

Romances em “trans-ação”. Uma cibertextualidade pós-digital

Sandra Bettencourt

Estudos Avançados em Materialidades da Literatura
Programa de Doutoramento FCT (PD/00334/2012)

Interface e o Futuro da Literatura Eletrônica

while interface does grant access, it also inevitably acts as a kind of magician's cape, continually revealing (mediatic layers, bits of information, etc.) through concealing and concealing as it reveals. (Emerson, 2014: X).

Perhaps, the future of digital literature is *readingwriting* that is born of the network but lives offline — digital literature transformed into book bound *readingwriting* that performs and embodies its own frictional media archeological analysis. (Emerson, 2014: 184)

Trans-ação

a transição do romance de um meio e interface para outro. Isto é, o livro que é atravessado por processos de transcodificação das especificidades da narrativa eletrónica (programação, geração, codificação) para as especificidades da narrativa impressa (fixação do texto na página, progressão narrativa, estabilidade e continuidade textual).

World Clock

```
def fresh(x):
    i = int(len(x)/2 + random() * (len(x) - len(x)/2))
    return [x[i]] + x[:i] + x[i+1:]

now = ['only a moment before', 'almost', 'right about', 'exactly', 'precisely', 'right at', 'as it happens']

negative = ['furtive', 'ramshackle', 'decrepit', 'cookie-cutter', 'undistinguished', 'typical', 'run-down', 'small', 'dim',
'plain', 'ordinary', 'cramped']
shuffle(negative)

positive = ['sturdy', 'charming', 'tidy', 'homey', 'suitable', 'adequate', 'sound', 'orderly', 'nestlike', 'comfortable',
'nice', 'decent']
shuffle(positive)

place = ['habitat', 'accommodation', 'shelter', 'residence', 'domicile', 'abode', 'habitation', 'dwelling', 'structure',
'location', 'house', 'edifice']
shuffle(place)

a_man = ['an individual', 'someone', 'a man', 'a person', 'a youth', 'an old man']

a_woman = ['an individual', 'someone', 'a woman', 'a person', 'a youth', 'an old woman']

named = ['named', 'named', 'named', 'named', 'known as', 'who is called']

male_names = ['Abdallah', 'Abdel-Rahman', 'Abi', 'Abinet', 'Abraham', 'Ahmed', 'Ali', 'Amanual', 'Araya', 'Asfaw',
'Bereket', 'Berhane', 'Bilal', 'Biniam', 'Biruk', 'Daniel', 'Dawit', 'Derege', 'Elias', 'Ephrem', 'Ermias', 'Eyobel', 'Ezra',
'Fasil', 'Gebre', 'Haile', 'Halim', 'Hamza', 'Hanok', 'Hassan', 'Hewan', 'Hussein', 'Ibrahim', 'Karim', 'Khaled', 'Kidus',
'Kirubel', 'Mahmoud', 'Mathios', 'Melak', 'Mohammed', 'Murad', 'Mustafa', 'Nahum', 'Omar', 'Robel', 'Selim',
'Shewit', 'Tadesse', 'Taha', 'Tamiru', 'Tamrat', 'Tareq', 'Teodros', 'Yared', 'Yassin', 'Yonas', 'Yordanos', 'Youssef',
'Zecharias', 'An', 'Bo', 'Cheng', 'De', 'Dong', 'Feng', 'Gang', 'Guo', 'Hui', 'Jian', 'Jie', 'Kang', 'Liang', 'Ning', 'Peng', 'Tao',
'Wei', 'Yong', 'Wen', 'Alexander', 'Sergei', 'Dmitry', 'Andrei', 'Alexey', 'Maxim', 'Evgeny', 'Ivan', 'Mikhail', 'Artyom',
'Bill', 'Jim', 'Nick', 'Darius', 'Leonardo', 'Scott', 'Christian', 'Darren', 'Kenny', 'Brad', 'Zenon']
shuffle(male_names)

female_names = ['Abeba', 'Alem', 'Almaz', 'Ashraquat', 'Aya', 'Azeb', 'Bethlehem', 'Dalal', 'Desta', 'Doha', 'Eden', 'Elsa',
'Fajr', 'Farida', 'Fatima', 'Fatin', 'Fatma', 'Feven', 'Gamalat', 'Gamila', 'Haben', 'Habiba', 'Hasnaa', 'Helen', 'Hosna',
'Hosniya', 'Jerusalem', 'Kedist', 'Leah', 'Lili', 'Luwam', 'Lydia', 'Maha', 'Mahlet', 'Manna', 'Marone', 'Messeret',
'Rahiel', 'Reem', 'Rowan', 'Ruth', 'Saba', 'Sahar', 'Samrawit', 'Sara', 'Senait', 'Shahd', 'Shaimaa', 'Shewit', 'Suha',
'Tigist', 'Tizita', 'Tsega', 'Tsege', 'Yeshi', 'Yohana', 'Zewdy', 'Ai', 'Bi', 'Cai', 'Dan', 'Fang', 'Hong', 'Hui', 'Juan', 'Lan', 'Li',
'Lian', 'Na', 'Ni', 'Qian', 'Qiong', 'Shan', 'Shu', 'Ting', 'Xia', 'Xian', 'Yan', 'Yun', 'Zhen', 'Anastasia', 'Yelena', 'Olga',
'Natalia', 'Yekaterina', 'Anna', 'Tatiana', 'Maria', 'Irina', 'Yulia', 'Jennifer', 'Katie', 'Ali', 'Buffy', 'Christie', 'Stephanie',
'Vanessa', 'Johanna', 'Amaranth', 'Jill', 'Katarzyna']
shuffle(female_names)

size = ['is on the small side', 'is of completely average stature', 'is rather large', 'towers over most people', 'usually
turns to look up to other people', 'is significantly smaller than others of the same age', 'is quite sizable and
imposing', 'is no larger or smaller than one would expect']
shuffle(size)

packaging = ['the ingredient list', 'the warning message', 'some sort of exclamation', 'an entirely made-up word', 'a
tiny numeric code']
```

0

It is now exactly 05:00 in Samarkand. In some ramshackle dwelling a person who is called Gang, who is on the small side, reads an entirely made-up word on a box of breakfast cereal. He turns entirely around.

It is now right about 18:01 in Matamoros. In some dim yet decent structure a man named Tao, who is no larger or smaller than one would expect, reads a tiny numeric code from a recipe clipping. He smiles a tiny smile.

It is now as it happens 19:02 in Grand Turk. In some sturdy yet undistinguished habitat a youth named Peng, who is quite sizable and imposing, reads a stained card. He sits up straight.

It is now only a moment before 02:03 in Windhoek. In some suitable structure someone named Ezra, who is significantly smaller than others of the same age, reads a canary-colored manuscript. He hums quietly.

Megawatt

0x0a news texts thesis contact

```
#### THE VOICES

text.append('\n# I\n\n')
def combine(num, words):
    final = []
    if num > 0 and len(words) >= num:
        if num == 1:
            final = final + [[words[0]]]
        else:
            final = final + [[words[0]] +
                             c for c in combine(num - 1, words[1:])]
        final = final + combine(num, words[1:])
    return final

## In Watt the voices = ['sang', 'cried', 'stated', 'murmured']
## And Watt understood = ['all', 'much', 'little', 'nothing']
## Here the voices did eight things and there are eight levels:
voices = ['sang', 'cried', 'stated', 'murmured', 'babbled', 'chattered',
          'ranted', 'whispered']
understood = ['all', 'most', 'much', 'half', 'little', 'less', 'bits',
              'nothing']

para = ''
preface = ', and sometimes they '
for num in range(len(voices)):
    for word_list in combine(num + 1, voices):
        para = para + preface + ' and '.join(word_list)
        if len(word_list) == 1:
            para = para + ' only'
para = ('Watt heard voices. Now these voices,' + para[5:] +
       ', all together, at the same time, as now, to mention ' +
       'only these ' + spelled_out[len(voices)] + ' kinds of voices, for ' +
       'there were others. And sometimes Watt understood ' +
       ', and sometimes he understood '.join(understood) + ', as now.')
text.append(para)
```

sometimes they cried only, and sometimes they stated only, and sometimes they murmured only, and sometimes they babbled only, and sometimes they chattered only, and sometimes they ranted only, and sometimes they whispered only, and sometimes they sang and cried, and sometimes they sang and stated, and sometimes they sang and murmured, and sometimes they sang and babbled, and sometimes they sang and chattered, and sometimes they sang and ranted, and sometimes they sang and whispered, and sometimes they cried and stated, and sometimes they cried and murmured, and sometimes they cried and babbled, and sometimes they cried and chattered, and sometimes they cried and ranted, and sometimes they cried and whispered, and sometimes they stated and murmured, and sometimes they stated and babbled, and sometimes they stated and chattered, and sometimes they stated and ranted, and sometimes they stated and whispered, and sometimes they murmured and babbled, and sometimes they murmured and chattered, and sometimes they murmured and ranted, and sometimes they murmured and whispered, and sometimes they babbled and chattered, and sometimes they babbled and ranted, and sometimes they babbled and whispered, and sometimes they chattered and ranted, and sometimes they chattered and whispered, and sometimes they ranted and whispered, and sometimes they sang and cried and stated, and sometimes they sang and cried and murmured, and sometimes they sang and cried and babbled, and sometimes they sang and cried and chattered, and sometimes they sang and stated and murmured, and sometimes they sang and stated and babbled, and sometimes they sang and stated and chattered, and sometimes they sang and stated and ranted, and sometimes they sang and stated and whispered, and sometimes they sang and murmured and babbled, and sometimes they

Cibertextualidade Pós-Digital

- Cibertexto

A cybertext is a machine for the production of variety of expression (Aarseth, 1997: 3);

a *textonomy* (the study of textual media) to provide the playing ground of *textology* (the study of textual meaning) (Aarseth, 1997: 15)

- Pós-Digital

an age where, on the one hand, “digital” has become a meaningless attribute because almost all media are electronic and based on digital information processing; and where, on the other hand, younger generation media-critical artists rediscover analog information technology.

(Cramer, 2012)

However, this writing is no longer merely a human activity (a practise of human language), but also the outcome of a software system that is intrinsic to the writing of digital text formats (the practice of code and computation). Our approach is in line with software arts’ call to reflect the tool rather than the medium, as expressed by Kim Cascone.

(Andersen and Pold, 2014b: 2).

a return to the codex in order to rethink the post-digital reader as something other than an «object of capitalist accumulation.»
(Andersen and Pold, 2014b: 14)

that supposedly antiquated device, the book, is fast becoming a safe haven for *readingwriting* because its particulars cannot be tracked, monitored, indexed, fed into an algorithm, and given back to us as a commodity.
(Emerson, 2014: 184)