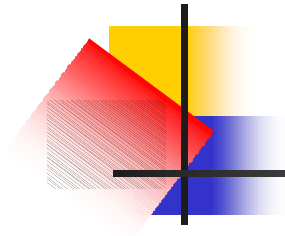
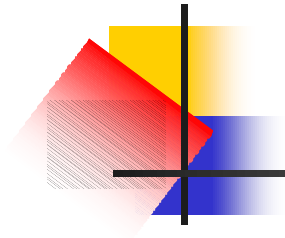


Burrows Wheeler Transform

- Difficult problems can often be simplified by performing a transformation on a data set.
- The Burrows-Wheeler Transform, or BWT, transforms a block of data into a format that is extremely well suited for compression.
- The BWT is an algorithm that takes a block of data and rearranges it using a sorting algorithm.
- The transformation is reversible, meaning the original ordering of the data elements can be restored with no loss of fidelity.

© Klein S. T. and Wiseman Y.



Example

s0	y	a	i	r	w	i	s	e	m	a	n
s1	a	i	r	w	i	s	e	m	a	n	y
s2	i	r	w	i	s	e	m	a	n	y	a
s3	r	w	i	s	e	m	a	n	y	a	i
s4	w	i	s	e	m	a	n	y	a	i	r
s5	i	s	e	m	a	n	y	a	i	r	w
s6	s	e	m	a	n	y	a	i	r	w	i
s7	e	m	a	n	y	a	i	r	w	i	s
s8	m	a	n	y	a	i	r	w	i	s	e
s9	a	n	y	a	i	r	w	i	s	e	m
s10	n	y	a	i	r	w	i	s	e	m	a

© Klein S. T. and Wiseman Y.

Example (cont.)

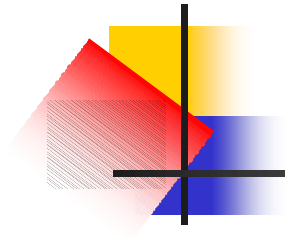
	F										L
s1	a	i	r	w	i	s	e	m	a	n	y
s9	a	n	y	a	i	r	w	i	s	e	m
s7	e	m	a	n	y	a	i	r	w	i	s
s2	i	r	w	i	s	e	m	a	n	y	a
s5	i	s	e	m	a	n	y	a	i	r	w
s8	m	a	n	y	a	i	r	w	i	s	e
s10	n	y	a	i	r	w	i	s	e	m	a
s3	r	w	i	s	e	m	a	n	y	a	i
s6	s	e	m	a	n	y	a	i	r	w	i
s4	w	i	s	e	m	a	n	y	a	i	r
s0	y	a	i	r	w	i	s	e	m	a	n

© Klein S. T. and Wiseman Y.

How to decode

L	F									
y	a	?	?	?	?	?	?	?	?	?
m	a	?	?	?	?	?	?	?	?	?
s	e	?	?	?	?	?	?	?	?	?
a	i	?	?	?	?	?	?	?	?	?
w	i	?	?	?	?	?	?	?	?	?
e	m	?	?	?	?	?	?	?	?	?
a	n	?	?	?	?	?	?	?	?	?
i	r	?	?	?	?	?	?	?	?	?
i	s	?	?	?	?	?	?	?	?	?
r	w	?	?	?	?	?	?	?	?	?
n	y	?	?	?	?	?	?	?	?	?

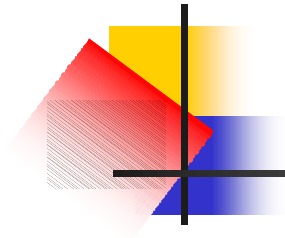
© Klein S. T. and Wiseman Y.



Snippet of a sorted data

- g ood and evil, thou shalt not eat of it: for in th
- g ood and evil. And a river went out of Eden to wat
- g ood and evil. And when the woman saw that the tre
- g ood and evil: and now, lest he put forth his hand
- l ood crieth unto me from the ground. And now art t
- g ood for food, and that it was pleasant to the eye
- g ood for food; the tree of life also in the midst
- l ood from thy hand; When thou tillest the ground,
- g ood that the man should be alone; I will make him
- f ood, and that it was pleasant to the eyes, and a
- g ood. And God blessed them, saying, Be fruitful, a
- g ood. And God said, Let the earth bring forth gras
- g ood. And God said, Let us make man in our image,
- g ood. And the evening and the morning were the fou
- g ood. And the evening and the morning were the six
- g ood. And the evening and the morning were the thi
- g ood: and God divided the light from the darkness.
- g ood: there is bdellium and the onyx stone. And th
- f ood; the tree of life also in the midst of the ga

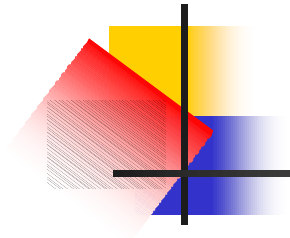
© Klein S. T. and Wiseman Y.



Move to Front

- A Move to Front encoder keeps all 256 possible codes in a list.
- Each time a character is to be output, the encoder:
  - sends its position in the list.
  - moves it to the front.
- g g g g l g g l g f g g g g g g g g f
  will be
  103,0,0,0,108,1,0,1,1,104,1,0,0,0,0,0,0,0,1
  (ASCII codes of f,g,l are 102,103,108 respectively)

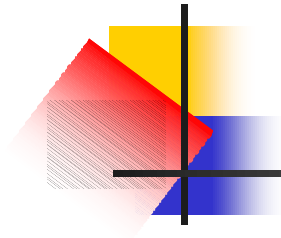
© Klein S. T. and Wiseman Y.



Final Step

- If the output of the BWT operation contains a high proportion of repeated characters, we can expect that applying the Move To Front encoder will give us a file:
  - filled with lots of zeros
  - heavily weighted towards the lowest integers.
- At that point, the file can finally be compressed using an entropy encoder, typically a Huffman or arithmetic encoder.

© Klein S. T. and Wiseman Y.

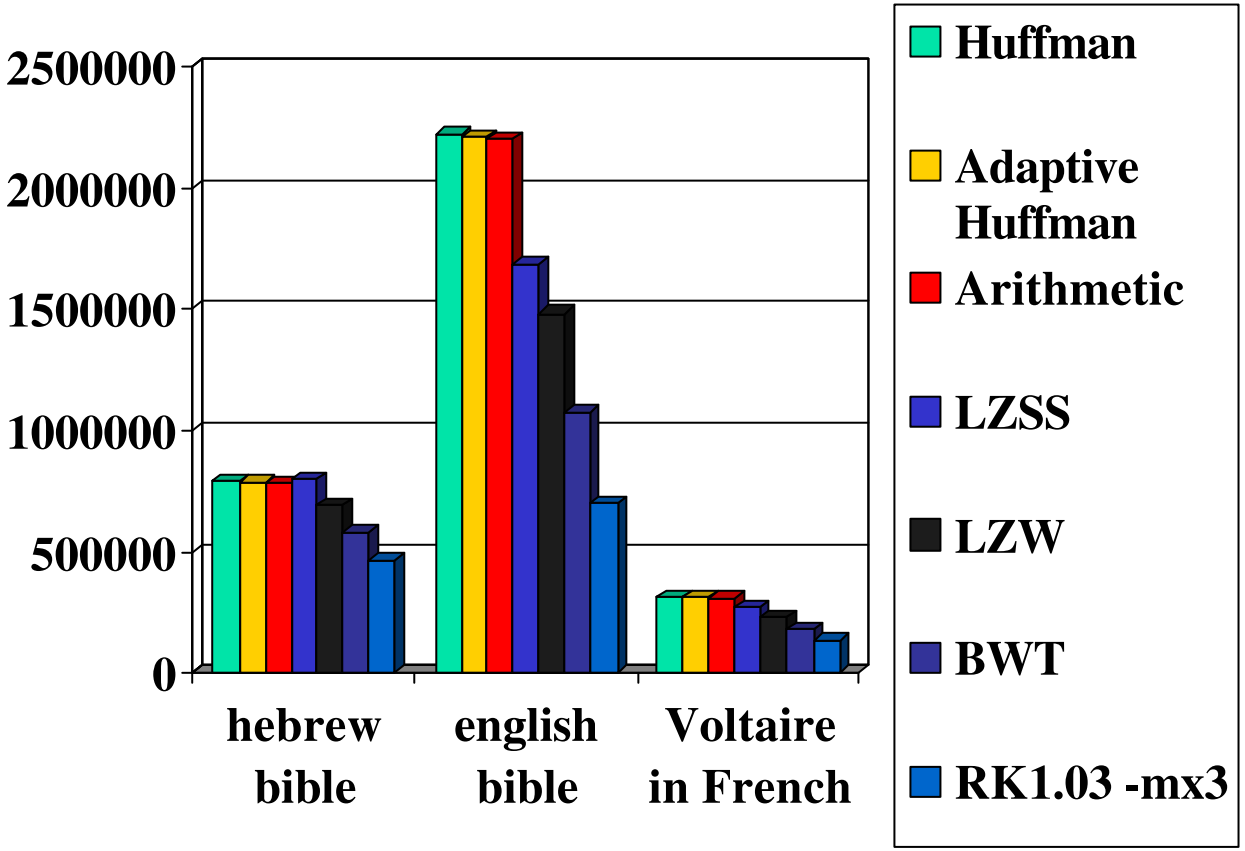


Disadvantages

- The BWT is performed on an entire block of data at once.
  - Most of today's familiar lossless compression algorithms operate in streaming mode, reading a single byte or a few bytes at a time. But with this new transform, we want to operate on the largest chunks of data possible.
  - Sometimes, the file must be split up and processed a block at a time.
- The sorting operations needed by BWT will have O(nlogn) performance, meaning bigger blocks might slow things down considerably.

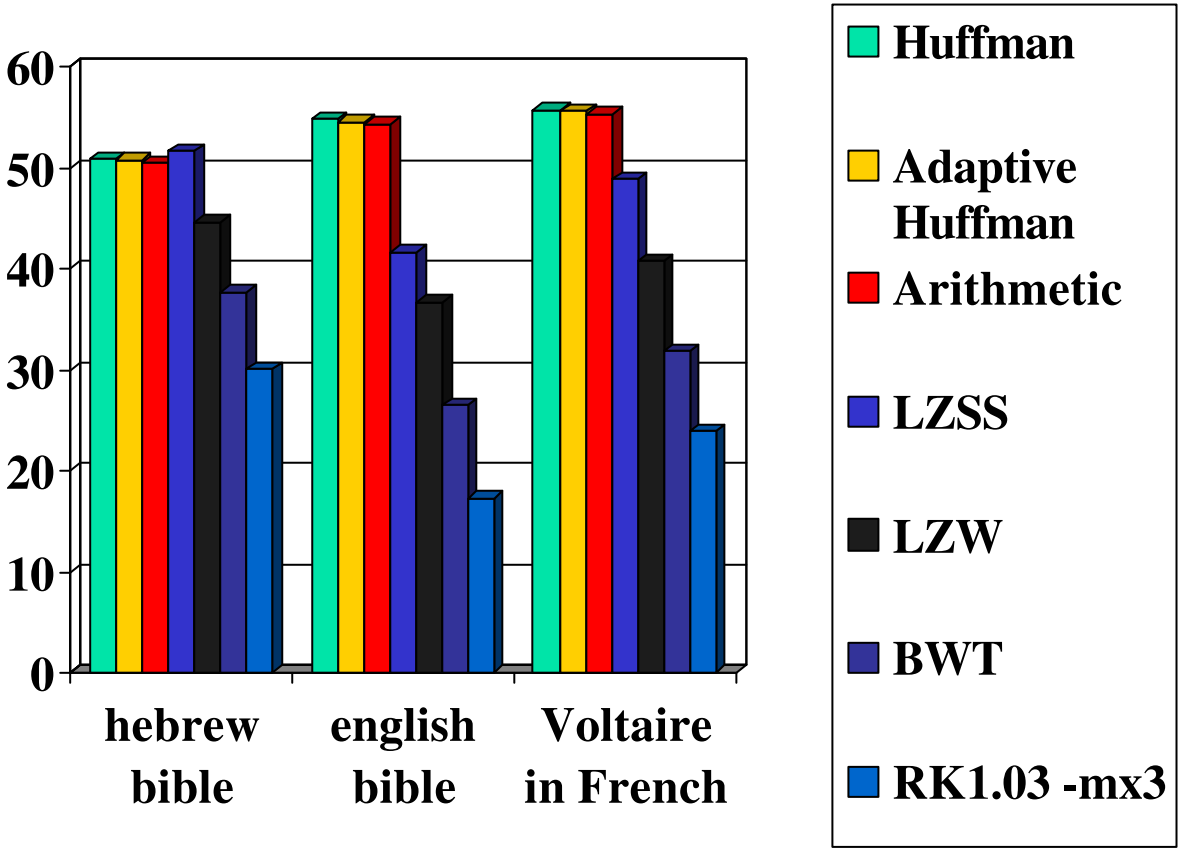
© Klein S. T. and Wiseman Y.

Comparison of methods



© Klein S. T. and Wiseman Y.

Comparison in percents



© Klein S. T. and Wiseman Y.