

Highlight this.

The Future of Listening *with* Audible *and the* NYC Media Lab

The Ear Marks from SVA & Columbia University
Kohzy Koh, Erin McMahon,
Ritwik Deshpande, & David Al-Ibrahim





A Focused Group





Solitary & Passive





Divided & Distracted





Tried & True

Quadrant II Focus. You need a tool that encourages you, motivates you, actually helps you spend the time you need in Quadrant II, so that you're dealing with prevention rather than prioritizing crises. In my opinion, the best way to do this is to organize your life on a weekly basis. You can still adapt and prioritize on a daily basis, but the fundamental thrust is organizing the week.

Organizing on a weekly basis provides much greater balance and context than daily planning. There seems to be implicit cultural recognition of the week as a single, complete unit of time. Business, education, and many other facets of society operate within the framework of the week, designating certain days for focused investment and others for relaxation or inspiration. The basic Judeo-Christian ethic honors the Sabbath, the one day every seven set aside for uplifting purposes.

Most people think in terms

As each generation builds on those that have preceded it, the strengths and some of the tools of each of the first three generations provide elemental material for the fourth. But there is an additional need for a new dimension, for the paradigm and the implementation that will empower us to move into Quadrant II, to become principle-centered and to manage ourselves to do what is truly most important.

THE QUADRANT II Tool

The objective of Quadrant II management is to manage our lives effectively—from a center of sound principles, from a knowledge of our personal mission, with a focus on the important as well as the urgent, and within the framework of maintaining a balance between increasing our production and increasing our capability.

This is, admittedly, an ambitious goal. It's not something that can be achieved in a day or two. It's something that needs to be achieved over time. It's something that needs to be achieved in the thick of thin things in Quadrant I.

A Quadrant II organizer will need

Cohesiveness. Cohesiveness suggests that the various parts of your mission and your production and plans and your personal and spiritual development all fit together. How should be a place

a Gnat-bite. That, at least, is what we can safely say as regards the great majority of the Spiders of our regions.

Nevertheless, a few are to be feared; and foremost among these is the Malmignatte, the terror of the Corsican peasantry. I have seen her settle in the furrows, lay out her web and rush boldly at insects larger than herself; I have admired her garb of black velvet speckled with carmine-red; above all, I have heard most disquieting stories told about her. Around Ajaccio and Bonifacio, her bite is reputed very dangerous, sometimes mortal. The countryman declares this for a fact and the doctor does not always dare deny it. In the neighbourhood of Pujaud, not far from Avignon, the harvesters speak with dread of *Theridion lugubre*, {1} first observed by Léon Dufour in the Catalonian mountains; according to them, her bite would lead to serious accidents. The Italians have bestowed a bad reputation on the Tarantula, who produces convulsions and frenzied dances in the person stung by her. To cope with 'tarantism' the name given to the disease that follows on the bite of the Italian spider, you must have recourse to music, the only effective remedy. The tunes have been noted down in the darkest corners of Calabria. There is medical choreography, medical music. And have we not the tarantella, a lively and nimble dance, bequeathed to us perhaps by the healing art of the Calabrian peasant?

Effective, Engaging & Accessible

in my twenties I found that I could score points with my elders by showing up and speaking respectfully. But then, suddenly—it mattered. My ability to go to a party and speak to anyone about anything, to natter and ask questions, to turn the conversation relentlessly towards the speaker, meant that I was gathering huge amounts of information about other people.

Here's a polite person's trick that I learned and failed me. I will share it with you because I like and respect you, and it is clear to me that you'll know how to apply it wisely: When you are at a party and are thrust into conversation with someone, see how long you can hold off before talking about what they do for a living. And when that painful lull arrives, be the master of it. I have come to revel in that agonizing first pause, because I know that I can push a conversation through. Just ask the other person what they do, and right after they tell you, say: "Wow. That sounds hard."

Because nearly everyone in the world believes their job to be difficult. I once went to a party and met a very beautiful woman whose job was to help celebrities wear Harry Winston jewelry. I could tell that she was disappointed to be introduced to a jowly, dimpled giant in an off-the-shoulder shirt, and I was too. It was difficult to me she had opened and spoke for 30 straight minutes about sapphires and Jessica Simpson. She kept touching me as she talked. I forgave her for that. I didn't reveal a single detail about myself, including my name. Eventually someone pulled me back into the party. The celebrity jewelry coordinator smiled and grabbed my hand. "I like your shirt," she said, and I replied, "I like yours."



The Audio Highlight



<https://vimeo.com/163105137>

Video Demo



```
WhisperSync.py * WhisperHighlight.py * earmarks.py * AudibleSound.py * Kindle.py * WhisperSync.py * WhisperHighlight.py * earmarks.py * AudibleSound.py * Kindle.py *  
1 import pandas as pd  
2 import nltk  
3 import os  
4 from string import punctuation  
5 import json  
6  
7 class kindle_highlights(object):  
8     def __init__(self,__file__,words,book):  
9         json_data = open(__file__).read()  
10        self.highlight_text = json.loads(json_data)  
11        self.loc = {}  
12        self.loc = self.find_highlights(words,book)  
13  
14    def find_h(self,h,words):  
15        indicies = [i for i, x in enumerate(words) if x == h[0]]  
16        l = len(h)  
17        for ind in indicies:  
18            if words[ind:ind+l] == h:  
19                return ind,ind+l  
20        for w in range(l):  
21            try:  
22                int(h[w])  
23                ind1, ind2 = self.find_h(h[0:w-1],words)  
24                return ind1, ind1+l+2  
25            except:  
26                pass  
27        print('Error: Cannot find highlights: '+str(h))  
28        return  
29  
30    def find_highlights(self,book):  
31        for h in self.highlight_text:  
32            h = ''.join(w for w in h if w not in punctuation)  
33            h = h.split()  
34            ind1, ind2 = self.find_h(h,words)  
35            quote_start = book.iloc[ind1]  
36            quote_end = book.iloc[ind2]  
37  
38            start = quote_start['start']  
39            end = quote_end['end']  
40            self.loc[key] = (start,end)  
41            assert len(self.loc) == len(self.highlight_text)  
42            self.loc[key] = (start,end)  
43            assert len(self.loc) == len(self.highlight_text)  
44            return self.loc  
Line 23, Column 25  
1 """WhisperSync. Pulls Amazon Kindle's most commonly highlighted passages/  
2 Usage:  
3     WhisperSync <book>  
4 """  
5  
6 import pandas as pd  
7 import numpy as np  
8 from pydub import AudioSegment  
9 import nltk  
10 import os, sys  
11 from docopt import docopt  
12 from string import punctuation  
13 from Kindle import kindle_highlights  
14 from AudibleSound import sound_overlay  
15  
16 class highlight_sync():  
17     def __init__(self,wsync,mp3,hjson):  
18         self.loc = self.locate_highlight(wsync,hjson)  
19  
20     def import_sync(self,__file__):  
21         book = pd.read_csv(__file__,delimiter=":",header=None)  
22         del book[0]  
23         book.columns = ['text_start','text_end','start','end','word']  
24         return book  
25  
26     def book_text(self,book):  
27         text = "".join(book['word'])  
28         word = ""  
29         ret = ""  
30  
31         for w in text:  
32             word += w  
33             if word in self.loc:  
34                 kh = kindle_highlights(hjson,words,book)  
35                 loc = kh.loc  
36  
37                 for s in range(loc['start'],loc['end']):  
38                     audio = sound_overlay('highlight_sound.wav',volume=3)  
39                     for s in range(loc['start'],loc['end']):  
40                         audio = S0.sound_slice(mp3, audio_loc[s])  
41  
42             ret += word  
43  
44         if name == " main ":  
43 characters selected
```

```
WhisperSync.py * WhisperHighlight.py * earmarks.py * AudibleSound.py * Kindle.py *
```

```
13 from string import punctuation
14 from Kindle import kindle_highlights
15 from sound import sound_overlay
16 from WhisperSync import highlight_sync
17
18 class highlight_find:
19     def __init__(self,mp3,book):
20         self.audio = AudioSegment.from_mp3(mp3)
21         self.book = book
22
23     def hit(self,milli_sec,back=60,forward=2):
24         sec = milli_sec / 1000
25         word_ind = book['start'][((book['start']-sec).abs()).argsort()].index[0]
26         word = book['word'][word_ind].replace(" ","")
27         if word_ind >= back and len(book)-word_ind >=2:
28             text = "".join(book['word'][word_ind-back:word_ind+forward])
29         elif word_ind < back:
30             text = "".join(book['word'][0:word_ind+forward])
31         elif len(book)-word_ind >=2:
32             text = book['word'][len(book)-forward:word_ind]
33         else:
34             text = book['word'][0:word_ind]
35
36         words = word_tokenize(text)
37         tree = nltk.ne_chunk(nltk.pos_tag(words))
38         sent_ = []
39         for i in range(len(tree)):
40             try:
41                 tree[i].label()
42             except AttributeError:
43                 if tree[i][0][0].isupper() and tree[i][0] != word and tree[i][1] != 'NNP':
44                     sent_.append((i,tree[i][0]))
45                 elif tree[i][0] == word:
46                     mid = i
47         sent_end = min(sent_,key=lambda x:abs(x[0]-mid))
48         sent_start = sent_[sent_.index(sent_end)-1]
49         for i in range(word_ind-sent_start[0]-5,word_ind):
50
51
52
53
54
55
56         clip_end = i
```

```
my_highlights.json UNREGISTERED

my_highlights.json

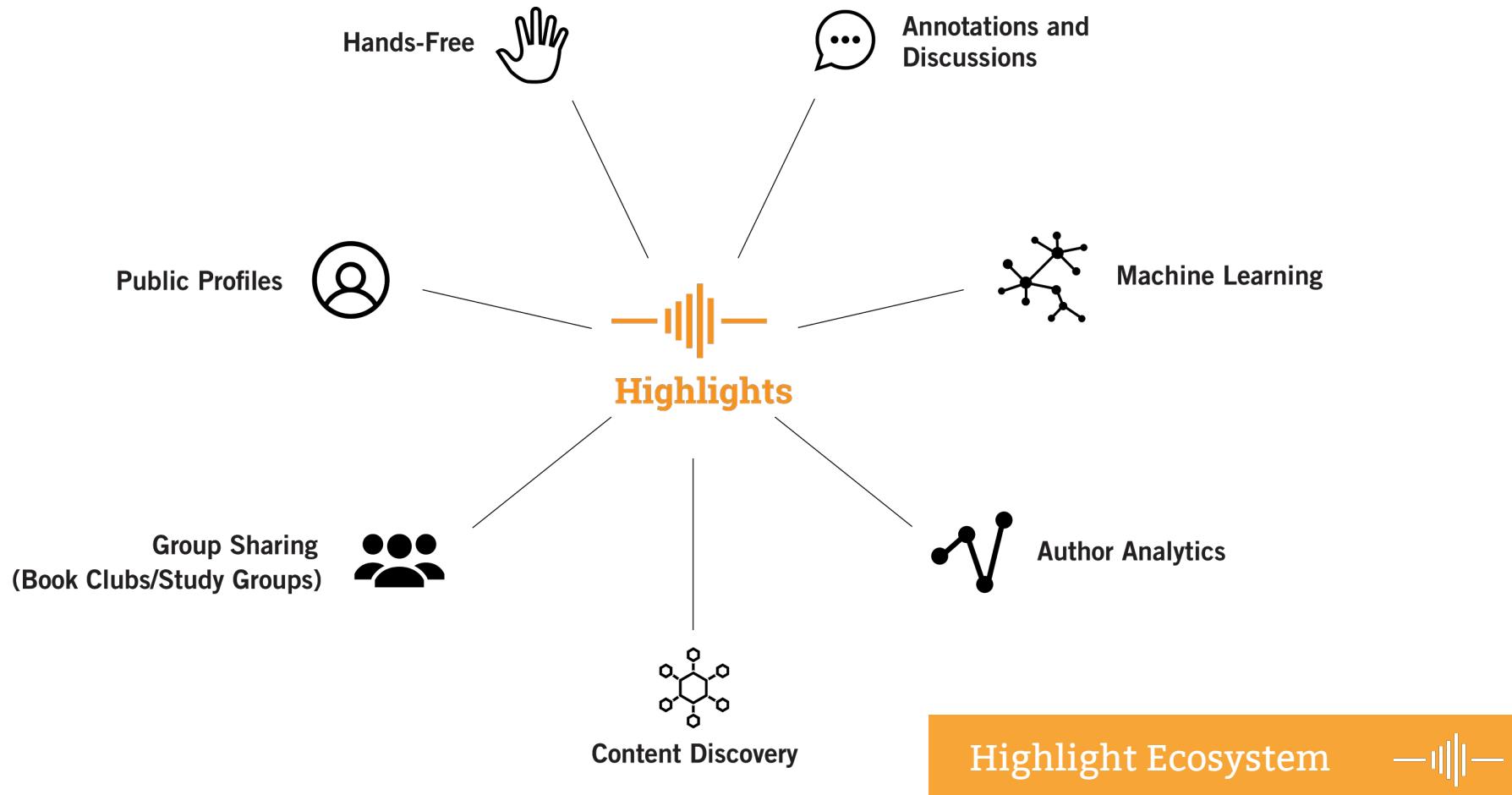
1  {
2      "2809": {
3          "loc": [1223.242, 1227.501],
4          "text": "...mannerist she wrote.  
HE EXCELS MOST IN THE  
DELINEATION OF DEEP FEELING.  
Upon seeing..."
5      },
6      "1799": {
7          "loc": [828.712, 839.0110000000001],
8          "text": "...tutored in mathematics  
.THE COMBINATION PRODUCED IN  
ADA A LOVE FOR WHAT SHE TOOK  
TO CALLING POETICAL SCIENCE  
WHICH LINKED HER REBELLIOUS  
IMAGINATION TO HER  
ENCHANTMENT WITH NUMBERS. For  
many..."
9      }
10 }
```

Generates Text

Line 10- Column 2

Tab Size: 4

ISON





Walter Isaacson



The Innovators

Published: 2014

Audio Released: 2014

Length: 17 hours 28 min

Filtered by:

USA Listeners



TOP HIGHLIGHTS

Such episodes gave Hopper the reputation of being irreverent. But she also had a software hacker's ability to combine irreverence ...

Chapter 3 768 highlights

When the sales guys run the company, the product guys don't matter so much, and a lot of them just turn off," Jobs said.

Chapter 12 560 highlights

"Money is not the greatest of motivators," Torvalds said. "Folks do their best work when they are driven by passion."

After marrying a professor of comparative literature, Vincent Hopper, Grace joined the faculty of Vassar. Unlike most math professors

Author Analytics



••• Sketch 9:41 AM 100%

Home

Katherine Tsao 1m
Well about six weeks ago, she heard the name Gatsby for the first time in years. It was when...
 The Great Gatsby 37s

Oprah Winfrey 3m
The Mark II version of the Harvard computer was in a building without window screens.
 The Innovators 18s

Nishal Gupta 25m
Good writers are avid readers. They have absorbed a vast inventory of words, idioms, ...
 The Sense of Style 26s

Trey Alvarez 42m
"Sir Gawain," Axl said finally. "We look to you, sir. Let's keep no more disguises between us..."
 The Buried Giant 33s

Home Book Clubs Me

••• Sketch 9:41 AM 100%

Breakfast Book Club

MEMBERS

The Innovators
Walter Isaacson
 17 hours 28 min

Nishal 26s
He absorbed their ideas, took credit for some, but also made it clear that nobody should claim...
 Serena This was a really interesting point! Can we discuss at meeting?
Carla I don't necessarily agree with how Isaacson portrays it here.

Katherine
The Mark II version of the Harvard computer was in a building without window screens.

Home Book Clubs Me

••• Sketch 9:41 AM 100%

Oprah Winfrey

78k followers 41 books 2k highlights

The Innovators 25m
The Mark II version of the Harvard computer was in a building without window screens.
 26s

The Daily Wrap 1 day
Good writers are avid readers. They have absorbed a vast inventory of words, idioms, ...
 13s

The Buried Giant 3 days
"Sir Gawain," Axl said finally. "We

Community Book Clubs Me

