# Datalogiforelæsning

Gerth Stølting Brodal

Datalogi Introdage, Institut for Datalogi, Aarhus Universitet, 21. august 2024

May 10, 1994

SRC Research Report 124

# A Block-sorting Lossless Data Compression Algorithm

M. Burrows and D.J. Wheeler

digital

**Systems Research Center**
130 Lytton Avenue
Palo Alto, California 94301

May 10, 1994

SRC Research Report

124

**A Block-sorting Lossless Data Compression Algorithm**

M. Burrows and D.J. Wheeler

digital

Systems Research Center
130 Lytton Avenue
Palo Alto, California 94301

# Burrows - Wheeler transformation

Michael Burrows

David J. Wheeler

Opfundet af Wheeler (1927-2004) in 1978

Burrows var ph.d. studerende af Wheeler @ Cambridge

Teknisk rapport 1994

”Annual Data Compression Conference”
afviste at publicere resultatet

Burrows: 2022 ACM Paris Kanellakis Theory and Practice Award

# Det var åbenbart en god idé…

DIMACS

*Center for Discrete Mathematics & Theoretical Computer Science*
*Founded as a National Science Foundation Science and Technology Center*

**DIMACS Working Group on The Burrows - Wheeler Transform: Ten Years Later**

**August 19 - 20, 2004**
**DIMACS Center, CoRE Building, Rutgers University, Piscataway, NJ**

**Organizers:**
**Paolo Ferragina**, University of Pisa
**Giovanni Manzini**, University of Piemonte Orientale
**S. Muthukrishnan**, Rutgers University, muthu@cs.rutgers.edu

Presented under the auspices of the Special Focus on Special Focus on Data Analysis and Mining.

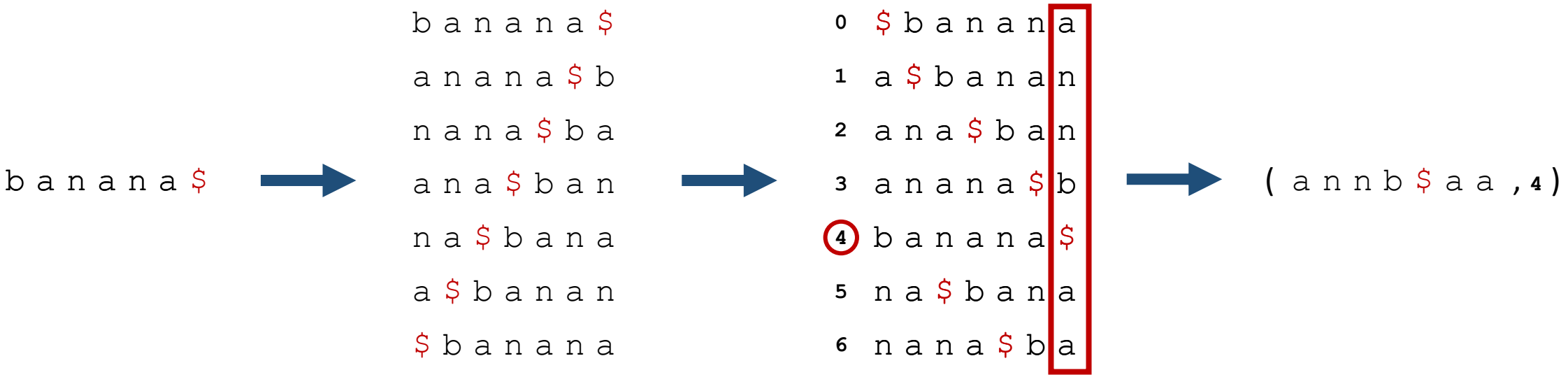
Report from Dagstuhl Seminar 19241

25 Years of the Burrows-Wheeler Transform

Edited by
Travis Gagie[1], Giovanni Manzini[2], Gonzalo Navarro[3], and Jens Stoye[4]

Tidsskriftsudgave dedikeret til BWT
doi.org/10.1016/j.tcs.2007.07.011

# Burrows-Wheeler transformation




## Input

En streng, som ender på et tegn $, der ellers ikke forekommer og er mindre end alle andre tegn ($ < a < b < n)

## Trin 1

Konstruer alle cykliske rotationer/skift af strengen

## Trin 2

Sorter alle rotationerne alfabetisk/ leksikografisk

## Output

Den sidste kolonne som en streng + indeks på rækken der indeholder input strengen

# Burrows-Wheeler transformantionen af **batman$** ?

a) (bmn$taa,3)

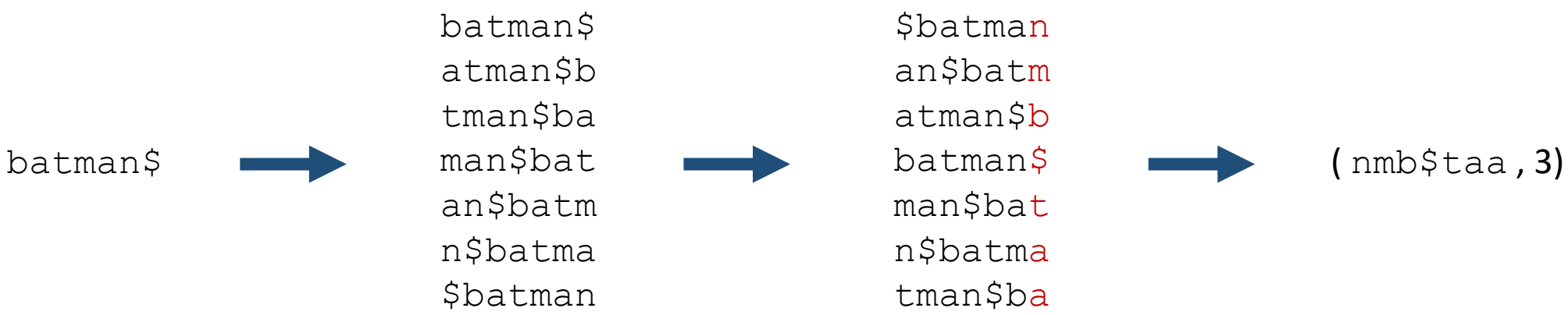
b) ($batman,0)

 c) (nmb$taa,3)

d) ($aabmnt,0)

e) (nmb$aat,3)

**www.menti.com**
**6188 6358**



# Tre oplagte spørgsmål

- Kan transformation inverteres ?

`(nmb$taa,3)` → `batman$`

- Kan transformationerne beregnes effektivt ?

`somelongstring$` → 

```
$somelongstring
elongstring$som
g$somelongstrin
gstring$somelon
ing$somelongstr
longstring$some
melongstring$so
ng$somelongstri
ngstring$somelo
omelongstring$s
ongstring$somel
ring$somelongst
somelongstring$
string$somelong
tring$somelongs
```

1.000.000 tegn

midlertidigt
1.000.000.000.000 tegn
men output kun
1.000.000 tegn

- Kan transformationen overhovedet bruges til noget ?

Google scholar giver +2500 citationer

# Inverse Burrows-Wheeler transformation

Hvilket index får man fra Burrows-Wheeler transformationen?

(itd$ogala,_)

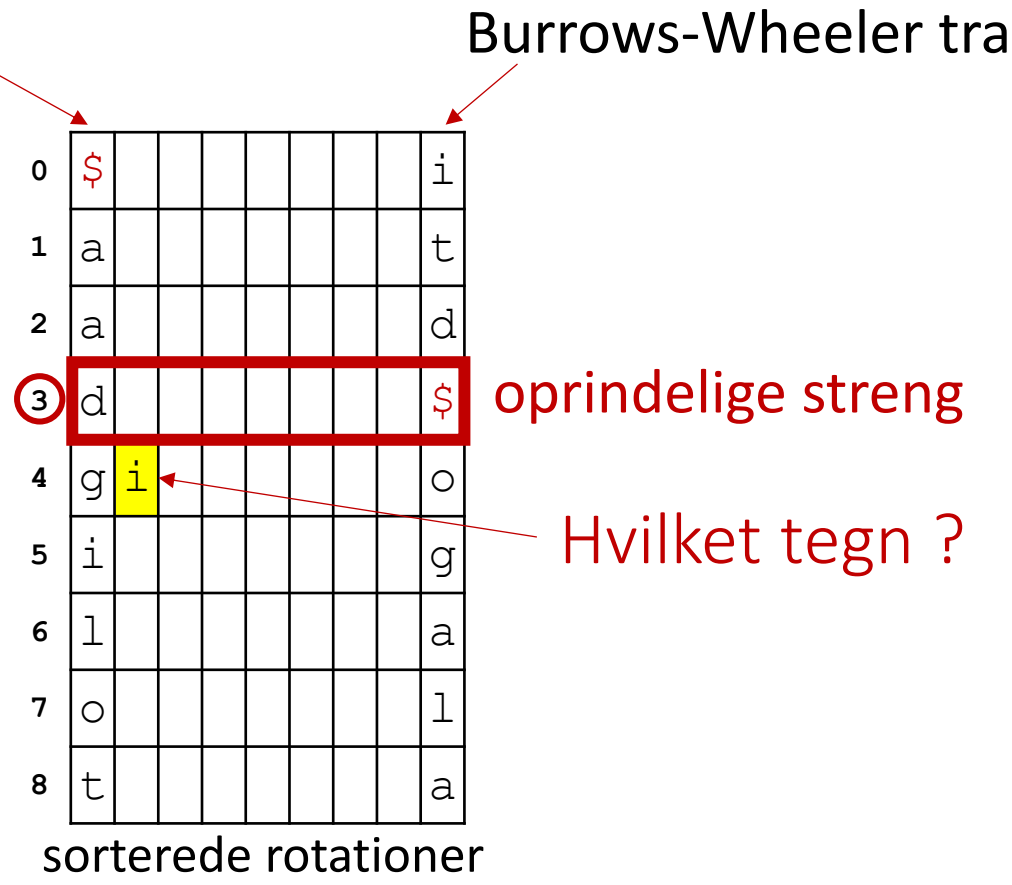
a) 0
b) 1
c) 2

d) 3 = indekset på $ i Burrows-Wheeler transformationen
e) 4
f) 5
g) 6
h) 7
i) 8

**Bemærk**
Tilstrækkeligt enten at tilføje $ til sidst i strengen eller gemme et indeks

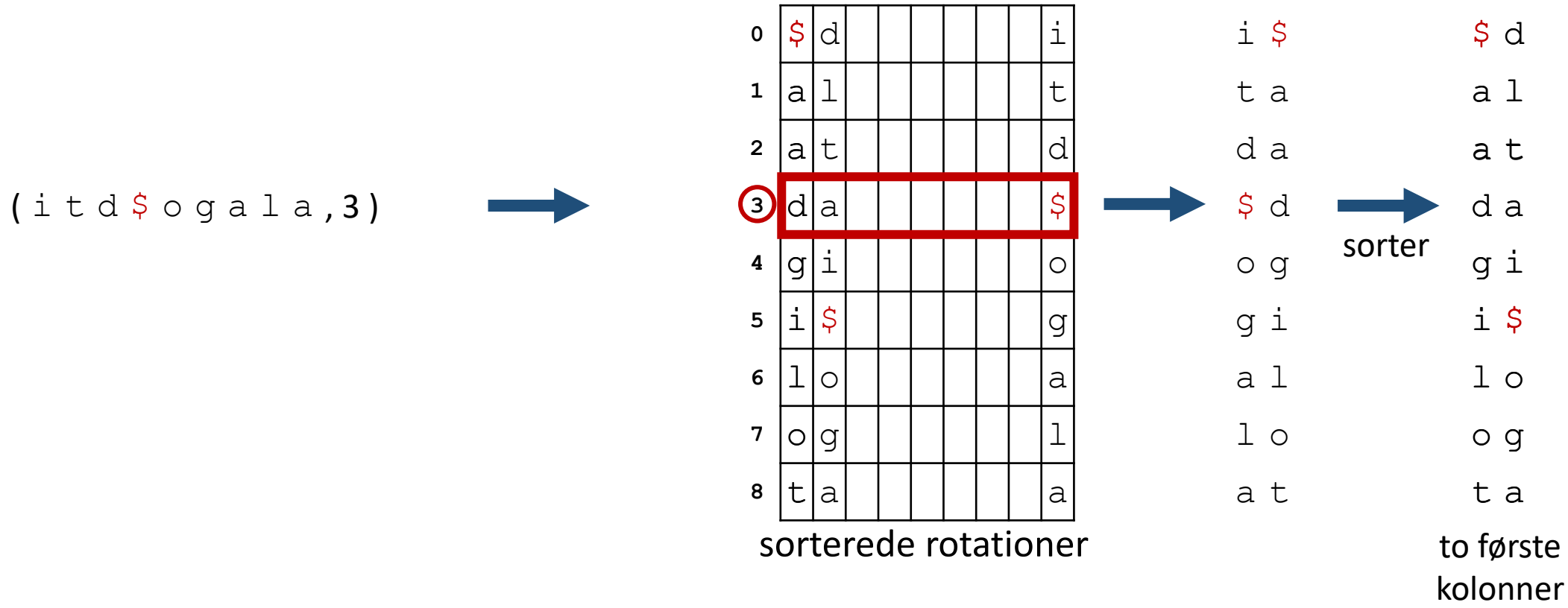
# Inverse Burrows-Wheeler transformation

(itd$ogala,3)

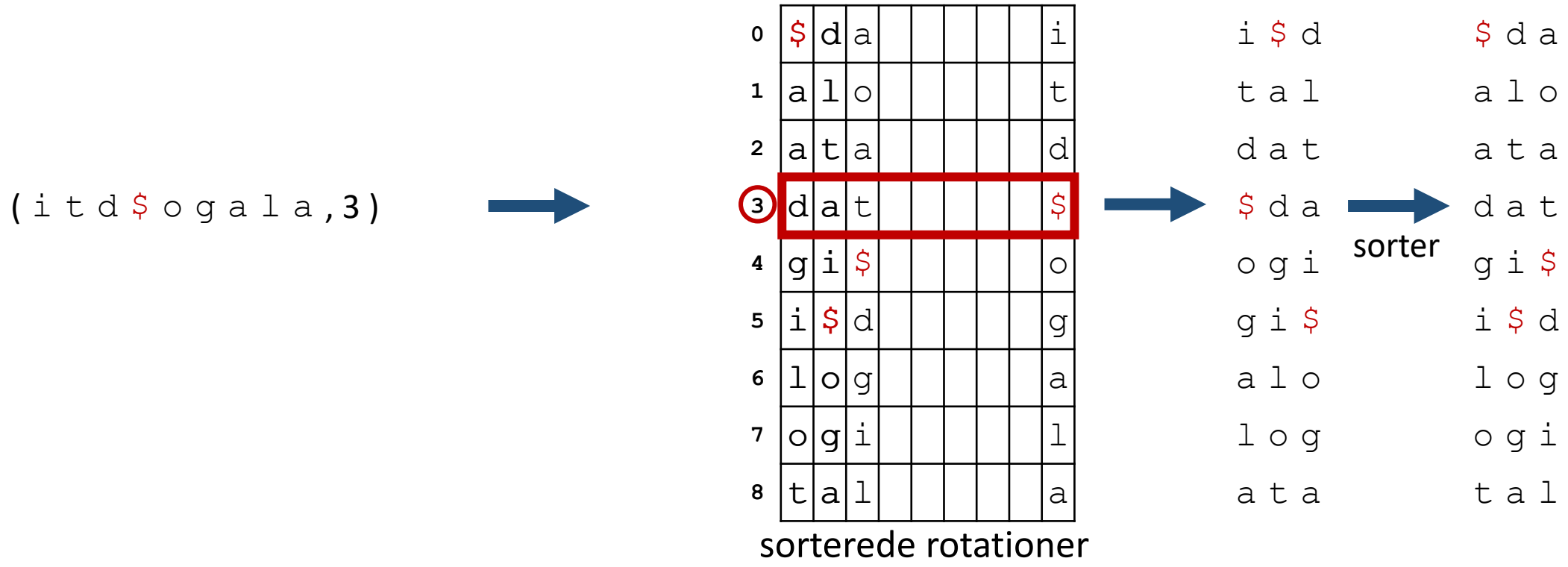



a) $
b) a
c) d
d) g
e) i
f) l
g) o
h) t

# Inverse Burrows-Wheeler transformation



# Inverse Burrows-Wheeler transformation



# Inverse Burrows-Wheeler transformation

(itd$ogala,3)

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 0 | $ | d | a | t | a | l | o | g | i |
| 1 | a | l | o | g | i | $ | d | a | t |
| 2 | a | t | a | l | o | g | i | $ | d |
| 3 | d | a | t | a | l | o | g | i | $ |
| 4 | g | i | $ | d | a | t | a | l | o |
| 5 | i | $ | d | a | t | a | l | o | g |
| 6 | l | o | g | i | $ | d | a | t | a |
| 7 | o | g | i | $ | d | a | t | a | l |
| 8 | t | a | l | o | g | i | $ | d | a |

sorterede rotationer

datalogi$

# Invers Burrows-Wheeler transformation

**Sætning**

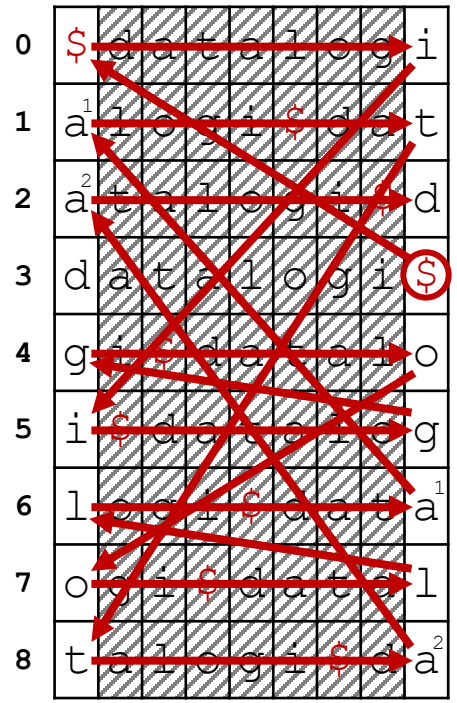
Man kan fra Burrows-Wheeler transformationen af en streng rekonstruere den oprindelige streng

(itd$ogala,3) → datalogi$

# Hurtigere inverse Burrows-Wheeler transformation

**Idé** – konstruer kun første og sidste kolonne

(itd$ogala,3)

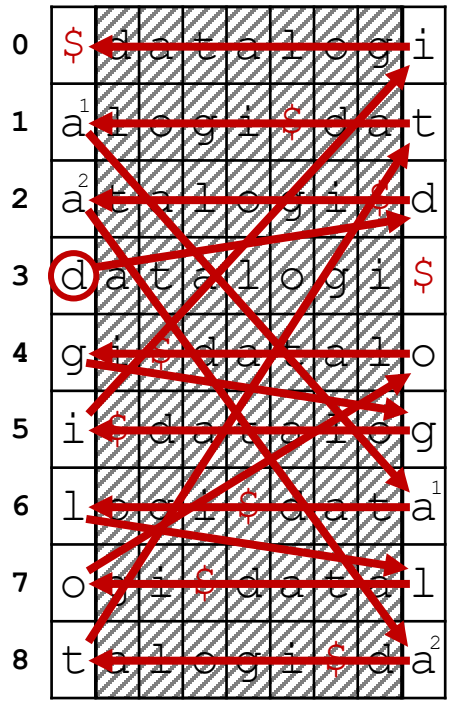



datalogi$

# Hurtigere inverse Burrows-Wheeler transformation

**Idé** – konstruer kun første og sidste kolonne

(itd$ogala,3)




d a t a l o g i $

# Python implementation

```
def BurrowsWheeler(text):
    S = sorted(text[i:] + text[:i] for i in range(len(text)))
    idx = S.index(text)
    bwt = ''.join(x[-1] for x in S)
    return (bwt, idx)

def InverseBurrowsWheeler(bwt, idx):
    S = [''] * len(bwt)
    for _ in range(len(bwt)):
        S = sorted(c + s for s, c in zip(S, bwt))
    return S[idx]

def EfficientInverseBurrowsWheeler(bwt, idx):
    r = []
    cnt = {}
    for char in bwt:
        r.append(cnt.get(char, 0))
        cnt[char] = cnt.get(char, 0) + 1
    s = 0
    for char in sorted(cnt):
        s, cnt[char] = s + cnt[char], s
    S = []
    for _ in range(len(bwt)):
        S.append(bwt[idx])
        idx = cnt[bwt[idx]] + r[idx]
    return ''.join(reversed(S))
```

# Den Grimme Ælling (Hans Christian Andersen, 1843)

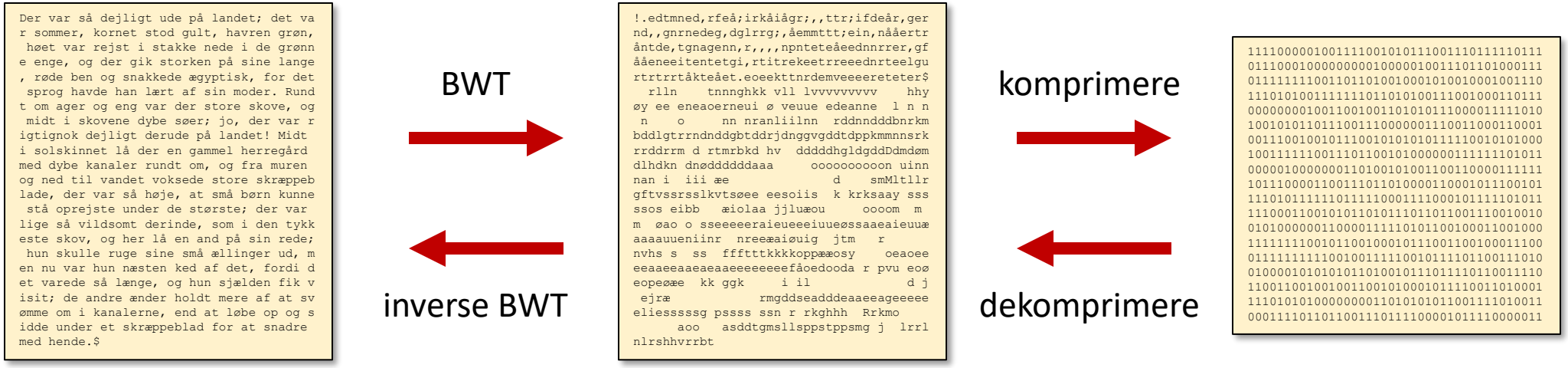
```
Der var så dejligt ude på landet; det var sommer, kornet stod gult, havren grøn, høet var rejst i st
akke nede i de grønne enge, og der gik storken på sine lange, røde be
 sprog havde han lært af sin moder. Rundt om ager og eng var der stor
 søer; jo, der var rigtignok dejligt derude på landet! Midt i solskin
med dybe kanaler rundt om, og fra muren og ned til vandet voksede st
e, at små børn kunne stå oprejste under de største; der var lige så v
este skov, og her lå en and på sin rede; hun skulle ruge sine små æll
 ked af det, fordi det varede så længe, og hun sjælden fik visit; de
ømme om i kanalerne, end at løbe op og sidde under et skræppeblad for
```

Burrows-Wheeler transformation

```
!.edtmned,rfeå;irkåiågr;,,ttr;ifdeår,gernd,,gnrnedeg,dglrrg;,åemmttt,
npnteteåeednnrrer,gfååeneeitentetgi,rtitrekeetrreednrteelgurtrtrrtål
  rlln    tnnnghkk vll lvvvvvvvvv    hhyøy ee eneaoerneui ø veuue ede
iilnn  rddnndddbnrkmbddlgtrrndnddgbtddrjdnggvgddtdppkmmnnsrkrrddrrm c
dlhdkn dnøddddddaaa     ooooooooon uinnnan i  iii æe                d
soiis  k krksaay sssssos eibb   æiolaa jjluæou     oooom  m m  øao o
aaaauueniinr  nreeæaiøuig  jtm   r      nvhs s  ss  fffttkkkkoppææos
efåoedooda r pvu eoøeopeøæe  kk ggk     i il             d j ejræ
eliesssssg pssss ssn r rkghhh  Rrkmo          aoo   asddtgmsllsppstpp
```

```
416 f at svømme om i kan...ldt mere a
417 f det, fordi det var...sten ked a
418 f sin moder. Rundt o...han lært a
419 fik visit; de andre ...n sjælden
420 for at snadre med he...ræppeblad
421 for det sprog havde ...ægyptisk,
422 fordi det varede så ...d af det,
423 fra muren og ned til...dt om, og
424 g der gik storken på...ne enge, o
425 g eng var der store ... om ager o
426 g fra muren og ned t...undt om, o
427 g havde han lært af ...r det spro
428 g her lå en and på s...te skov, o
429 g hun sjælden fik vi...å længe, o
430 g midt i skovene dyb...e skove, o
431 g ned til vandet vok...ra muren o
432 g sidde under et skr... løbe op o
433 g snakkede ægyptisk,...røde ben o
434 g var der store skov...ager og en
435 gammel herregård med...lå der en
436 ge sine små ællinger... skulle ru
437 ge så vildsomt derin...der var li
438 ge, og der gik stork... grønne en
439 ge, og hun sjælden f...ede så læn
440 ge, røde ben og snak...å sine lan
```

# BWT kan forbedre eksisterende tabsfri komprimerings metoder




original dokument

Burrows-Wheeler transform

komprimeret (mindre)

Eksempel: bzip2

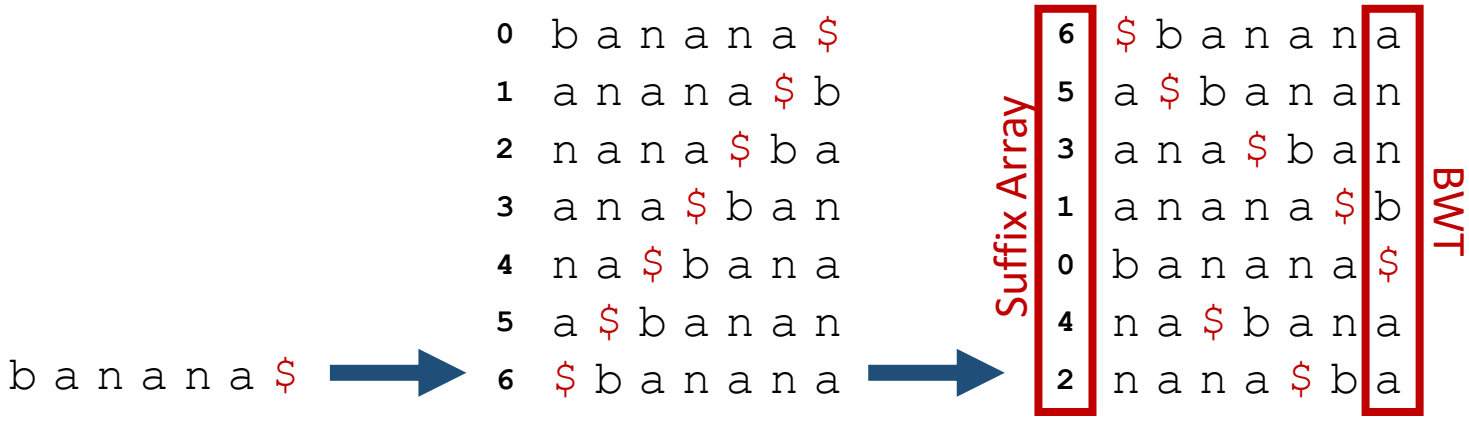
(BWT + run-length encoding + move-to-front + Huffman codes + delta encoding + …)

# Mønstergenkendelse (FM-index)

Givet en streng, byg en **søgestruktur** så vi effektivt kan finde alle forekomster af en delstreng (mønster)

```
Der var så dejligt ude på landet; det var sommer, kornet stod gult, havren grøn, høet var rejst i st
akke nede i de grønne enge, og der gik storken på sine lange, røde ben og snakkede ægyptisk, for det
 sprog havde han lært af sin moder. Rundt om ager og eng var der store skove, og midt i skovene dybe
 søer; jo, der var rigtignok dejligt derude på landet! Midt i solskinnet lå der en gammel herregård
med dybe kanaler rundt om, og fra muren og ned til vandet voksede store skræppeblade, der var så høj
e, at små børn kunne stå oprejste under de største; der var lige så vildsomt derinde, som i den tykk
este skov, og her lå en and på sin rede; hun skulle ruge sine små ællinger ud, men nu var hun næsten
 ked af det, fordi det varede så længe, og hun sjælden fik visit; de andre ænder holdt mere af at sv
ømme om i kanalerne, end at løbe op og sidde under et skræppeblad for at snadre med hende.$
```

Kan løses ved at kombinere BWT med ”Suffix-Arrays” (plus lidt mere)



# Opsummering

- Burrows-Wheeler transformation
  - kan beregnes effektivt (ikke vist)
  - der findes en invers transformation
  - den inverse kan beregnes effektivt

Michael Burrows

David J. Wheeler

- Mange (overraskende) anvendelser, f.eks.
  - komprimering af sekvenser og billeder
  - hukommelseseffektive søgestrukturer til mønstergenkendelse
  - bioinformatik, f.eks. ”sequence alignment”
    (AU bioinformatikkursus *Genome-Scale Algorithms*)

https://en.wikipedia.org/wiki/Burrows–Wheeler_transform

# Velkommen til *itd$ogala*

Link til slides