



## Digital transcoding: an extension of translation?

Philippe Bootz

### ► To cite this version:

Philippe Bootz. Digital transcoding: an extension of translation?. New Perspectives on Translating Electronic Literature: Whither Now?, Jan 2020, Aubervilliers, France. sic\_03213834

**HAL Id: sic\_03213834**

**[https://archivesic.ccsd.cnrs.fr/sic\\_03213834](https://archivesic.ccsd.cnrs.fr/sic_03213834)**

Submitted on 30 Apr 2021

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Digital transcoding: an extension of translation?

## 1 The impact of theoretical models on the concept of extended translation

### 1.1 The concept of translation as an "umbrella" model

Søren Pold & all (2018) propose to extend the concept of (linguistic) translation in 4 directions: translinguistics, transcoding, transmedia and transcreation. In this perspective, updating, remediation, adaptation of a medium and classical linguistic translation would be based on a single model: translation. What would be the added value of such a broadening of the concept of translation, and what biases could it bring?

Grouping different concepts in the same class is not without danger. Indeed, bringing concepts together under the "umbrella" of another, considering them *in fine* as belonging to the same class defined by this "umbrella" concept changes the way of thinking about these concepts by projecting onto them a new understanding and constraints stemming from the philosophical and ideological characteristics of the generative concept of the class. This is not insignificant at all.

Linguistic translation is based on two fundamental characteristics: the existence of a source text and the fact that the *entire* source text is translated into a *single* target text. The existence of a source text is a *sine qua non* of linguistic translation, and ideally it should be an original text. This means that, as part of an extended concept of translation, all the features of the original source digital work must be translated *as a whole* into a *single* target digital work in which, on the one hand, the target program is a translation of the original program and, on the other hand, the screen rendering of the target program while running, translates the original rendering. This is exactly what it says:

*“we could say that reinventing sound and meaning correlations [...] or reinventing sound, meaning and visual correlations [...] are theoretically equivalent to reinventing the correlations between those verbal and visual layers, on the one hand, and software layers, on the other, in electronic literature.”* (Pold & all, 2018: 3)

We will argue that the concept of "extended translation" comes up against this concomitance of several "translations".

### 1.2 The underlying theoretical model of the digital work

Pold & all explicitly base their argument on cybertext theory (Aarseth, 1997):

*“Translation of this textoniclayer (in Aarseth’s terminology) can be described as the translation of processes rather than texts (Cayley 2018). Translation of electronic literature thus seems to require that our focus is placed on the programmed compositional processes as much as on the textual and media instantiations that constitute the more or less transient configuration of the scriptonic layers”* (Pold & all, 2018: 2)

## 2 Cybertext and its alternative procedural model

### 2.1 The cybertext model

The following schema, which applies to any operator (author as well as reader), defines the cybertext

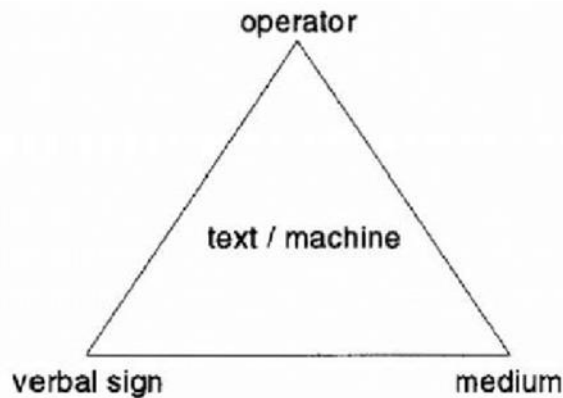


Figure 1: schema of the cybertext (Aarseth, 1997: 21)

This model has many qualities but also several limitations that make it ineffective for analyzing certain properties of real digital works. Let's examine the implicit and limitations of cybertext in the case of the digital.

### 2.2 The work in the cybertext model

#### a) Cybertext is classically based on the "reader's point of view"

The theory of cybertext is part of the classical paradigm of the work, which can be formulated, for digital works, in the following way: "*the work is what can be read on the screen*". This point of view appears implicitly in Aarseth's definition of the fundamental objects of his model:

*"A text, then, is any object with the primary function to relay verbal information (...) It is useful to distinguish between strings as they appear to readers and strings as they exist in the text, since these may not always be the same. For want of better terms, I call the former scriptons and the latter textons (...) In addition to textons and scrip'tons, a text consists of what I call a traversal function-the mechanism by which scriptons are revealed or generated from textons and presented to the user of the text."* (Aarseth, 1997: 62)

An absent author: The author is the great absentee of cybertext. The model never talks about what the text machine is to the author. Although Aarseth refutes the theoretical relevance of the distinction between author and reader, in his examples he only mentions reading operations and only talks about "manipulation". One can only manipulate what already exists. Verbal signs have to be created for the text machine to work. Indeed, a "verbal" sign is a sign readable by man in his language, in other words a sign that emerges on reading. We deduce from this that, in a very classical way, writing is conceived as a reading anticipated by the reader; its sole purpose is the creation of the textual machine of reading.

Lack of semiotic autonomy of textons: Aarseth explicitly states that verbal information is relayed between textons and scriptons through the implementation of a traversal function. The traversal function relays all the information contained in the textons. This implies that, even if a given reader does not discover all the information allowed by the machine, the infinite set of readers will, globally, read all the information contained in the textons. There is therefore no semiotic autonomy of textons, i.e. under no circumstances can they carry information that would never appear on reading. This subordination of the created sign to the read sign is the direct consequence of the author's absence: since the author is only a reader by anticipation, the meaning of any sign he creates (texton) is necessarily readable in reading situations, even if the form of the sign that actualizes it is different (scripton). In semiotic terms, we can say that textons and scriptons point to the same meaning in the encyclopedia or, put another way, that scriptons are merely a re-mediatization of textons. The true nature of the traversal function then appears: it is a process of re-mediatization.

The traversal function is not a transformation of the code into a media but a transmission: The traversal function is not *in fine* interested, neither in the computing specificity of the code, nor in the media built during the execution of this code, but only in the transmission of the information of the textons to the scriptons. Cybertext therefore implicitly takes into account only part of the computer code and part of the media produced. This truncated consideration of physical reality is also characteristic of a classical conception according to which the text is an abstract, purely linguistic object, an "object in itself" disconnected from all materiality. The cybertext schema (Figure 1) explains this conception of text as an "object in itself". In the classical conception this induces the concept of support (the text is in material relation with its support) and the cybertext is only an extrapolation of this conception in which the support becomes a machine which is nothing but a set of manipulable supports: the screen is the support of scriptons and the code is the support of textons.

Cybertext falls into the category of "front-end dispositives": Finally, we can say that cybertext is part of a long western tradition that the development of printing has forged, then later cinema and recorded music, and which I call the "front-end dispositive". In this conception, writing only aims at constructing readability via a single reading modality. In other words, the text is "transparent" for this reading modality, it has no hidden parts. Everything happens as if the author and the reader were on the same side of the text, one writing what the other reads or, in the case of cybertext, one building the machine that the other manipulates. The communicational model underlying this dispositive is none other than that of Shannon and Weaver.

**b) Cybertext does not address the actual technical properties of the "text machine".**

The cybertext model says nothing about the actual technical nature of the traversal function, nor about the actual physical dispositive in which it is embedded. In this model, the relationship between textons and scriptons is causal: the scripton is totally determined by the texton and the traversal function. However, the real digital dispositive has a fundamental property that cybertext does not deal with: lability. The lability consists in the dependence of the result of the execution of a program on the technological context of this execution. It is above all a technological fact. Clearly, the same program that is neither generative nor hypertextual, but merely describes a displayed process, for example an animation, actually produces different animations depending on the technological context in which it is executed.

This is why this feature was only discovered in 1990 (Bootz, 1990) on animated poetry and not on text generators, the generative character masking it.

This technical lability induces of course meaning effects and is therefore accompanied by semiotic lability; it impacts reading, sometimes in a very significant way (Bootz, 2012).

Considering that the effective traversal function during an execution is an instance of a class of functions would make it possible to account for some of the observed effects. In this case, we should consider that Aarseth's analysis of traversal functions is relative, not to the actual traversal functions at work, but to the classes they instantiate. In other words, the real traversal function would be virtual, dependent on an unpredictable context of execution. This approach does not destroy the causal character of the traversal function.

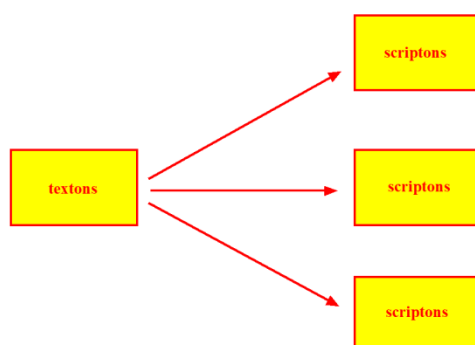


Figure 2: treatment of lability in cybertext by virtualization of the traversal function

Unfortunately, the actual observation of lability phenomena shows that sometimes scriptons appear, signs in the read media, which have no cause in the code, which do not emanate from any texton. This phenomenon is logical because lability is not caused by the program but by the complete technological context at runtime. This phenomenon is also observed in cases where the technological context is modified by a hardware change that has no impact on the execution of the program. For example, Xavier Hautbois and I observed that the reading of a video file of a sequence from Hans Richter's *Rythm 21* generated two different semiotic units depending on whether it was viewed from the same computer on a CRT or LCD screen. The cybertext model cannot account for this observation.

For all these reasons, I have proposed a more pragmatic alternative model of the digital work: the procedural model (Bootz, 1996, 2001).

## 2.3 The procedural model

### a) Presentation of the model

It is a general model of communication, which has been mainly constructed from the analysis of digital literary works but that is not limited to this type of production. Its generic presentation also uses examples that are not digital (Bootz, 2016). Let us present it very quickly:

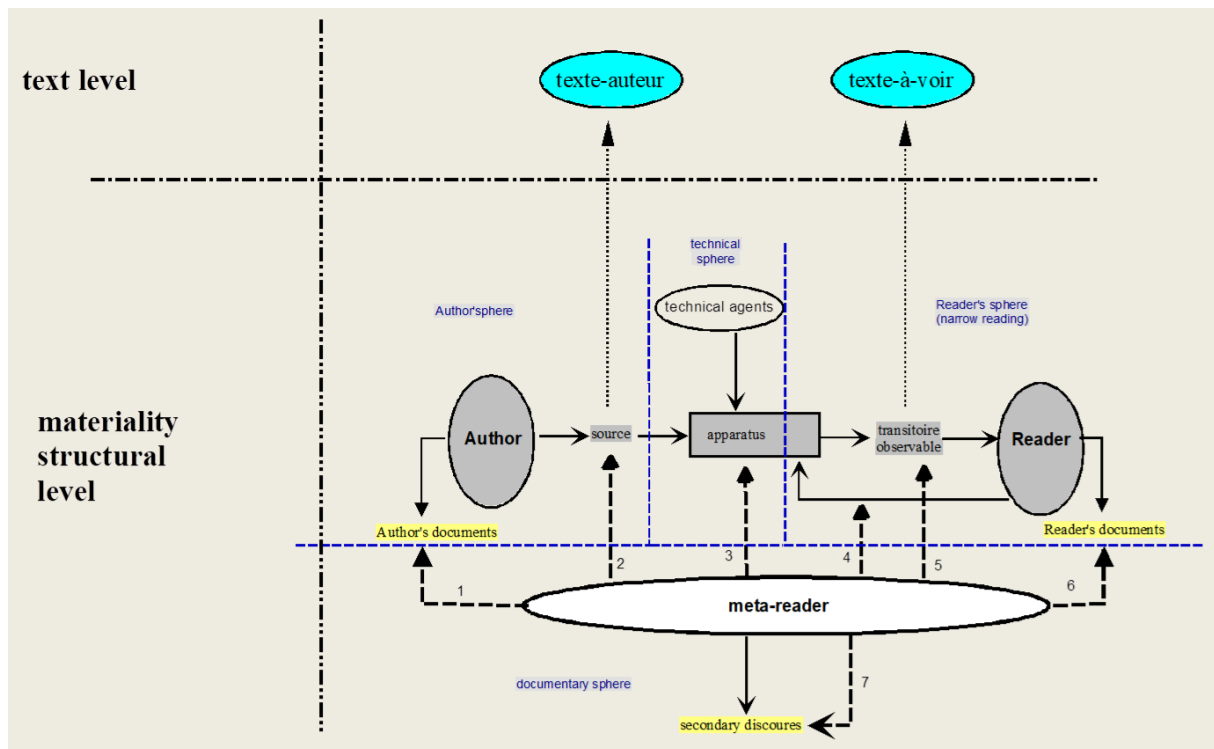


Figure 3: schema of the procedural model

This model makes a strict difference between 3 dimensions: material (and technological), semiotic, and psychological in the broadest sense (influence of culture, waiting horizon, emotional state, spatio-temporal context of individuals...). It considers that the situation of communication results from a physical transformation of a material work material, the *source*, into another work material, the *transitoire observable*, via an apparatus. This transformation then defines a "main communicational axis" that carries the apparatus and two particular roles: *Author* and *Reader* who are at the two extremities of the transformation.

In digital productions, the *source* is the source program, a form that the author can manipulate in her authoring environment. The *transitoire observable* is the physical phenomenon appearing on the reader's screen. The apparatus consists of all the machines involved in the transformation between the *source* and the *transitoire observable*. In the case of a Web production, the apparatus includes part of the Internet. *Source* and *transitoire observable* are not signs but physical phenomena that an ad-hoc dispositive can capture, for example a printer for an encoded *source* and a video capture dispositive for a *transitoire observable* produced on a screen. These physical events constitute the stimulus, in the Klinkenberg (1996) sense<sup>1</sup>, of texts which are: the *texte-auteur* (on the *source*) and the *texte-à-voir* (on the *transitoire observable*). It is quite obvious that, in a programming or writing operation, an author "writes" a *texte-auteur*, even if he acts strictly speaking on the *source*, and a reader, conversely, reads a *texte-à-voir*, even if he perceives a *transitoire observable*. Physically, he does not manipulate the *transitoire observable* but rather the apparatus.

Materiality is not described by a "machine" but by a "dispositive" that encompasses all the technological artifacts that participate in the transformation and all the human actors in contact, in one way or another, with the main communication axis. These individuals are divided, at a given time, into classes that define roles: *Author*, *Reader*, *technical agents* and

<sup>1</sup> The stimulus is the materiality of the sign

*meta-reader*. These roles may be collective, just as, conversely, a given individual may alternate various roles during his or her participation in the dispositive. This dispositive is structural. In other words, the transformation can be synchronous (face-to-face conversation, for example) or asynchronous (reading a previously written book). The part of the dispositive that carries the transformation and thus defines the main communication axis is called the main dispositive (shaded part in figure 3).

The psychological constraint is represented in the model by an external influence on perception and cognition, called “*profondeur de dispositif*”, which influences, for each actor, the construction of the *texte-à-voir* and the *texte-auteur*.

#### **b)      *Autonomy of texte-à-voir and texte-auteur***

The physical transformation is not a traversal function. Due to the existence of lability and of the *profondeur de dispositif*, the procedural model establishes a total independence between the *texte-auteur* and the *texte-à-voir*. The model is therefore symmetrical: *texte-auteur* and *texte-à-voir* can be considered separately. No causal relation exists between them, only correlations, even if the intentionality of the writing aims at creating a causal relation.

#### **c)      *Braids and snatches in texts***

In this model, the concepts of *texton* and *scripton* are not relevant. This is shown in figure 4: a concrete poem is spotted on a portion of the *texte-auteur* of *first screening* (BpNichol, 1984), but the execution of this fragment of source code does not produce in the *texte-à-voir* any sign of the same level of articulation, but only the beginning of a sentence. In other words, in this example, the *textons* form a *lexie* (Barthes, 1970) but not the associated *scriptons*. This is why the procedural model does not decompose *texte-auteur* and *texte-à-voir* into *textons* and *scriptons*, but into *braids* and *snatches*, respectively. *Braids* and *snatches* are *lexies*. They therefore depend directly on the individual who perceives and constructs the *texte-auteur* and/or the *texte-à-voir*. Depending on one's culture, a *lexie* for one person, for example a concrete poem, is not necessarily a *lexie* for another. Because of the autonomy of the *texte-auteur* and the *texte-à-voir*, no semiotic transformation function links the *braids* to the *snatches*. The physical transformation connecting the part of the *source* stimulus of a perceived *braid* to the part of a *transitoire observable* stimulus of a perceived *snatch* only exists, just as the physical transformation that has transformed part of the *source* into the part of the *transitoire observable* stimulus of a perceived *snatch* exists.

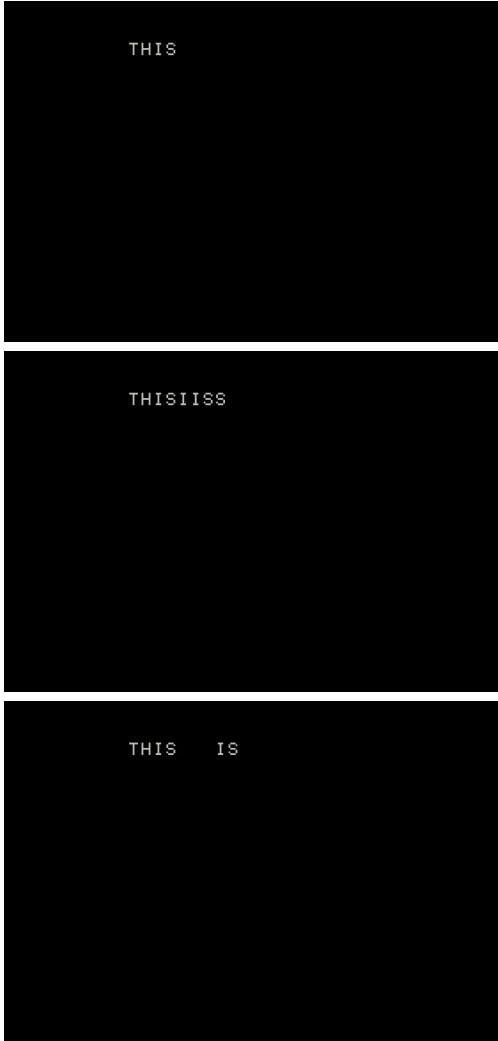
<pre> 1920 HOME 1925 FOR PAUSE = 1 TO 2500: NEXT 2005 SPEED= 210 2010 VTAB 4: HTAB 5: PRINT "THIS" 2015 VTAB 4: HTAB 5: PRINT " THIS" 2020 VTAB 4: HTAB 5: PRINT "  THIS" 2025 VTAB 4: HTAB 5: PRINT "   THIS" 2030 VTAB 4: HTAB 5: PRINT "    THIS" 2035 VTAB 4: HTAB 5: PRINT "     THIS" 2040 VTAB 4: HTAB 5: PRINT "      THIS" 2045 VTAB 4: HTAB 5: PRINT "       THISS" 2050 VTAB 4: HTAB 5: PRINT "        THISSS" 2055 VTAB 4: HTAB 5: PRINT "         THIISSS" 2060 VTAB 4: HTAB 5: PRINT "          THIISSSS" 2065 VTAB 4: HTAB 5: PRINT "           THIISSSS" 2070 VTAB 4: HTAB 5: PRINT "            THIISSSS" 2075 VTAB 4: HTAB 5: PRINT "             THIS ISS" 2080 VTAB 4: HTAB 5: PRINT "              THIS IS" 2085 VTAB 4: HTAB 5: PRINT "               THIS IS" 2090 VTAB 4: HTAB 5: PRINT "                THIS IS" 2100 VTAB 4: HTAB 5: PRINT "                 THIS IS" 2105 VTAB 4: HTAB 5: PRINT "                  THIS IS" 2110 VTAB 4: HTAB 5: PRINT "                   THIS I " 2115 VTAB 5: HTAB 22: PRINT "S" 2120 VTAB 4: HTAB 5: PRINT "    THIS    " 2125 VTAB 5: HTAB 21: PRINT "IS" </pre>	
Concrete <i>Texte-auteur</i> concret in <i>source</i>	screen production by this <i>source</i>

Figure 4: *braid* on the *source* of *first screening* and screenshot of its execution

#### d) Meta-reading

Let us insist on one point: any individual participating in a role having access to the *source* builds on it a *texte-auteur* and any individual having access to the *transitoire observable* builds on it a *texte-à-voir*. The model indicates that a particular role, the meta-reader, has joint access to these two material components.

The meta-reader role is a determining element of the model. In particular, any person analyzing a work is in a meta-reading situation. We will argue below that this role is also a real position of reception of the work, complementary to the digital reading to which we are accustomed (called *narrow reading* in the model), in that it gives access to aesthetic and semiotic dimensions that never appear on the screen. Any meta-reading is based on the joint use of several modalities of access to the material components of the *dispositif* among the 7 identified (figure 3). These modalities are instrumented, using protocols and investigative tools.



## 2.4 Comparison between the two models

The Aarseth text machine appears to be the form in which the main dispositive is seen from the position of the *Reader* in the procedural model. Yet profound differences exist between these two models.

In both models, text is divided into two sets of autonomous signs: scriptons and textons in the cybertext, *texte-auteur* and *texte-à-voir* in the procedural model. The textons are in the *texte-auteur* and the scriptons are in the *texte-à-voir*. However, the two models do not really deal with the same types of signs. In cybertext, textons and scriptons are "verbal signs", i.e. linguistic entities. In the procedural model, on the other hand, *textes-auteur* and *textes-à-voir* can consist of any type of signs. On the other hand, textons and scriptons are not equivalent to *braids* and *snatches*.

In cybertext, all readers who perform the same manipulations access the same textons and scriptons when the program, if it is generative, makes the same decisions. This is not the case in the procedural model: *textes-auteur* and *textes-à-voir* may differ due to lability and differences in the individual *profondeurs de dispositif*.

We have pointed out that lability requires a redefinition of the traversal function. Moreover, the position of meta-reader is totally absent in cybertext and there is no way to add it. For the meta-reader, *texte-auteur* and *texte-à-voir* are two distinct parts of the same set of signs. She can even add a third set: the Reader's behavior. These three sets are therefore not linked by a traversal function type transformation, but by rhetorical relationships, some of which we present below. Rather than considering a function that transforms signs, the procedural model considers a material transformation that transforms entities in the material world, and which therefore remains relevant whatever are the semiotic processes in the dispositive.

In the end, the procedural model is not compatible with the frontal dispositive, especially since the three roles access different and disjointed parts of the dispositive.

## 2.5 Rhetoric of Dual Signs in Meta-Reading

### a) The three classes of signs in the main dispositive

The sign space develops in the procedural model in 3 physical spaces disjointed from the main dispositive (*source*, *transitoire observable* and reader's reaction) that only the *meta-reader* can apprehend simultaneously. The model thus defines 3 classes of signs, different but not exclusive, with the same sign belonging to one or two classes. These classes are:

- The unary signs: These signs are perceived in a *texte-auteur* or a *texte-à-voir* read autonomously, without any relation with the dispositive, as if the dispositive only is the support. Most of the analyses are limited to the consideration of a single *texte-à-voir* and therefore deal only with unary signs. All roles have access to unary sign spaces.

- The dual signs. These signs develop on two distinct physical spaces. They are the most interesting signs in digital literature because they allow a rich set of rhetorical figures. Only the roles *Reader* and *meta-reader* can access dual signs. For the *Reader*, the textual space of dual signs consists of the *texte-à-voir* and its reading activity (the signs

corresponding to the latter constitute "double reading" (Bootz, 2004)), whereas for the meta-reader, the dual signs relate his text-author and his text-to-view.

- Ternary signs: these develop on the *texte-à-voir*, the *texte-auteur* and the reader's reaction. Only the *meta-reader* accesses this category of signs through the joint implementation of meta-reading modalities 2, 4 and 5. These signs mainly concern interactive works and the aesthetics of frustration.

Signifying spaces are added to these spaces of signs. They are physical spaces that do not allow the construction of signs, such as the inside of the apparatus, but to which one can give meaning. (Baldwin, 2009) gives a typical example of this.

## b) Rhetorical figures on dual signs

Figures of rhetoric develop between the *texte-auteur* and the *texte-à-voir* that a *meta-reader* perceives. Let us recall some examples that I have previously analyzed in several papers and lectures. They are a sequence of *passage* (Bootz, 2009) - top illustration in figure 5, and an obfuscated C work by Eric Marshall (1986) - bottom illustration. The rhetorical figures detected are different depending on whether they are based solely on meta-reading modalities 2 and 5 or if we add to them the consideration (3) of the apparatus. The latter then constitute "material metaphors" (Hayles, 2002) since they are based on the physical functioning of the main dispositive.

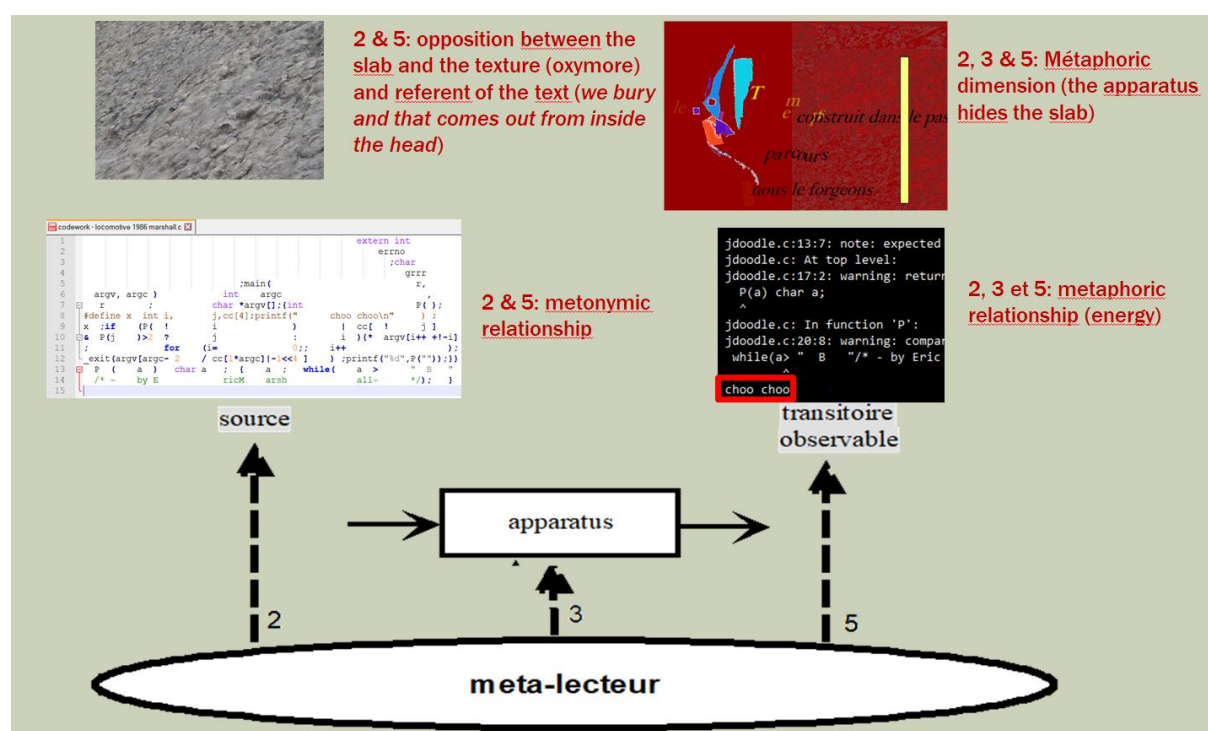


Figure 5: rhetorical figures in dual signs

In the sequence of *passage*, the *texte-auteur* includes a photo of the largest ammonite slab in Europe. This photo never appears in the *texte-à-voir*, but it is used to create a very moving and rhythmic texture in which geometric figures that are absent from the *texte-auteur* appear. Photo and textures constitute respectively a graphic *braid* and a graphic *snatch*. Observation through meta-readings modalities 2 and 5 puts them in relation within an oxymoron type rhetorical figure: the ammonite slab being a static object with a connotation of

death, whereas the texture is moving and dynamic with a connotation of life. The text scrolling on the screen does not refer to the slab in the unary sign of the *texte-à-voir*, whereas in the dual sign, the ammonite slab appears as the referent, which sheds new light on this text and changes its meaning.

The apparatus can be perceived as meaningful. It constitutes the screen that prevents the reader from accessing the photo. The apparatus then plays the role of the rock covering the fossil slab: it must be broken (and metaphorically switched to *meta-reading* and not *narrow reading*) to access the slab. The material metaphor here is of a metaphorical type whereas it is based on the same dual sign!

In Marshall's work, the comparison (2 and 5) between the *texte-auteur*, a locomotive calligram, and the *texte-à-voir*, the choo choo written on the screen, creates a rhetorical figure of metonymic type: the locomotive and the "choo choo" are in semantic continuity. On the other hand, taking into account the apparatus gives a metaphorical dimension to the main dispositive: the apparatus must be supplied with energy so that the program writes "choo choo" just as the locomotive must be supplied with energy so that it emits a sound typical of its operation.

### c) The overall rhetorical figure

Generally, the perception that the *texte-auteur* gives of the work in its unary signs space does not cover the perception that emanates from the *texte-à-voir* in its one. This difference cannot be perceived by the roles *Author* and *Reader* since they do not access the same spaces. On the other hand, it is manifested in *meta-reading*. This difference is then perceived as a trope: it constitutes the overall rhetorical figure that emanates from *meta-reading*.

This figure necessarily exists in digital literature simply because of the fundamental difference in status between the *source* and the *transitoire observable*: the *source*, as a source program, is perennial, and even printable, whereas the *transitoire observable* is ephemeral, it exists only when the transformation is active in the main dispositive. This perennial/ephemeral opposition is projected onto the *texte-auteur/texte-à-voir* couple. By the concomitant presence in *meta-reading* of these opposed qualities, the overall figure thus has at least an oxymoronic dimension. Following the advice of the mu group, which recommends that the names of linguistic rhetorical figures should not be used in other semiotic codes because the process of the figure and its characteristics are different there, I propose to name "proxymoron" (for procedural oxymoron) this overall figure.

The oxymoronic dimension of the proxymoron can be reinforced by other oppositions or supplemented by other dimensions. For example, in recent works, the *texte-auteur* is often fragmented, split on several components of the *source* (several files in the case of a coded program, timeline (or graph) and scripts in other cases...) whereas the *texte-à-voir* constitutes a single homogeneous whole, especially in non-hypertextual works. This feature amplifies the oxymoron dimension of the proxymoron.

As an example, I have noted the oppositions between the *texte-auteur* and *texte-à-voir* that construct the proxymoron in my meta-reading of Dutey's work *Le mange-texte* (1989). In figure 6, the general oppositions present in all the early works are indicated under the more specific oppositions in this work.

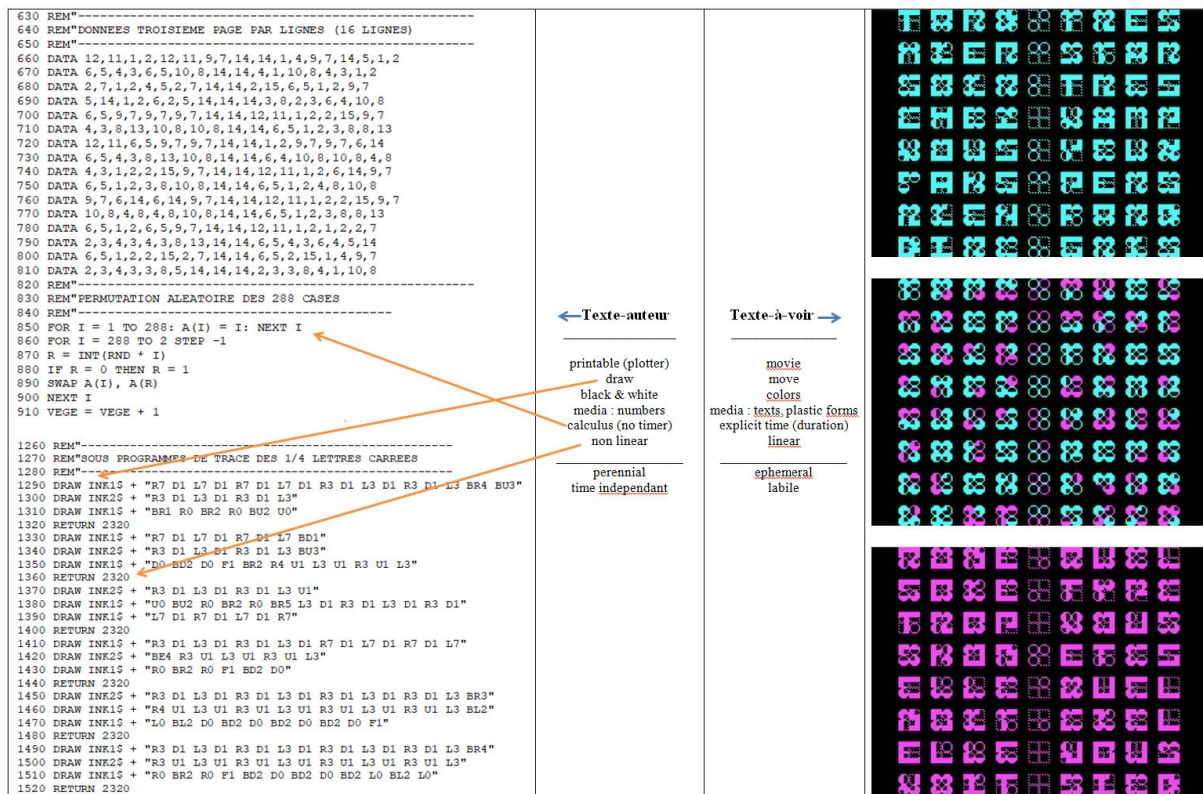


Figure 6: proxymoron in *le mange-texte*

## 2. 6 Elements of analysis of Tibor Papp's work *Orion*

### a) Programmed writing versus computer programming

The study of *Orion*, a Tibor Papp's (2000) work programmed under Macromedia Director, highlights figures of *meta-reading*. This work appears on the screen as a long continuous animation in which it is difficult to distinguish sharp sequences because those determined from sound and those determined from the background images overlap. The *texte-à-voir* is therefore very linear.

The *source* is programmed in a very particular way, which a computer scientist would consider profoundly naive because the same scripts are repeated when it would have sufficed to reuse them, and because they are sometimes made up of exactly the same instructions, only the values of the data change. It would therefore have been wiser to introduce an argument to pass to a single script the different values used in the different parts of the work. Tibor knew how to program perfectly well, and therefore a *meta-reader's* eye is not fooled by this first impression left by the functional, technical dimension of the *source*. In the *texte-auteur*, the source program is not reduced to its computer dimension alone. The challenge in constructing a *texte-auteur* is precisely to capture the non-computer dimensions whose *source* is also a stimulus in the sense of Klinkenberg.

In *Orion*, these computer "naiveties" are precisely the clue to true programmed writing: the order of repetition of the scripts marks the true sequences of the work. We can see in particular that they are based on the visual and not on sound. The "naïve" scripts that do not use variables are all time delay scripts. It is therefore understandable that the temporality in the *texte-auteur* is not continuous as it is in the *texte-à-voir*, but built on 2 bricks of

elementary duration of 5 and 3 seconds that the author arranges a bit like measures on a score. Moreover, the sequences are actually non-linear constructions in the *texte-auteur*. If we consider that the sequence is an element at the level of a "paragraph", each sequence is constructed from two types of "sentences": the main pattern, different from one sequence to another but unique for each sequence, and the "transversal processes" which are animations. The main pattern is an animation on a background image. Transversal processes are animations of elements without a background image. The processing on the timeline of these two components is not the same: the main pattern develops on the timeline according to a succession of frames on which timing scripts can be added to break its rhythm, whereas the transversal processes are totally programmed, each on one or two frames. They thus pause, as it were, one frame of the main pattern. Transversal processes are never repeated, whereas the main pattern can be repeated, often with sonic variations. Thus, the programming of the work obeys a perfectly defined and rigorous grammar: it is by no means a "description" of what appears on the screen, but a real statement; a *texte-auteur*. In *meta-reading*, the opposition between this non-linear grammar of the *texte-auteur* and the linear grammar of the animation in the *texte-à-voir* reinforces the oxymoronic dimension of the proxymoron.

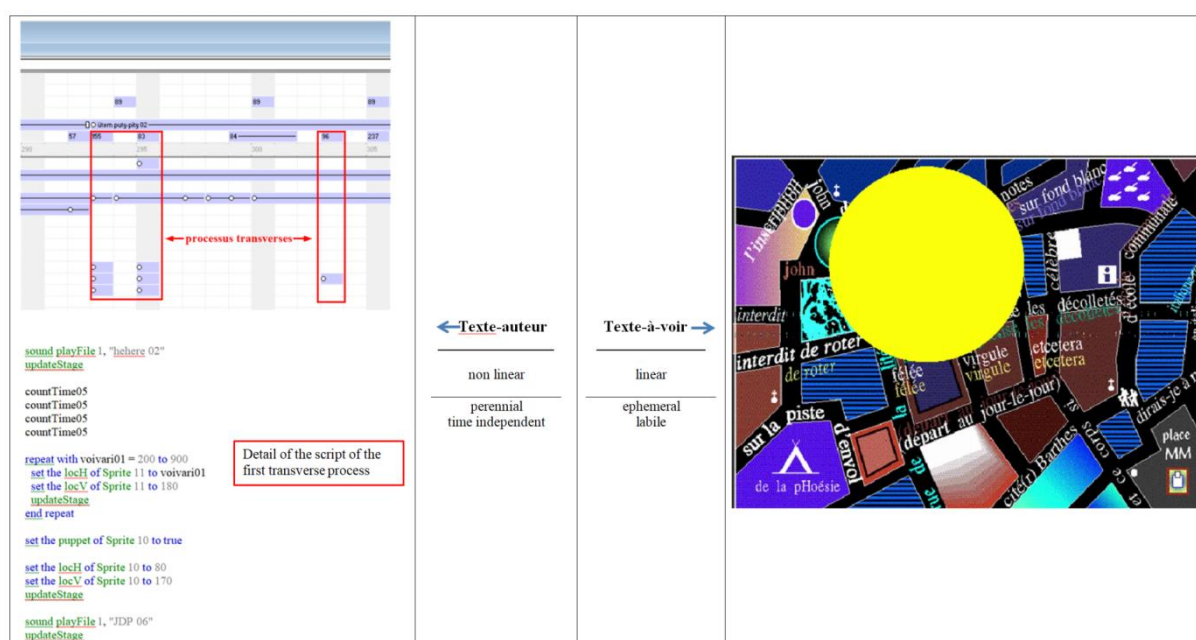


Figure 7: grammatical reinforcement of the proxymoron in an *Orion* sequence

Main pattern and transversal processes are *braids*. By deconstructing the program so that they play alone, the *transitoire observable* correlated to them can be produced and recorded. The corresponding *textes-à-voir* are *snatches* that are found, depending on the case, intact or unstructured in the real *texte-à-voir*<sup>2</sup>.

## b) A pseudo-random sequence

A specific use of the random function in the sequence that tells Orion's story reinforces the proxymoron. In general, Tibor Papp only uses randomness to establish rhythmic variations

<sup>2</sup> You can find a video of the main pattern of the sequence analyzed in figure 7 recorded using this process in <http://bootz.fr/orion/aaaOrionD8501-nuits-sur-carte-fond%20lin%c3%a9aire.mp4> and a video of 1 transversal process in <http://bootz.fr/orion/aaOrionp02-sequence-nuits-sur-carte-processus-frame295.mp4>. The total sequence is recorded in <http://bootz.fr/orion/aaOrionp02-sequence-nuits-sur-carte.mp4>



within the *texte-à-voir*. But he uses it in a more generative way in this sequence to create random bifurcations in the content. The *texte-à-voir* in this sequence then seems generative. The unstructured and erratic nature of this sequence amplifies this generative combinatorial dimension because the main drawback of combinatorial generation is that it produces statements that often flirt with nonsense or, at best, a surrealist image.

The story is told and broken down into 9 moments (noted from 01 to 09 in figure 8). The correct chronological sequence defines the pivot sequence (noted 6). These 9 moments are sequentially rearranged in a non-random way in 8 variants. These variants also use a sound variant (denoted 0102...) of 5 of the 9 sound files in the pivot sequence. In the end, the choice of the played sequence among these 9 possible sequences (numbered from 1 to 9)<sup>3</sup> is random, but the number of variants thus allowed is far from the total number of 11.612.160 sequences that a purely random choice would have allowed.

Figure 8: pseudo-random sequence in *Orion*

### 3.1 Problematics of transcoding

The procedural model views transcoding as more than simply the technical programming of the translated *snatches*.

<sup>3</sup> The sequence 1 is recorded in <http://bootz.fr/orion/sequence-orion1.mp4> and sequence 9 in <http://bootz.fr/orion/sequence-orion9.mp4>

relations and signs of the *texte-à-voir*. There is therefore no reference state that can constitute a single source text to be translated.

In the procedural model, transcoding does not only act on the Reader's *texte-à-voir*, but also on the semiotic relations between the *meta-reader's* *texte-auteur* and *texte-à-voir*, and on the *Author's* *texte-auteur*. The code being the stimulus for the latter, transcoding must ensure its translation. It is then necessary to transcode according to 3 dimensions that may conflict with each other:

- The performative dimension: the new code must produce at runtime the same processes of animation and media transformation in the translated *texte-à-voir* as in the original. It is a performative translation, with no equivalent in print or subtitling.

- The syntactic dimension of the instruction: the new code must reproduce the meaning of each original instruction. This is a "line-by-line" transcoding, equivalent to a literal linguistic transcoding because an instruction is similar to a sentence in a computer code.

- The semantic dimension: transcoding must translate or preserve the *braids* of the *texte-auteur*, independently of any relation with the *texte-à-voir*, i.e. independently of any computer consideration. This dimension concerns the autonomy of the *texte-auteur* and is equivalent to a translation of meaning.

As in linguistic translation, it is better to privilege the semantic dimension over the syntactic dimension in case of conflict. The performative dimension must be ensured in any case.

### 3.2 Non-preservation of the *texte-auteur* in the cybertext model

When transcoding is carried out in accordance with the cybertext model, the translation of the *texte-auteur* is impossible because transcoding, in this model, only preserves the performative dimension of the source program. This can be seen by comparing the *texte-auteur* of the *first screening* original program in basic, created on Apple IIe, and the transcoding into JavaScript by Marko Niemi (2006). This example is exemplary because Marko Niemi has tried, on the one hand, to preserve the temporality of the animation he saw on his screen when executing the program of the work, and on the other hand, to preserve as much as possible the literal dimension of the original program by transcoding the basic instructions by instructions that most resemble them:

*"Some parts of the javascript code are meant to be as directly equivalent to the original BASIC code as possible; for instance, the 'Print(vtab, htab, text)' javascript function works pretty much the same as 'VTAB Y: HTAB X: PRINT "<TEXT>"' lines in BASIC (VTAB and HTAB define the coordinates of the text in question; in javascript, the top-left position of the screen is at (0,0) whereas in BASIC it's at (1,1)).*

*SPEED variables have their equivalents in 'pause()' arrays (SPEED=255 is the fastest tempo in BASIC and SPEED=0 probably the slowest, whereas in javascript the smaller the 'pause', the faster the tempo), and the 'Home()' function clears the screen the same way as the HOME command in BASIC. "* (Niemi, 2006, mail answer to Jim Andrews about the JavaScript version)

Marko Niemi also points out that he used an Apple IIe emulator to compare the *transitoire observable* produced by the original program and his program. As lability is linked to the technical compatibility of the devices, one can hope that the emulator used emulates the Apple IIe well enough to have a low lability. The question of the reference state is then resolved.

We can therefore affirm that Marko Niemi has achieved a good "computer" transcoding, i.e. one that respects the letter and spirit of the instructions, and thus the computer dimension of the program. The comparison of the two programs shows that, unfortunately, the other dimensions of the original *texte-auteur* are not preserved, especially the concrete *snatches*. The JavaScript program modifies the explicit nature of the data: from text to list, which gives a different semiotic vision of the program. It introduces non-linearity with an *if...else* structure as well as an explicit reference to time (with the *setTimeout* method) that do not exist at all in the original program. By numbering the lines<sup>4</sup>, the original program insists on linearity. If it would have been easy to preserve the *snatches*, the predominance of the concept of text over that of data and avoid the *if...else* structure, it would have been at the price of a literal and computer "naive" writing of the JavaScript program. It is understandable that a programmer refuses to do so. On the other hand, it is impossible to avoid calling a JavaScript callback<sup>5</sup> function to manage timers. It is therefore impossible to avoid the non-linearity of the JavaScript code that results from the asynchronous nature of callbacks and the presence of functions.

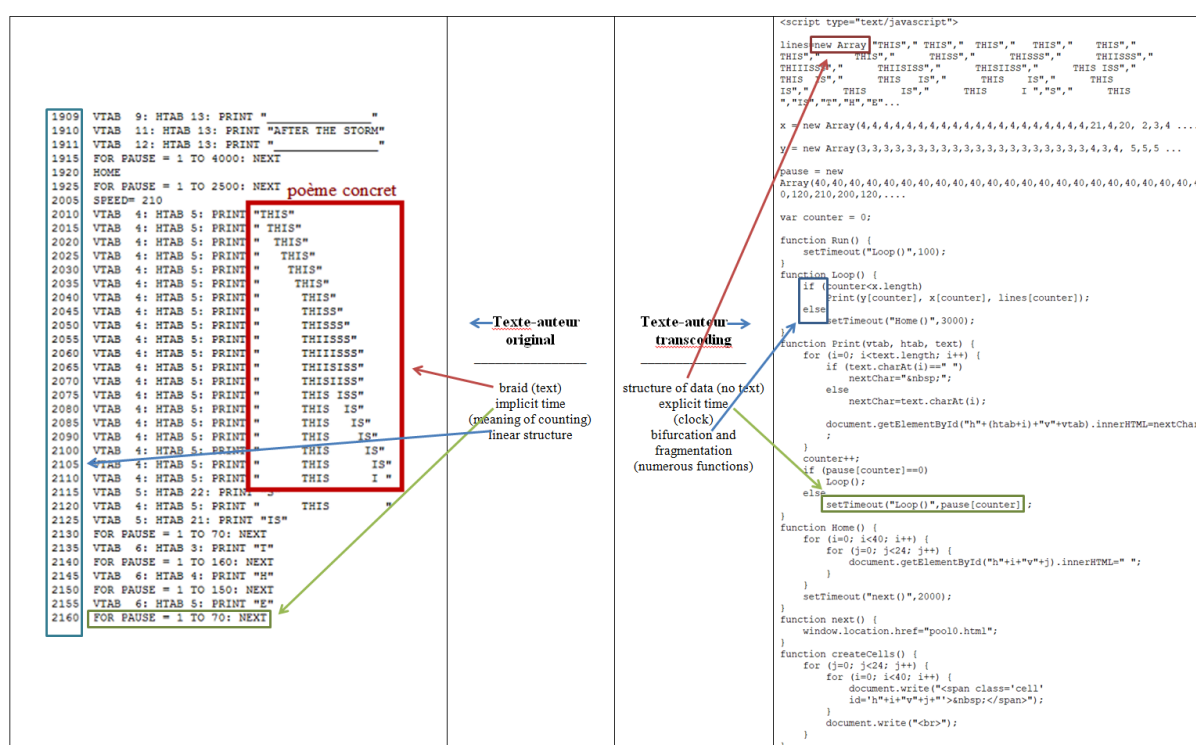


Figure 9: comparison of the 2 first screening programs

Whereas this transposition is not conceived as a translation (no modification of the natural language), we can see how difficult, if not impossible, it is to conceive of transcoding

<sup>4</sup> Irrespective of the fact that this is an inherent constraint of the basic and not of the author, even unintentionally, linearity is emphasized.

<sup>5</sup> Callbacks are functions that are executed at the end of the timer



as a translation operation: literal translation and translation of meaning are simultaneously impossible, as in classical translation, but with consequences that can be assumed to be much more dramatic: a translation that is too far removed from literal translation would risk corrupting the transposition of the *transitoire observable* via the other programming language and thus distorting the translation of the latter. I have also observed, with regard to translations in German and English of a *Reader's texte-à-voir* from the *rabot poète* (Bootz, 2005), that the linguistic semantic translation could impose a modification of the animation script, which profoundly modifies the *text-auteur* built on the source program.

## 4 “Reading machines” as an alternative

Whatever it is (translation, transcoding, transcreation...), transferring a literary digital work poses a difficult challenge when one wants to transpose all the facets of the original into a single target work. One of the solutions is to overemphasize the importance of transcreation and to consider the result produced by this transfer, not as a new version of the work, but as a work in its own right, an adaptation, just as one adapts a novel to the cinema.

The "reading machines" constitute an alternative that makes it possible to stay as close as possible to the *texte-auteur* and *texte-à-voir* of the original production and to preserve the rhetoric of the dual signs. A reading machine, in the procedural model, is a second discourse emanating from *meta-reading* (figure 10). It is a digital or video production that reconstructs for screen reading the dimensions perceived in *meta-reading* that do not appear in *narrow reading*. It is therefore a component that allows the *Reader* to access all the perceptible dimensions of the work. These reading machines can quite possibly contain translations. The reading machine is based on a deconstruction in lexies of all the sign spaces of the work. There is no need to bring them all together in a single object. One can thus perform the linguistic translation(s) of the initial *texte-à-voir* videotapes when possible, or construct a target *transitoire observable* via programs that only "translate" the functional dimension of the original program. They do not even need, then, to constitute a literal translation of the original program. One can even envisage different translations to translate different *snatches* that may overlap. On the other hand, the proxymoron and the *braids* of the *texte-auteur* can be given to read from the original source program. The same applies to ternary signs.

The first reading machines should be installed at BnF on the Richelieu site in 2021.

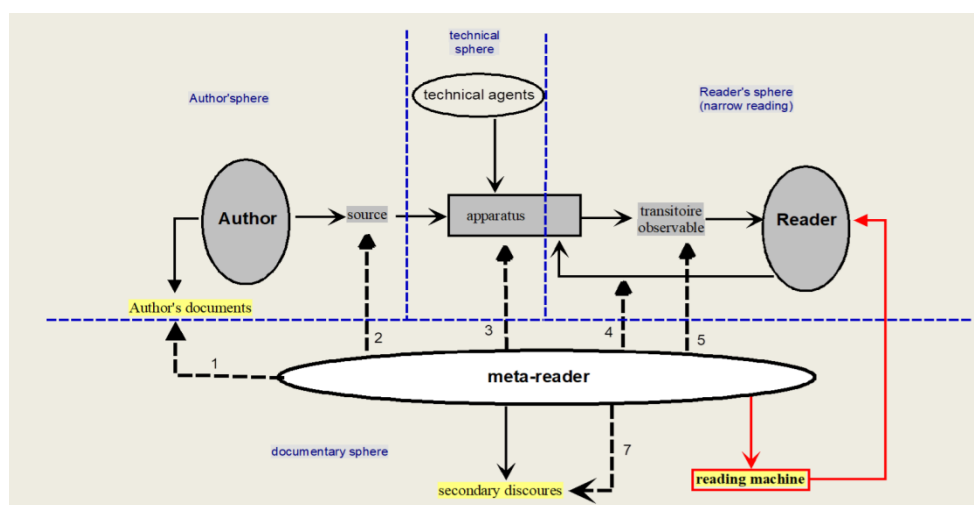


Figure 10 : reading machines in the procedural model

## 5 References

- Aarseth, Espen J. *Cybertext: Perspectives on Ergodic Literature*. Baltimore & London: Johns Hopkins University Press, 1997.
- Baldwin, Sandy. "Ping Poetics", electronic book review, October 2, 2009.  
<https://electronicbookreview.com/essay/ping-poetics/> (consulted on January 04, 2020)
- Barthes, Roland. *S/Z. points*. Paris: Le Seuil, 1970.
- Bootz, Philippe. « Notes ». *alire* livret, n° 3 (juin 1990), p. 11-13.
- Bootz, Philippe. « Un modèle fonctionnel des textes procéduraux ». *Les Cahiers du CIRCAV*, n° 8 (1996), p. 191-216.
- Bootz, Philippe. « Formalisation d'un modèle fonctionnel de communication à l'aide des technologies numériques appliqué à la création poétique ». Thèse de doctorat, Paris 8, 2001.  
<<http://tel.archives-ouvertes.fr/tel-00012165/fr/>> (consulted on January 04, 2020).
- Bootz, Philippe. "der/die leser ; reader/readers", in Block F., Heibach C., Wenz K. (éds), *p0esIs. Asthetik digitaler Poesie, The Aesthetics of Digital Poetry*, Ostfildern : Hatje Cantz Verlag, 2004, p. 93 – 121.
- Bootz, Philippe. *Le rabot poète* (2005). *alire* 13, 2008.
- Bootz, Philippe. *passage 2009*. *alire* 14, 2009.
- Bootz, Philippe, "signs and apparatus in digital poetry: the example of Jean-Marie Dutey's *Le mange-texte*", *LLC/Literary and Linguistic Computing* volume 27, issue 3, September 2012, p. 273-292.
- Bootz, Philippe. « Une approche modélisée de la communication ; Application à la communication par des productions numériques ». Mémoire d'Habilitation à Diriger des Recherches, Paris 8, 2016. <<https://archivesic.ccsd.cnrs.fr/tel-02182750>> (consulted on January 04, 2020).
- bpNichol, *first screening*, Toronto: Underwich Editions, 1984.
- Cayley, John. 2018. 'The Translation of Process'. *Amodern* 8: Translation-Machination, January 2018. <<https://amodern.net/article/the-translation-of-process/>> (consulted on 01/04/2020)
- Dutey, Jean-Marie. *Le Mange-texte*. *alire* 1, 1989.
- Hayles, N. Katherine. *Writing Machines*. Cambridge & London: MIT Press, 2002.
- Klinkenberg, Jean-Marie. *Précis de sémiotique générale*. Bruxelles: De Boeck & Larcier, 1996.
- Marshall, Eric, untitled poem, 3rd International Obfuscated C Code Contest, 1986.  
<https://www.ioccc.org/1986/marshall/marshall.c> (consulted on January 04, 2020). *Copyright (c) 1986, Landon Curt Noll & Larry Bassel. All Rights Reserved. Permission for personal, educational or non-profit use is granted provided this this copyright and notice are included in its entirety and remains unaltered. All other uses must receive prior permission in writing from both Landon Curt Noll and Larry Bassel.*
- Niemi, Marko. JavaScript transcoding of bpNichol's first screening, vispo, 2006  
<https://vispo.com/bp/javascriptversion.htm> (consulted on January 04, 2020)
- Papp, Tibor. *Orion*. *alire* 11, 1999.
- Pold, Søren, Portela, Manuel, and Mencia, Maria, "Electronic Literature Translation: Translation as Process, Experience and Mediation", *Electronic Book Review*, May 30, 2018, <https://doi.org/10.7273/wa3v-ab22> (consulted on January 04, 2020)