An interface for the interactive design of artistic screens

N. Rudaz, R. D. Hersch, V. Ostromoukhov

Ecole Polytechnique Fédérale, Lausanne, Switzerland

Abstract. This work presents the concepts and the tools involved in the interactive design of artistic screens. The screen elements are derived from a small set of analytical contours provided by the screen designer. We present the requirements that these contours must satisfy in order to generate consistent screens. Software tools have been developed which provide automatic means for verifying and enforcing these constraints. They include a way of specifying the periodicity of the screen dot and a graphical interface offering a convenient way of specifying and tuning the growth of the screen dot.

1 Introduction

1.1 What is artistic screening

Artistic screening [1] is a new image reproduction technique incorporating freely created screen elements shapes for generating halftones. Artistic screening enables the shape of the screen dots to be tuned according to the designer's will. Freely shaped screen dots can be used to reproduce artistically screened images that incorporate two levels of information, one carried by the image itself and the other carried by the screen shapes. The new information layer represented by the screen shapes may be conceived as an aesthetic enhancement of a picture, or it can also be used to forward a message.

1.2 The artistic screening process

The process for creating artistically screened images consists of three main steps. In the first step, the contours of the screen dot must be manually designed for a few typical, key intensity levels. Such key contours are specified as analytical Bézier splines (Fig. 1). This step is usually performed by a graphic designer using an outline drawing software.

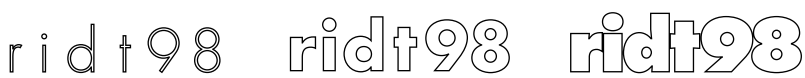

Fig. 1. The contours of an artistic screen dot at high, medium and low intensity levels.

In the second step, the contours of the screen dot for all intensity levels are automatically interpolated between the key contours specified at the first step. These interpolated contours are then rasterized into a collection of bitmaps forming the elements of an artistic screen. Some of the bitmaps derived from the key contours in Fig. 1 are displayed in Fig. 2.

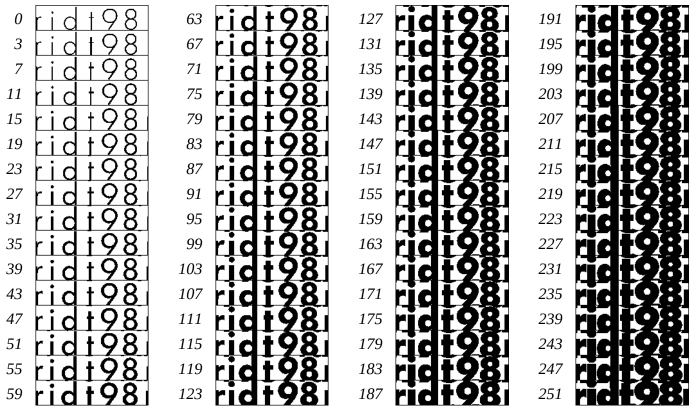

Fig. 2. A subset of the screen elements obtained through the interpolation and the rasterization of the contours described in Fig. 1.

The third and last step involves the production of an artistically screened image by using the discrete screen elements created at the previous step to produce a gray-level image. In this step, the intensity value of a pixel in the original image is used as an index for selecting the corresponding element of the artistic screen [1]. The exact position of a screen element cell for a given output image pixel is obtained by calculating the coordinates of that pixel, modulo the dimensions of the screen element. An example of a simple reference image reproduced with the artistic screen generated from the contours outlined in Fig. 1 is shown below in Fig. 3.

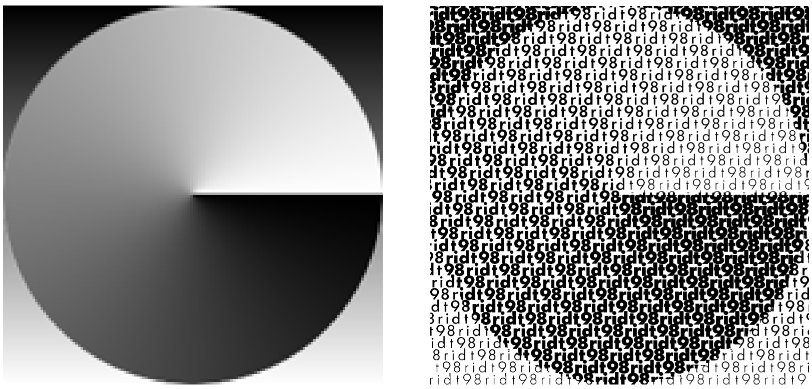

Fig. 3. A grayscale reference image, and the halftoned image obtained by using the artistic screen described in Fig. 1 and Fig. 2.

1.3 Artistic screening requirements

The fact that the shape of the screen element is created according to the designer's choice does not mean that it may incorporate arbitrary contour descriptions. Designing artistic screens involves numerous constraints and requirements. Therefore, a set of software tools providing a consistent user interface [2] has been created in order to facilitate the creation and the management of artistic screens. These tools include the following features, which are presented in this paper:

- interactive design of Bézier splines (paths)
- checking and enforcing the coherence between groups of paths (shapes)
- specifying the periodicity and dimensions of screen dots
- aligning the key shapes inside the cell tile
- specifying the growth rate of the screen dots
- associating intensity levels to key shapes

2 The screen design software

2.1 Interactive design of Bézier splines

Designing the contours of the key screen elements is the part of the process that requires the most intervention from the user. Since these contours are described by closed Bézier splines, it makes sense to use an interactive Bézier spline editor. In order to avoid developing a new software package, it is preferable to use an existing, extensible application. The functionalities and the open, well-documented plug-in architecture of Adobe's Illustrator™ ([3], [4]) make this application a good choice. In this perspective, the key screen shapes consist of groups of closed Bézier splines, referred to as paths. Individual paths are designed and grouped into shapes in Illustrator. Additional information specific to the screen generation process is provided interactively by the screen designer, making use of several Illustrator plug-ins.

2.2 Constraints on the key screen elements

Screen element contours are interpolated for each intensity level between a set of user-defined key contours, or shapes. For this application, interpolation for a given range of intensity levels requires key shapes that are specified in pairs. Each pair of shapes define the initial and final appearance of the screen element at a given stage of its evolution. Thus, both members of a given pair must have the same number of paths. It is therefore sometimes necessary to artificially change the number of paths in some shapes in order to satisfy this constraint. This is illustrated in Fig. 4, where digit 7 of the first shape is morphed into digit 9 of the second shape. The left pair of shapes is not coherent, because path 9.2 in the bottom shape cannot be assigned a corresponding path in the top shape. This situation has been corrected in the right pair by removing path 9.2 and by altering path 9.1 in order to preserve the central hole of digit 9.

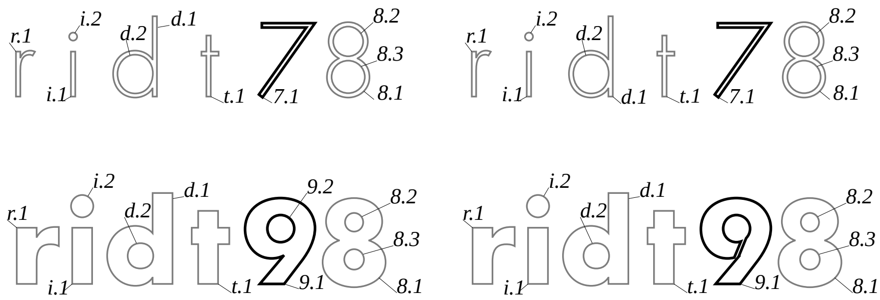

Fig. 4. Not all paths of shapes in the left example can be matched; the right example shows a possible correction.

The condition that each member of an interpolation pair must include the same number of paths is necessary but not sufficient. The intermediate shapes are obtained through a linear interpolation of the individual control points of each path. Therefore, there should be a one-to-one correspondence based on the number of anchor points between the paths of each member of the pair. For instance, if one shape includes two paths with five anchor points, then the other shape must also have two paths with five anchor points. It is therefore sometimes necessary to add one or more control points to the paths forming a shape (Fig. 5).

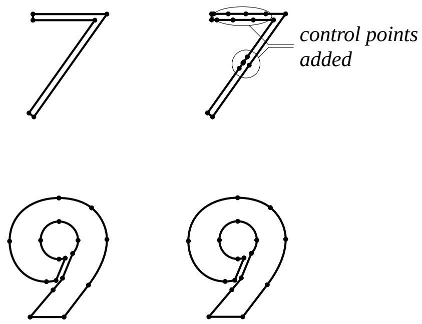

Fig. 5. Equalizing the number of control points in two corresponding paths.

The Illustrator plug-in used to generate artistic screens automatically checks the coherence of the key shapes. It also tries to match corresponding individual paths whenever possible. Sometimes, the user is required to manually specify the exact correspondence. This is done through a dialog that enables selecting one path in a shape and visualizing the corresponding path in the associated shape, making it possible to change the paths correspondence (Fig. 6).

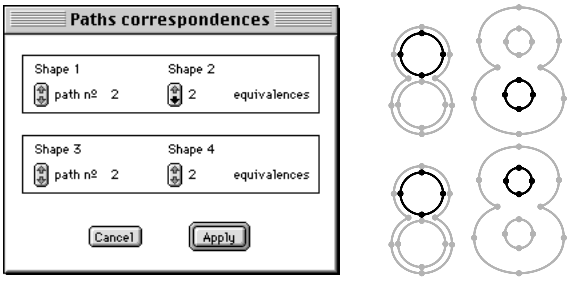

Fig. 6. The paths correspondence in the top shapes is wrong. The correct correspondence is shown in the bottom shapes.

A correct ordering of the control points is required in order to ensure a consistent interpolation. As Illustrator does not allow the user to specify the order of the anchor points inside a path, an additional plug-in implements this functionality. It enables the user to specify the start of a path by simply clicking on a selected control point. By using a modifier key, it is also possible to toggle the orientation of the path.

There is also an option for automatically assigning correspondences between paths based on the position of their bounding boxes. This option was specifically designed for large screen tiles made of many characters, and it uses common properties of character shapes as well as the character positions. Paths are first sorted according to their relative positions, i.e. from left to right and from top to bottom. Within individual characters, disjoint paths are associated according to the relative position of their bounding boxes.

The automatic paths association has been used to generate the artistic screen with which the image depicted in Fig. 7 was reproduced. The screen tile has the same size as the image. Each key screen element is composed of more than ten thousand paths. Manually tuning their correspondences would have been an impossible task.

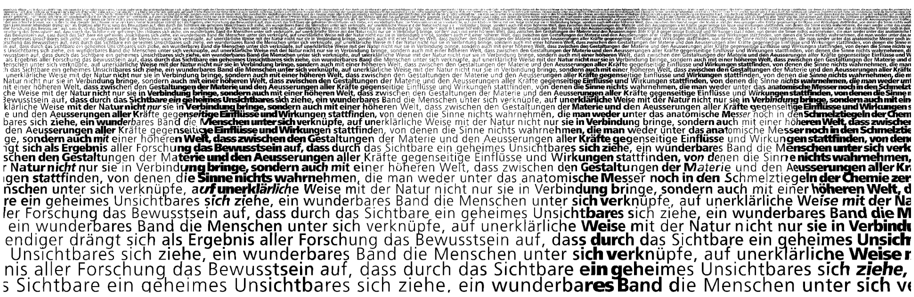

Fig. 7. A complex character-based artistic screen, consisting of a single screen element.

2.3 Periodicity and dimensions of the screen dots

Whether the screen dots tile the plane or not, it's their repetition at regularly spaced, periodic intervals that gives a screen its unique aspect. The same element can produce a wide variety of screens just by modifying its repetition period and orientation. Specifying these two attributes is thus a mandatory part of the screen design process. Two variations of screen period and orientation are illustrated in Fig. 8. The rectangular cell used for the left example yields a screen showing very apparent artifacts when viewed from far enough. The white spaces between the words are vertically aligned and therefore give the impression of alternated black and white vertical lines. And because the height of the cell is too big, the spaces between the lines are apparent and produce an alternance of black and white horizontal lines. These artifacts are much less visible on the right screen, whose screen tile is a parallelogram and whose height has been reduced.

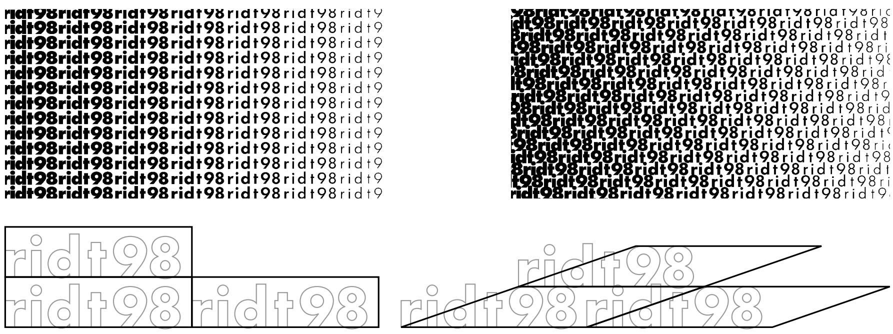

Fig. 8. The influence of the periodic cell on the aspect of the screen.

For any planar repetitive pattern, its is sufficient to define a parallelogram of appropriate dimensions [5]. The sides of the parallelogram define the repetition vectors (Fig. 9).

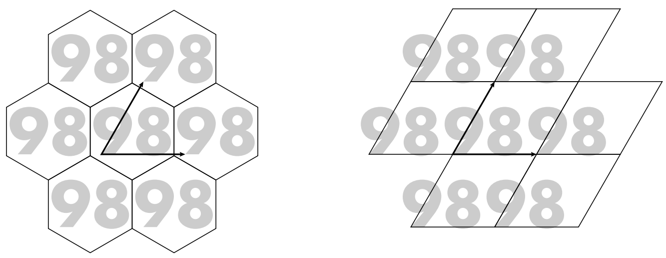

Fig. 9. Equivalent hexagonal and parallelogram periodic cells.

Presently, the user must enter manually the coordinates of the period vectors into a dialog box, together with an optional scaling factor (Fig. 10, left). A fully graphical interface for specifying the periodic cell is currently under development and is illustrated in the right part of Fig. 10. It takes the form of a dialog showing three copies of a key shape, along with the period vectors given by their relative positions. Two of these copies can be freely moved until a satisfying screen tile has been created.

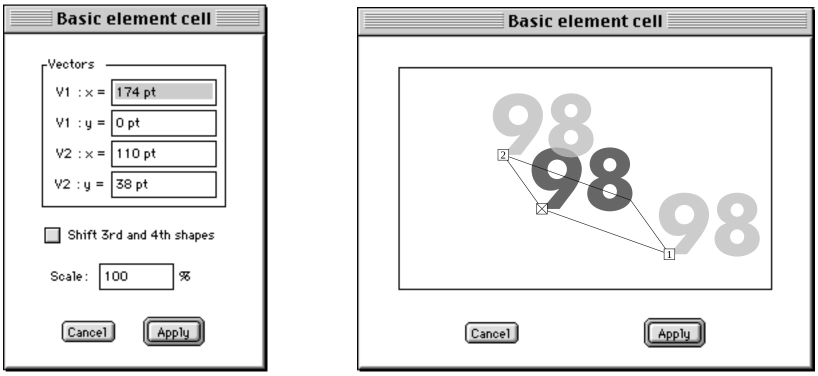

Fig. 10. Entering the dimensions of the screen tile and scaling the screen. At left, the actual interface; at right, the planned, final interface.

2.4 Position of the key shapes inside the cell tile

Before the contours of the intermediate screen elements are interpolated, it is necessary to position the key shapes relatively to the screen tile. By default, the key shapes are aligned with the bottom left corner of the screen tile, but it is possible to specify another location. If different locations are specified for two key shapes, the resulting intermediate screen elements will appear to be progressively displaced inside the screen tile.

A location different from the default one can be specified by selecting a path whose bottom left corner will be used as a reference point for the alignment of the screen tile. This reference path can be independently selected for every key shape through the interactive dialog illustrated in Fig. 11.

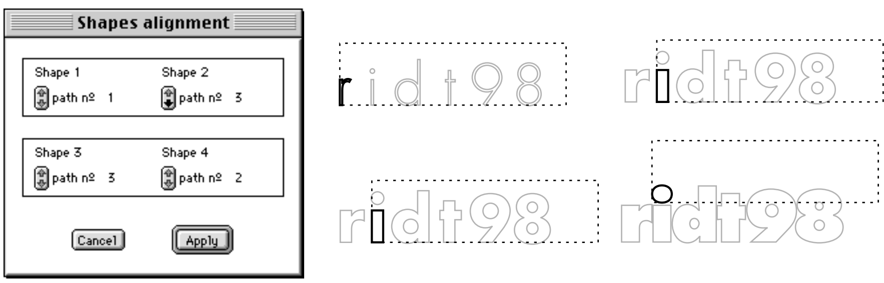

Fig. 11. Selecting different reference paths for different key shapes modifies their positions inside the screen tile.

2.5 Growth rate of the screen dots

Interpolating between key elements is not always sufficient to describe the evolution of the screen elements. The interpolated elements must sometimes be scaled up or down, and the scaling factor for a given screen element may be a function of its associated intensity level. The control provided for specifying the growth of the screen elements takes the form of a dialog including a curve that can be freely manipulated by the user, as illustrated in Fig. 12. The default curve is a straight line, and it specifies a linear growth of the area of the screen elements as a function of intensity.

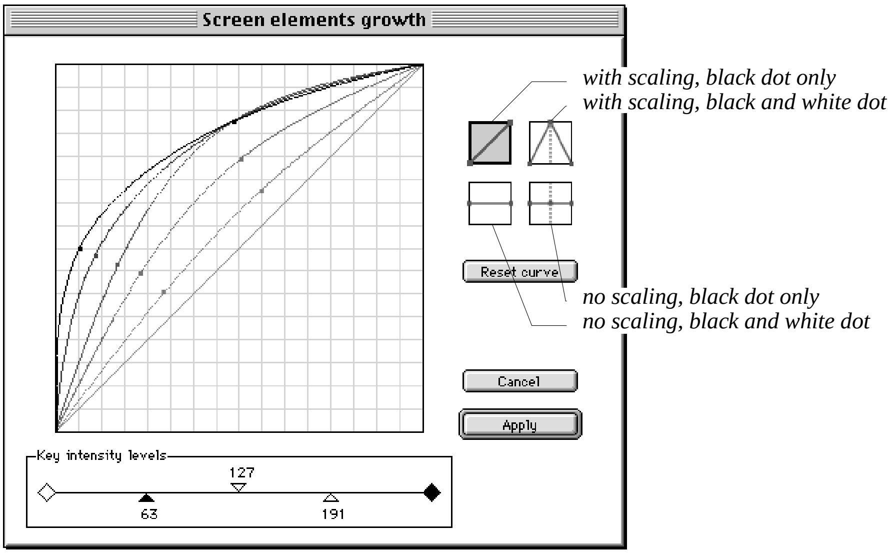

Fig. 12. Specifying the type and growth of the screen elements.

Together with the growth curve, information about the evolution of the elements may be provided by the dialog box of Fig. 12. The behavior of the screen must be selected from four possible options, which result from the combination of two independent criteria. The first criterion establishes a distinction between screen elements whose size is entirely specified by the associated shapes, and screen elements that need an additional scaling operation in order to reach their final size. The second criterion differentiates between screens which grow as a black dot over the whole intensity range, and screens which first grow as a black dot and then shrink as a complementary white dot. In the latter case, at a specific intensity, the background of the screen shifts from white to black. In that case, key shapes for the white dot which are the geometrical complement of the key shapes of the black dot need to be designed.

An example of each of the four growth types resulting from the combination of these two criteria is shown in Fig. 13.

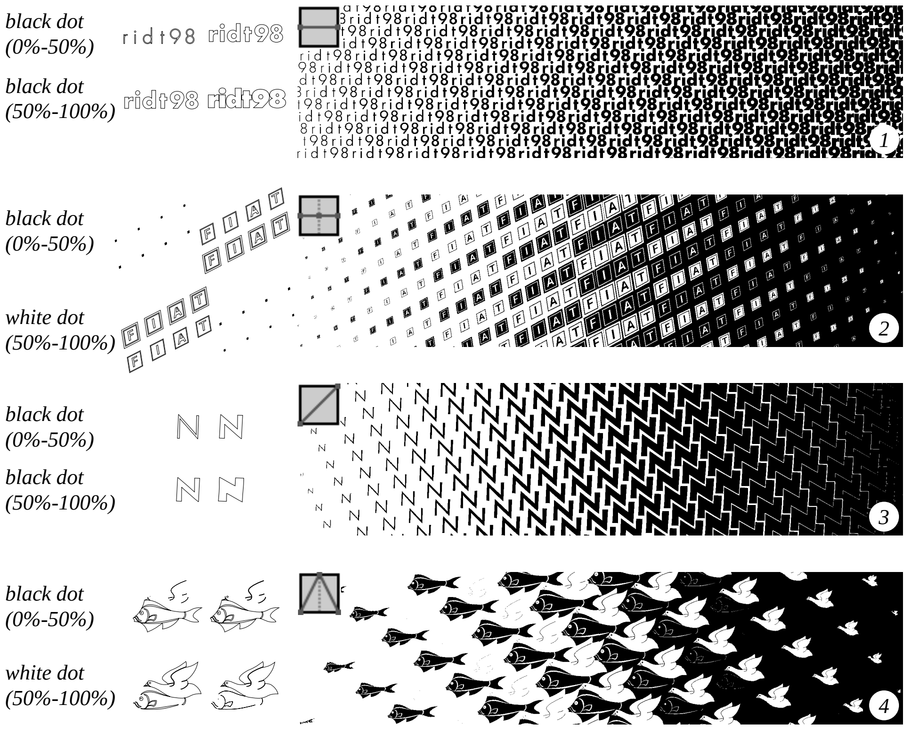

Fig. 13. The four possible growth types. (1) Without scaling, without background inversion. (2) Without scaling, with background inversion. (3) With scaling, without background inversion. (4) With scaling, with background inversion.

2.6 Key screen elements and key intensity levels

In addition to the growth curve, the dialog shown in Fig. 12 enables associating given intensity levels to the key shapes. The associated levels are materialized by small triangular cursors that can be moved along a horizontal line. Their movement is limited by the positions of the surrounding cursors, as shown in Fig. 14.

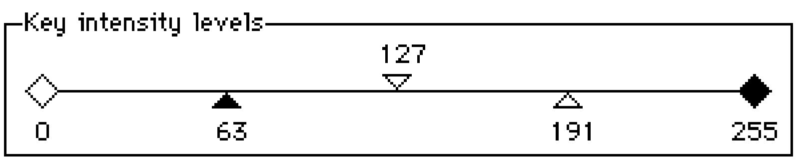

Fig. 14. Adjusting the association between key shapes and intensity levels.

3 Conclusion

The interface elements presented in this paper enable the user to create complex screens which respect the constraints of the artistic screening technique. The key outlines of the screen elements are interactively designed using Adobe Illustrator. The additional functionalities required by artistic screening are provided in the form of Illustrator plug-in modules. The task of producing coherent groups of paths has been automated for screens based on characters and has been made easier in all other cases. A general method for specifying the periodicity of the screen dot has been outlined, as well as the corresponding interface. The growth type of the screen dot specifies whether to scale the screen dot or not, and whether only black dots are considered over the whole intensity range or a succession of black and white dots. By using a visual tool, the screen dot's growth curve as a function of intensity may be freely adjusted.

The principles and the interface tools presented in this paper were developed and refined with the help of graphic professionals. The resulting artistic screening software package is a unique tool enabling the creation of an entirely new category of images. Such images fit well in the trend of rendering pictures by non-photorealistic means, like the pen-and-ink technique [6]. Artistic screening is used in the design of posters, product packages and letterheads. Future work on the interactive design of artistic screens will involve further automation of the design process, the extension of artistic screening to color images, and a tighter integration with existing image processing software.

References

1. V. Ostromoukhov, R. D. Hersch, "Artistic Screening", SIGGRAPH'95, Computer Graphics Proceedings, Annual Conference Series, 1995, pp. 219-228.
2. B. Shneiderman, "Designing the User Interface", Addison-Wesley, 1992.
3. M. Foster, "Adobe Illustrator 6.0 Software Development Kit", 1995.
4. G. Ishii, "Adobe Illustrator 7.0 Software Development Kit",[1] 1997.
5. B. Grünbaum, G. C. Shephard, Tilings and Patterns, an Introduction, W. H. Freeman and company, New York, 1989.
6. M. P. Salisbury, S. E. Anderson, R. Barzel, and D. H. Salesin, "Interactive Pen-and-Ink Illustration", Proceedings of SIGGRAPH94, in Computer Graphics Proceedings, Annual Conference Series, pp. 101-108, 1994.

1. ftp://ftp.adobe.com//pub/adobe/devrelations/sdk/illustrator/mac/Illustrator70SDKr1MacInst.hqx