also be seen in that window.

Revert **⌘R**

This is another menu choice which I don't know anything about . But I wont type as much this time.

Page Setup...

This menu choice is used to set up the printer for output. A dialog with buttons and choices will be opened when the choice has been selected. This is a system dialog and the same as used by other applications.

Print One %P

This menu choice prints a document. It will only print the active document on top of the desktop.

Print...

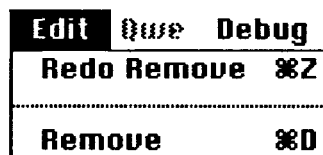
This menu choice is used to print all opened documents. The output is governed by the page setup. Which is a menu choice above.

Quit %Q

This menu choice is used to leave the application. All documents will be closed prior to exit. If there are any documents which have been changed without being saved a dialog, asking whether they should be saved or not, is opened for all documents.

Edit menu

The Edit menu is used for removing items and also has a Undo function.



Redo Remove %Z

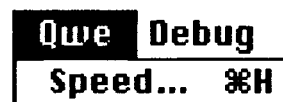
This menu choice is used to undo the most recent command. If the most recent command was an undo it will redo the command.

Remove %D

This menu choice is used to remove procedures selected in the library window.

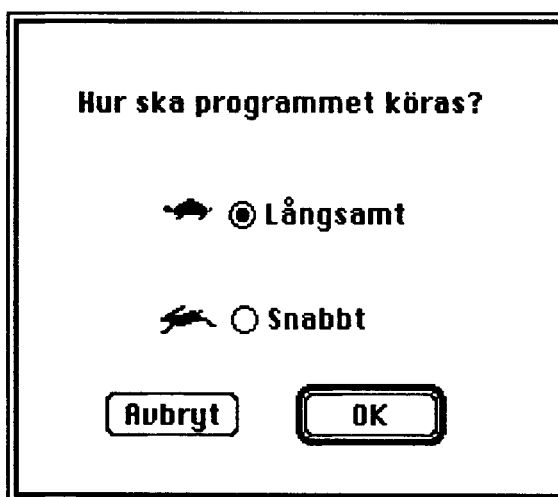
Qwe menu

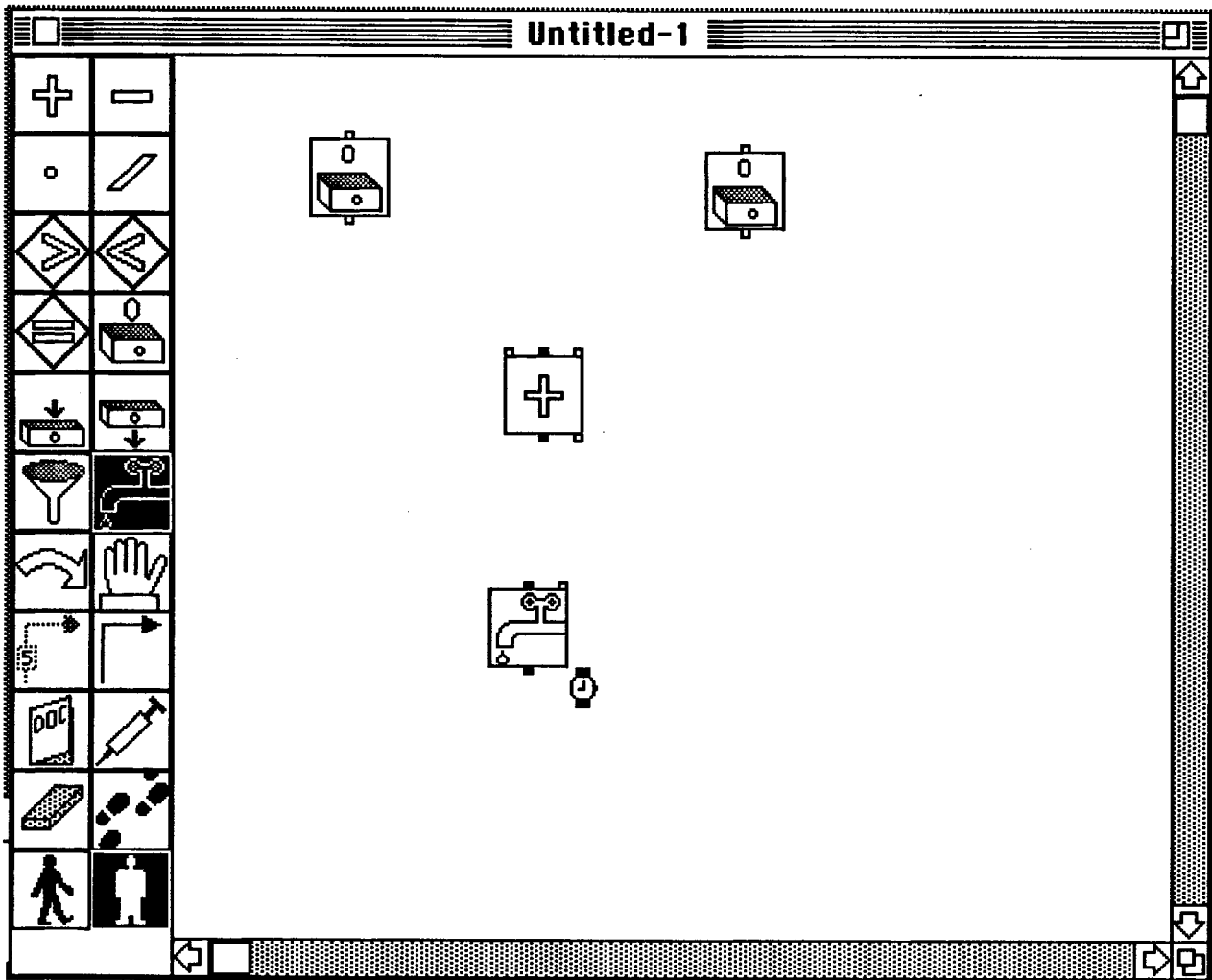
The Qwe menu is used for miscellaneous operations. At present there is only one command.



Speed... %H

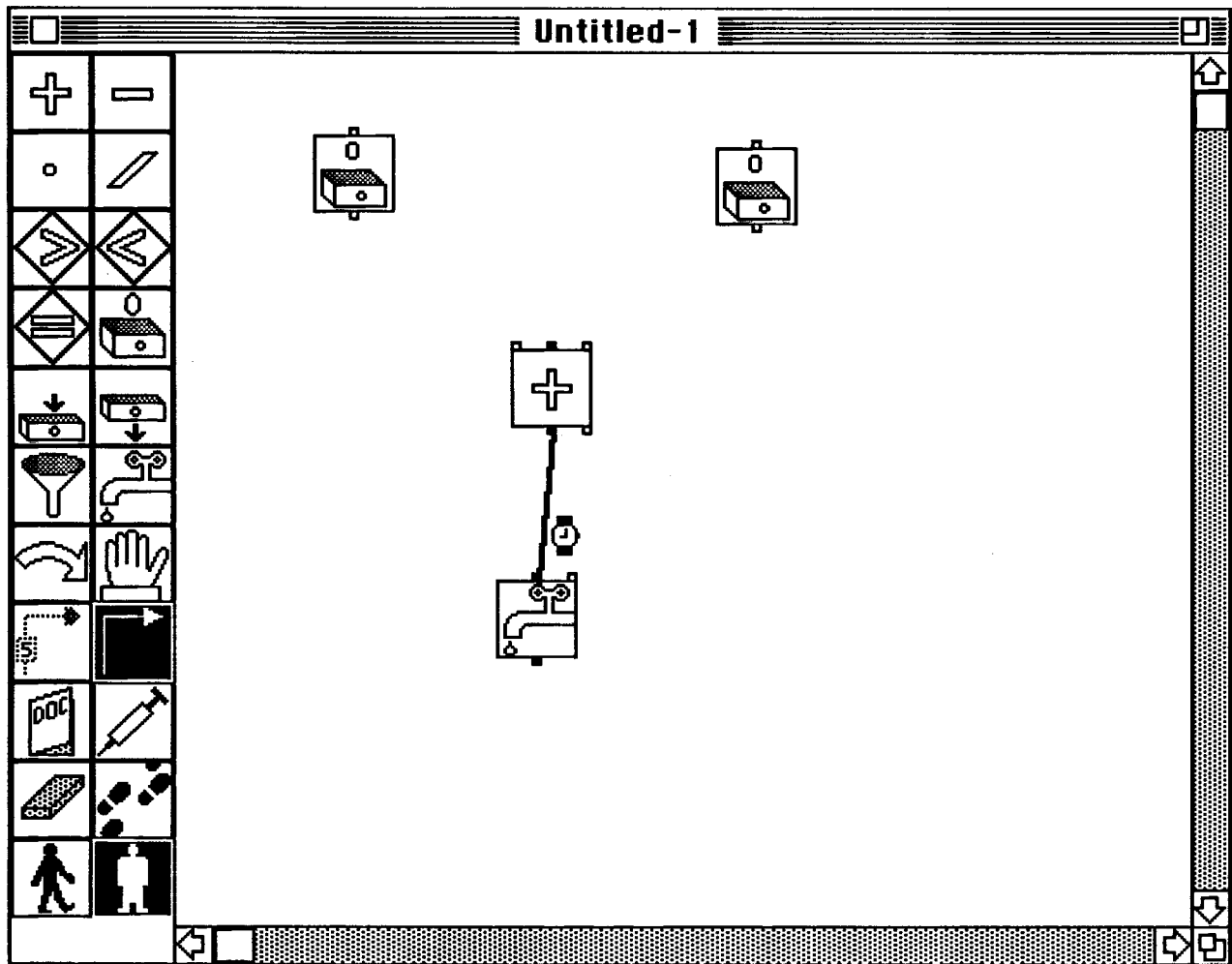
The speed menu choice will open a dialog (seen below). Asking which speed of execution is wanted. The turtle symbolizes slow, and the hare symbolizes fast.





Procedures

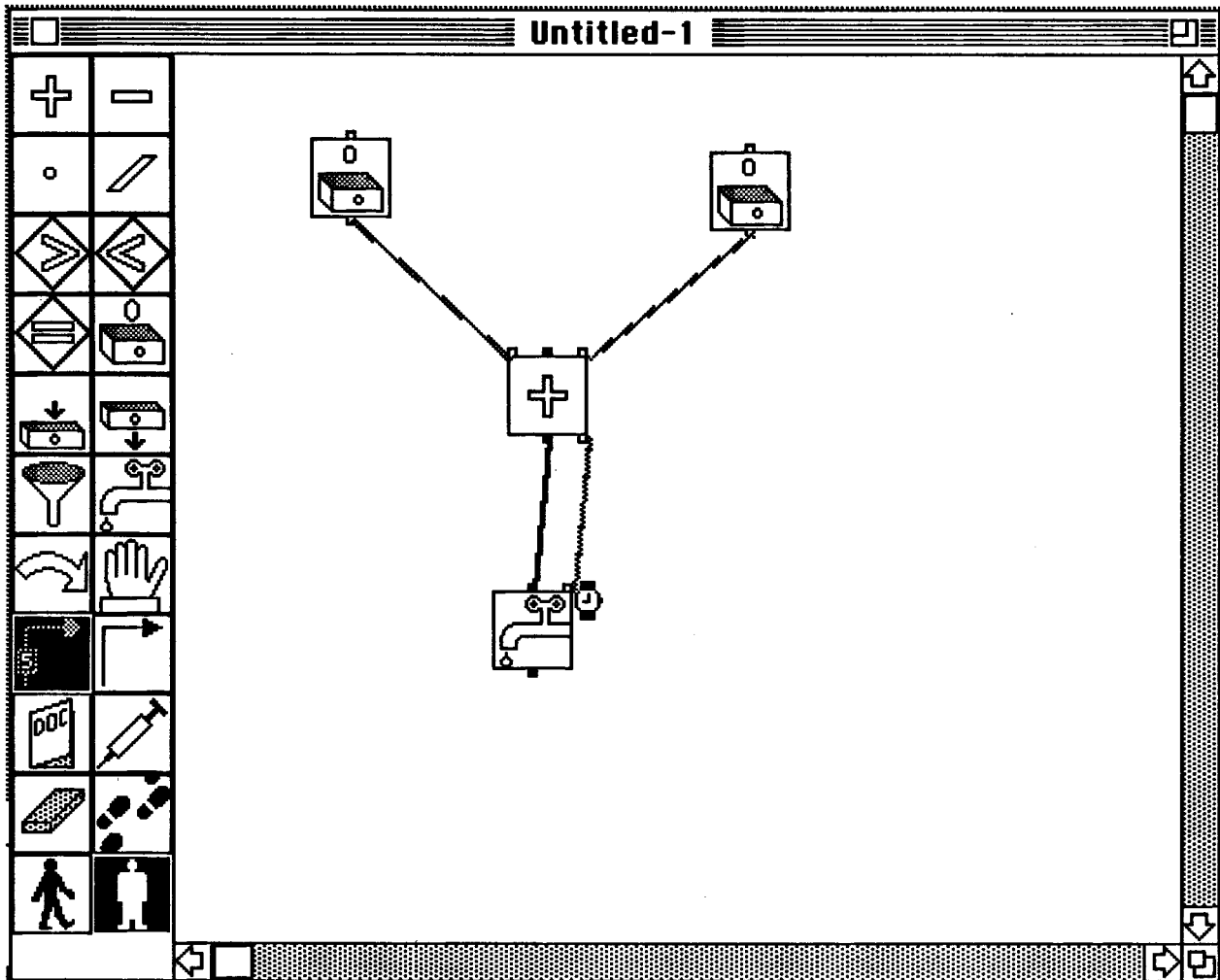
The user can create programs by using operators and variables. An operator or a variable can be selected by clicking the mouse over one of the items in the palette. The mouse cursor will then change shape to resemble the selected item. When the mouse is clicked at a position, an operator or variable will be created at that position.



When all operators and variables are created a control arrow can be selected. It is the solid arrow. This is used to interconnect the control flow of the operators. The arrow is started from a filled square of the operator. This represents a connection. It is then routed to a filled square on a receiving operator. It is possible to use several vertices as midpoints for the arrow.

The data flow arrows are the routed to the hollow squares on the operators and the variables.

If an operator or arrow is unwanted, it is possible to delete it with the eraser. All connections to an operator which is erased will also be deleted.



Execution

The execution can be performed by either a straight execution, or single step execution. The straight execution is performed by selecting the command icon Start. The single step execution is started by pressing the single step command icon.

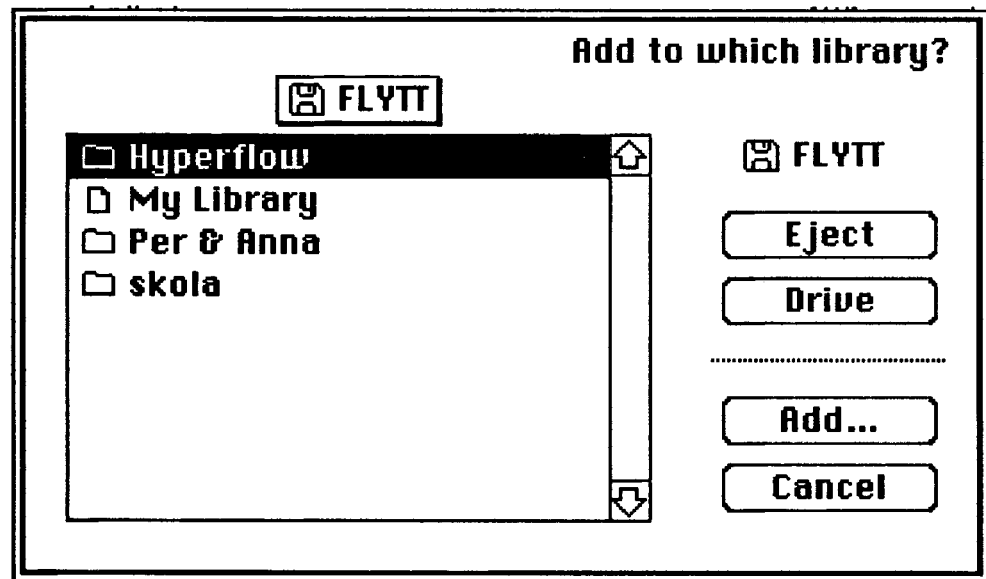
The execution can be stopped in three fashions. Either by a normal procedure termination when execution control can't be passed to any operator. It can also be stopped by a breakpoint in the procedure. When a breakpoint has been encountered, normal execution can be resumed at this point by restarting with the start command, or the single step command. The execution can also be stopped by selecting the stop command. Execution can likewise be resumed from the position where it was stopped.

The execution is performed by passing control first to the operator which has no control input at the top of the box. I.e. the execution start is implicit

in this case. If no such operator can be found, an alert will be shown. When an operator has been executed control will be passed to the procedure connected by a control arrow at the bottom of the current operator.

Procedures

A procedure which has been written can be saved and treated as a unit. The procedure can then be used as if it was an operator. Recursive calls are allowed. The procedure can also be connected with dataflows. This is accomplished by using procedure parameters and results. The number of parameters on the input and the representation of the procedures data flow connections icon is automatic.



Libraries

The icon representation of a procedure can be saved in libraries. This is done by selecting the command Add to library... in the File command. A dialog as seen above will be opened. The library is a separate kind of document. The information in the library is NOT that of the operators and connections of a procedure but rather a reference to a file containing this information. The first thing that happens when a procedure is called is that the data in the host procedure is saved on a stack. The arguments are transferred to the called procedure by use of the procedure parameter boxes. This means that the environment in which a procedure is executed is always local. It also implies that recursion is allowed.