



Grahame Weinbren and Roberta Friedman.
The Erl King (1982–85). 1985 version, shown at
the Whitney Museum of American Art, New York.
Collection of the artist, image courtesy of the artist

Grahame Weinbren et Roberta Friedman,
The Erl King (1982–85). Version 1985, présentée au
Whitney Museum of American Art, New York.
Collection de l'artiste, image gracieuseté de l'artiste

SOURCES: reproduced (three laserdiscs); encoded
(Pascal program with CP/M operating system)
MATERIAL: duplicated (wood and metal construction)
DISPLAY: duplicated (two monitors)
PRESENTATION: installed (approx. 8 x 10 x 12 feet);
interactive (touch screen)

SOURCES : reproduit (trois disques laser); encodé (pro-
gramme Pascal avec un système d'opération CP/M)
MATÉRIEL : dupliqué (construction en bois et métal)
ÉQUIPEMENT : dupliqué (deux moniteurs)
PRÉSENTATION : installé (environ 2,5 x 3 x 3,5 m);
interactif (écran tactile)

Grahame Weinbren and Roberta Friedman,
The Erl King, 1982–85
Jeff Rothenberg

Grahame Weinbren and Roberta Friedman's video piece *The Erl King* (1982–85) is in many ways a work ahead of its time. Running the minimal **CP/M operating system** on a primitive Zilog Z-80 based personal computer with a touch screen and a custom-built interface to multiple **laser discs**, this innovative piece provided a degree of **interaction** that is impressive even today. The **authoring program** that was created to enable the piece was far advanced in comparison to personal-computer software of the time. The innovative interactivity of the piece alone is sufficient to make it worthy of preservation, but when combined with the technological prowess that achieved this degree of interactivity using the relatively crude and inexpensive equipment available, preservation becomes all the more desirable—and all the more problematic.

In order to preserve both its behavior and the technological context in which it was created, it would be ideal to preserve the hardware—computer, videodisc drives, touch-sensitive display—utilized by the piece as well as the video content and the software, which created and controlled its behavior. No other approach would preserve *The Erl King* in its original form while retaining the ability to see how it was created and how it overcame the limitations of its hardware/software environment. Since preserving physical hardware indefinitely is not feasible, **emulation** of that hardware appears to be the most promising approach. This would entail preserving all of the work's original software (including its video

Jeff Rothenberg is an independent computer science researcher.

Grahame Weinbren et Roberta Friedman,
The Erl King, 1982–85
Jeff Rothenberg

La vidéo *The Erl King* (1982–85) de Grahame Weinbren et Roberta Friedman est à maints égards une œuvre avant-gardiste. Exploitant le **système d'exploitation CP/M** (programme de commande pour microprocesseur) minimal sur un ordinateur personnel primitif Zilog Z-80 avec un écran tactile et une interface personnalisée pour multiples **disques laser**, cette pièce novatrice offre un degré d'**interaction** qui étonne même de nos jours. Le **programme auteur** créé pour faire la pièce était de loin plus avancé que les logiciels pour ordinateurs personnels de l'époque. L'interactivité novatrice de l'œuvre est en soi suffisante pour qu'elle mérite d'être préservée, mais combinée aux prouesses technologiques qui permettent ce degré d'interactivité au moyen d'équipements disponibles relativement rustiques et bon marché, la préservation devient d'autant plus souhaitable — et d'autant plus problématique.

Afin de sauvegarder tant son comportement que le contexte technologique dans lequel la pièce a été créée, l'idéal serait de préserver le matériel — ordinateur, lecteurs de disques vidéo, écran tactile — utilisé pour la pièce ainsi que le contenu vidéo et le logiciel, qui créait et contrôlait son comportement. Aucune autre approche ne permettrait de préserver *The Erl King* dans sa forme originale tout en conservant la capacité de voir comment l'œuvre a été créée et comment elle vient à bout des limitations de son environnement matériel et logiciel. Étant donné que la préservation indéfinie de l'équipement n'est pas possible, l'**émulation** du matériel semble être l'approche la plus prometteuse. Cela nécessiterait de préserver le logiciel originel de l'œuvre (y compris son contenu

Jeff Rothenberg est un chercheur informaticien indépendant.

content) in **bitstream** form and running that software on a virtual re-creation of the work's original hardware, by writing programs that emulate that hardware on a modern computer.

Note that although the software for *The Erl King* was written in a version of the **Pascal** programming language (Pascal MT+), which originally needed to be **compiled** into **machine code**, the emulation approach would not require running the original compiler to recompile the original **source code**: instead, the executable **object code** would be run directly on the emulated version of the original computer.

Although the emulation approach to preserving this piece seems promising, it does involve some challenges. Emulating the Z-80 processor and the display are relatively straightforward, and it is unnecessary to emulate the details of the controllers for the original floppy-disk drives from which the program ran. The program can simply be transcribed from its original floppy disks onto a modern storage device and fed to the emulated processor as a logical bitstream (at a speed corresponding to the transfer rate of the original floppy disks). However, the videodisc drives and the special-purpose hardware that was built to control them and to allow directing their outputs to the display screen and audio amplifier would require specialized emulation. As with the floppy-disk drives, the best way to do this would probably be to transcribe the videodiscs' contents into bitstreams stored on modern storage devices (or in **main memory**, if speed turns out to be an issue) and to write software that emulates the logical behavior of the original special-purpose controller hardware. Assuming that these challenges can be overcome, emulation appears to offer the best chance of preserving the full richness of *The Erl King*.

vidéo) sous forme de **flux binaire** (*bitstream*) et de l'exploiter sur une re-création virtuelle du matériel originel de l'œuvre, en écrivant des programmes qui imitent ce matériel sur un ordinateur moderne.

Il est à noter que, bien que le logiciel pour *The Erl King* ait été écrit dans une version du langage de programmation **Pascal** (Pascal MT+), qui nécessitait à l'origine d'être **compilé** en **code machine**, l'approche de l'émulation n'exigerait pas d'exploiter le compilateur original pour recompiler le **code source** original : le **code objet** exécutable tournerait directement sur la version émulée de l'ordinateur original.

Si l'approche d'émulation pour préserver cette pièce semble prometteuse, elle comporte certaines contraintes. Émuler le processeur Z-80 et l'écran est assez simple, et il n'est pas nécessaire d'émuler les détails des contrôleurs des lecteurs originels de disquettes qui permettaient d'exploiter le programme. Le programme peut tout simplement être transféré des disquettes d'origine à une unité de stockage moderne, et transmis au processeur émulé en tant que flux binaire logique (à une vitesse correspondant au débit de transfert des disquettes d'origine). Toutefois, les lecteurs de disques vidéo et le matériel spécialisé qui a été conçu pour les contrôler et permettre l'acheminement de leurs données de sortie vers l'écran de visualisation et l'amplificateur audio nécessiteraient une émulation spéciale. Comme dans le cas des lecteurs de disquettes, la meilleure manière de procéder serait probablement de transcrire le contenu des disques vidéo en flux binaire mis en mémoire dans une unité de stockage moderne (ou dans la **mémoire principale**, si la vitesse pose un problème) et d'écrire le logiciel qui émule le comportement logique du matériel spécialisé originel de contrôle. En supposant qu'on puisse surmonter ces difficultés, l'émulation semble offrir les meilleures chances de préserver toute la richesse de *The Erl King*.

The following are excerpts from a conversation with Jon Ippolito, Caitlin Jones, Jeff Rothenberg, Carol Stringari, and Grahame Weinbren, which took place at Weinbren's studio, New York, on October 18, 2002.

Grahame Weinbren: This piece started in about 1982. Roberta Friedman and I based the three-videodisc set up on technology that was developed for the 1982 World's Fair, which I had worked on in 1981 as a video editor and interactive designer. The idea was that it would be a kind of seamless, cinematic piece that the viewer could interrupt at any moment. It was based on a connection I made between two 19th-century texts, "Erlkönig" by Goethe and *The Interpretation of Dreams* by Freud. I became interested in the relationship between these two works and wanted to make a piece that would allow the viewer to similarly discover this connection.

A remarkable thing about Freud is not his analysis of dreams as much as his descriptions of the dreams themselves. He was an incredibly visual writer and I'm interested in the content of the dreams as Freud *saw* them. In one description, a sleeping father dreams that his child taps him on the shoulder and says, "Father, don't you see I'm burning?" And then he wakes up and runs into the other room to find that his child's body is on fire. It's a very intense, visual, cinematic moment that I was totally obsessed with for about ten years. I didn't really know what to do with it until I discovered Goethe's song "Erlkönig," which has this repeated chorus, "Father, don't you see the Erl King? Father, don't you hear the Erl King?" It's about this child who's riding with his father through the woods, and this supernatural creature keeps appearing and saying to the child, "Come with me."

On trouvera ci-dessous des extraits d'une conversation avec Jon Ippolito, Caitlin Jones, Jeff Rothenberg, Carol Stringari et Grahame Weinbren, tenue à l'atelier de Weinbren, New York, le 18 octobre 2002.

Grahame Weinbren : Cette œuvre a commencé autour de 1982. Roberta Friedman et moi avons choisi pour support aux trois vidéodisques la technologie conçue pour l'Exposition universelle de 1982, à laquelle j'avais travaillé en 1981 en tant que spécialiste de montage vidéo et de concepteur interactif. L'idée était que ce serait une sorte d'œuvre filmique en continue que le public pourrait interrompre à tout moment. Elle reposait sur un rapprochement que j'ai fait entre deux textes du XIX^e siècle, « Erlkönig » de Goethe et *L'interprétation des rêves* de Freud. Je me suis intéressé à la relation entre ces deux textes et je désirais faire une œuvre qui permettrait aussi au spectateur de découvrir ce rapprochement.

Un des aspects remarquables de Freud n'est pas tant son analyse des rêves que sa description des rêves mêmes. C'était un auteur très visuel et ce qui m'intéresse, c'est le contenu des rêves tels qu'il les *a vus*. Dans une description, un père endormi rêve que son enfant lui tape sur l'épaule et dit : « Père, ne vois-tu pas que je brûle? » Le père s'éveille et court vers la chambre voisine où son enfant est en feu. C'est un moment filmique et visuel très intense, qui m'a totalement obsédé pendant une dizaine d'années. Je ne savais pas vraiment quoi en faire jusqu'à ce que je découvre « Erlkönig » de Goethe, dans lequel le chœur répète, « Père, ne vois-tu pas le roi des aulnes? Père, n'entends-tu pas le roi des aulnes? ». Il s'agit d'un enfant chevauchant dans la forêt avec son père et qui voit apparaître une créature surnaturelle lui disant, « Viens avec moi ».

Chaque fois que l'enfant voit la créature, il se tourne vers son père et lui demande, « Père, ne vois-tu pas le

Each time the child sees the creature, he turns to his father and says, “Father, don’t you see the Erl King?” to which the Father replies, “No, no, it’s just the shadows of the trees” or “a cloud of mist,” etc. The discovery of parallels between this German, Romantic poem, which is put to music by Schubert, and the dream described by Freud is what really led to the impulse of making a piece in which the viewer could make the same kind of discovery by wandering through a mass of audiovisual material.

Jon Ippolito: When Grahame first showed me a glimpse of the programming behind *The Erl King*, I was struck with this program’s inherent value as an historical artifact, apart from its merely functional role in the larger installation. The **authoring system** has elements of what I, in my loose sense, would call **object-oriented** programming. It’s not a timeline, instead **go-tos** are built like objects that you’re adding properties to, or removing properties from. But I have no idea what programming in **Pascal** was like in 1982; I don’t know how this would compare to other applications written at the time.

Jeff Rothenberg: Well, Pascal was a fairly forward-thinking language and has most of the pre-object-oriented concepts of programming. It was actually a nice environment in which to be doing this sort of thing, but the interface is almost independent of the programming language it’s written in.

Jl: One of the things you notice when you learn a little bit more about these menu items is the level of complexity of the “map”—the ability to program different tracks that cue up in sequence based on how many times someone has touched a particular target on the screen. That structure entrains the authoring system with the data output. You

roi des aulnes? » ce à quoi le père répond, « Non, non, ce n’est que l’ombre des arbres » ou « une traînée de brume », etc. La découverte de parallèles entre ce poème romantique allemand, mis en musique par Schubert, et le rêve décrit par Freud est vraiment ce qui a conduit à faire une œuvre qui permettrait au public de faire une découverte semblable en s’aventurant dans une masse de matériel audiovisuel.

Jon Ippolito : Lorsque Grahame m’a donné un premier aperçu de la programmation derrière *The Erl King*, j’ai été frappé par la valeur inhérente de ce programme en tant qu’artéfact historique, mis à part son rôle purement fonctionnel dans l’installation. Le **système-auteur** a des éléments de ce que je qualifierais, assez librement, de programmation **orientée objet**. Il ne s’agit pas d’un programme temporel mais plutôt d’instructions « **aller à** » conçues comme des objets auxquels on ajoute ou retranche des propriétés. Mais je n’ai aucune idée à quoi la programmation en **Pascal** ressemblait en 1982; je ne sais pas comment cela pourrait se comparer à d’autres applications écrites à l’époque.

Jeff Rothenberg : Bien, Pascal était un langage assez avant-gardiste possédant la plupart des concepts antérieurs à ceux de la programmation orientée objet. C’était en fait un bon environnement pour ce genre de projet, mais l’interface est presque indépendante du langage de programmation utilisé pour l’écrire.

Jl : Une des choses que l’on constate lorsqu’on en sait un peu plus au sujet de ces commandes de menu est le degré de complexité de la « carte » — la capacité de programmer différentes pistes qui s’ordonnent en séquence selon le nombre de fois où l’on touche une cible particulière sur l’écran. Cette structure entraîne le système-auteur avec la sortie de données. Vous avez mentionné avoir travaillé auparavant avec un programmeur qui avait essayé de **migrer** le code. Cela a été ma première réflexion, « Oh, essayons cela dans

mentioned that you had been working before with a programmer who had tried to **migrate** the code. That was my first thought, “Oh, let’s try this in **Director**, we’ll just get all the **cue points** and all the video,” but the cue points are intricately connected to this map that’s been made of *what* someone has touched *when*. So it’s not at all obvious how you’d even extricate data from code; they seem to be intertwined.

GW: Not only are the code and data inseparable, but the authoring system (the part of the software the artist uses to plot pathways through the data) and the run-time system (the part of the software that controls the hardware system during exhibition) are equally inseparable. For example, the use of **global conditionals**, which define a kind of map of the data, indicates which parts have already been accessed. Based on this map, the software closes off some passages and makes others available. Another set of issues comes out of the data structures, which depend on separating video from audio. The most significant of these is the “interactive cutaway”: the viewer touches the screen, a new video image appears while the audio continues, and the screen eventually returns to the original video. This syntax is used throughout the piece and requires a lot of complex parameters which are defined in the authoring system and acted on by the run-time system.

Those are the issues that don’t translate easily. And in fact, I used Director recently for my piece *Frames*—it’s a lot simpler than *The Erl King*. We couldn’t come close to the elaborate structures of *The Erl King* with Director. Nothing is missing from *Frames*, which uses a much simpler paradigm of interactivity, but the options just aren’t there. [Another programmer] tried to migrate the Limosine system (this is what we

Director, nous obtiendrons tous les **points de signaux** et toute la vidéo », mais les points de signaux sont reliés de manière compliquée à cette carte générée par *les cibles* touchées à l’écran et *le moment* où elles sont touchées. Donc la façon d’extraire des données du code n’est pas si évidente que cela; les deux semblent indissociables.

GW : Non seulement le code et les données sont-ils inséparables, mais le système-auteur (la partie du logiciel que l’artiste utilise pour tracer des voies d’accès dans les données) et le système d’exécution (la partie du logiciel qui contrôle le matériel durant l’exposition) sont aussi inséparables. Par exemple, l’utilisation de **conditionnels globaux**, qui définissent une sorte de carte des données, indique les parties auxquelles on a déjà accédé. Le logiciel ferme certains passages et en rend d’autres accessibles en fonction de cette carte. D’autres questions émergent des structures de données, qui dépendent de la séparation de la vidéo et de l’audio. La plus importante d’entre elles est le « décrochage interactif » : une personne touche à l’écran, une nouvelle image vidéo s’affiche tandis que l’audio continue, et l’écran revient par la suite à la vidéo d’origine. Cette syntaxe est utilisée tout au long de l’œuvre et nécessite beaucoup de paramètres complexes qui sont définis dans le système-auteur et auxquels le système d’exécution donne suite.

Ce sont là des questions qui ne se traduisent pas aisément. Et en fait, j’ai utilisé Director récemment pour mon œuvre *Frames* — c’est beaucoup plus simple que *The Erl King*. Director ne nous permettait pas de produire même un tant soit peu les structures complexes de *The Erl King*. Rien ne manque à *Frames*, qui utilise un paradigme beaucoup plus simple d’interactivité, mais les possibilités sont tout simplement absentes. [Un autre programmeur] a essayé de migrer le système Limosine (c’est ainsi que nous appelions le logiciel pour *The Erl King*) vers le langage logiciel **Visual Basic**. Mais il y avait beaucoup trop de détails.

called the software that runs *The Erl King*) to the **Visual Basic** software language. But there were too many details.

JR: There are many reasons this piece would be a case for **emulation**. For the digital part of the work, it would be a shame to have to re-create a program that is still running. First of all it wouldn't be the same program when finished, and secondly, it takes a lot of work and is error prone. Who would decide whether it really was working the same way or not? So the idea is, there's no reason you can't keep running this program except that that machine becomes obsolete, but you can emulate that machine on any future computer and keep running that original program essentially forever.

That brings up another set of issues: Do we try to emulate analog devices such as **laser disc** players and the video monitor using digital technology. The most obvious solution, at this moment in technology, would be, "Let's take all this analog stuff and make it digital." So you copy essentially all the source material to digital form. Now you have a digital controller, which is talking to all these old analog devices, which you've now replaced. So you'd need to build a digital converter or translator, which takes all the output signals from the computer, which is saying, "OK, this meant go to frame such and such," to the videodiscs. I interpret that as doing something else on a DVD or a hard drive or some other way of accessing the material.

The problem is that you now have access feed, size, and speed issues with digital video. As Grahame was saying, that would be pushing current digital technology. Digital is not quite there yet; it's getting there, but it can't quite do all of

JR : Il existe maintes raisons pour lesquelles cette œuvre pourrait faire l'objet d'une **émulation**. Pour la partie numérique de l'œuvre, ce serait dommage d'avoir à re-cr  er un programme qui fonctionne toujours. Premièrement, ce ne serait pas le m  me programme une fois fini, et deuxi  mement, cela demande beaucoup de travail, sujet    l'erreur. Qui d  ciderait s'il fonctionne r  ellement de la m  me mani  re ou non? Donc, la seule raison qui emp  cherait l'ex  cution de ce programme est la d  suetude de la machine, mais on peut   muler cette derni  re sur tout ordinateur futur et continuer d'ex  cuter le programme d'origine pratiquement pour toujours.

Cela soul  ve d'autres questions : Tentons-nous d'  muler des dispositifs analogiques tels que des lecteurs de **disques lasers** et le moniteur vid  o au moyen de la technologie num  rique? La solution la plus   vidente, avec la technologie d'aujourd'hui, serait de dire : « Convertissons tout ce qui est analogique en num  rique ». Il s'agit essentiellement de copier tout le mat  riel source en num  rique. On aurait alors une unit   de contr  le num  rique qui communiquerait avec tous les vieux dispositifs analogiques qu'on aurait remplac  s. Il faudrait donc concevoir un convertisseur ou un traducteur num  rique, qui prendrait tous les signaux de sortie de l'ordinateur et qui indiquerait aux vid  odisques « O.K., cela signifiait aller    telle et telle image ». J'interpr  te cela comme faire autre chose sur un DVD ou un disque dur, ou comme une autre mani  re d'acc  der au mat  riel.

Le probl  me, c'est qu'on a maintenant des probl  mes d'acc  s, d'alimentation, de taille et de vitesse avec la vid  o num  rique. Comme le mentionnait Grahame, cela   quivaudrait    forcer la technologie num  rique actuelle. Le num  rique n'est pas encore rendu    ce point; il y arrive, mais il ne peut pas faire tout ce que fait l'analogique aussi rapidement et aussi uniform  ment que les lecteurs de disques lasers analogiques.

this analog stuff as fast and as seamlessly as the analog laser disc players.

Jl: Beyond the technical complexities, what's exciting to me about this case study is how Grahame's authoring system prefigures today's **software art**. Programs like Auto-Illustrator and nato are more artist-made mediums than single pieces, and *The Erl King* offers an invaluable precedent for how to preserve works of this kind.

Jl : Au-delà des complexités techniques, ce que je trouve passionnant au sujet de cette étude de cas, c'est la manière dont le système-auteur de Grahame préfigure l'**art logiciel** d'aujourd'hui. Des programmes comme Auto-Illustrator et nato constituent davantage des médiums conçus par des artistes que des œuvres uniques, et *The Erl King* sert de précieux précédent sur la manière de préserver des œuvres de ce genre.