

## Skakter

En skakt er en datastruktur som sammenkæder sammenlignelige passager i tekstfiler. I følgende eksempel har vi tre dokumenter med tre betydningenheder.

Det var en god historie

Det var en mørk

Det var en historie

Det var en mørk og stormfuld nat.

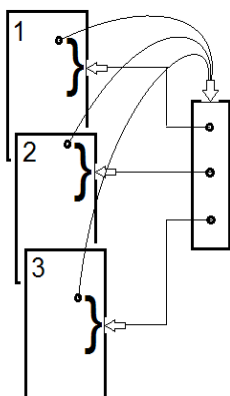
og stormfuld kat.

Greven lå på sin chaiselong. Det var en stormfuld nat.

Himmel og jord stod i ét

Det var en værre historie

“Jord og himmel stod i ét”,  
kunne man sige



Som man kan se indledes nr. 1 og 3 med et afsnit som i 2 er anbragt til slut. Den sammenlignelige tekst forekommer i nr. 3 midt i et afsnit, og tredje afsnit forekommer kun i 1 og 3.

De sammenlignelige dele af den enkelte tekst kaldes et *skaktmål*, som refererer til samme *skakt*. Selve skakten er en ekstern fil, hvorfra der refereres tilbage de enkelte mål, se figuren.

Kodningen af en skakt er i TEI foretaget efter reglerne i [TEI Guidelines 16.5](#), om "[alignment](#)", fx

*skakt.xml*:

```
<linkGrp type="alignment" n="ex">
  <link xml:id="skt01.1" type="movedText"
        target="ex1.xml#skt01.1 ex2.xml#skt01.1 ex3.xml#skt01.1" n="Om historien"/>
  <link xml:id="skt01"
        target="ex1.xml#skt01 ex2.xml#skt01 ex3.xml#skt01"/>
  <link xml:id="skt02"
        target="ex1.xml#skt02 ex2.xml#skt02 ex3.xml#skt02"/>
</linkGrp>
```

`<linkGrp>` bør have en `@type="alignment"`, som anført i Guidelines. Hver `<linkGrp>` giver anledning til en sammenligningsfil (se senere) som tager navn efter `@n`, fx *n.xml*.

Et `<link>` kan udover `@xml:id` have en `@type="movedText"` som angiver at dette tekststed ikke har samme position i alle versioner. De enkelte skaktmål angives som `@target` i en række adskilt med mellemrum, jf. Guidelines. Endelig kan tekststedet gives et navn, `@n`, som kan være nyttigt, specielt ved flyttede tekststeder.

I den enkelte version kodes en henvisning til et `<link>` i denne skaktstruktur med en reference fra det relevante element, fx

*ex1.xml*:

```
<p xml:id="skt01" corresp="skakt.xml#skt01">Det var en mørk og stormfuld nat.</p>
```

Der henvises til skaktstrukturen med `@corresp` og stedet gives mest hensigtsmæssigt samme id som skakten. Hvor flere afsnit (som her `<p>`) indgår i versionen kan man omkransede dem med et neutralt element:

*ex2.xml*:

```
<argument xml:id="skt01" corresp="skakt.xml#skt01">
  <p>Det var en mørk</p>
  <p>og stormfuld kat.</p>
  .
</argument>1
```

Dersom man har brug for at angive et område som bryder med xml-træet, benyttes `<milestone>`:

*ex3.xml*:

```
<milestone unit="p" xml:id="skt01.1" corresp="skakt.xml#skt01.1" spanTo="#skt01.1.end"/>
<p>Det var en historie</p>
<p>Greven lå på sin chaiselong.
  <milestone unit="p" xml:id="skt01.1.end"/>
  <milestone unit="p" xml:id="skt01" spanTo="#skt01.end" corresp="skakt.xml#skt01"/>
  Det var en stormfuld nat.
  <milestone unit="p" xml:id="skt01.end"/>
</p>2
```

## Udstrækning og granularitet

Skakter har to formål: At lede læseren til at læse de andre versioner i deres sammenhæng ('positiv-integralt variantapparat'<sup>3</sup>) og at kunne foretage en automatisk kollation.

Hvis vi et øjeblik fokuserer på det sidste og teknisk mest krævende, og hvis vi videre antager at vi ønsker at samlæse  $n$  versioner i  $n$  spalter fx som genstand for et kollationeringsprogram som *Juxta*<sup>4</sup> eller *CollateX*,<sup>5</sup> stilles vi over for to krav:

- Alle skakter må være repræsenteret i alle versioner og omvendt alle versioner i alle skakter, uanset om der er tekstmateriale til det og
- de henviste områder må dække hele versionens tekst.

Det første fordi vi ellers ikke kan vide hvor tom plads skal reserveres i versionen, det andet fordi der ellers vil mangle tekst i en version og dermed gøre spalteteksten urepræsentativ for versionen. I det ovenstående eksempel betyder det fx at skakten tilhørende det sidste afsnit også må refereres fra *ex2*, fx

```
<argument xml:id="skt01" corresp="skakt.xml#skt01">
  <p>Det var en mørk</p>
  <p>og stormfuld kat.</p>
  <milestone unit="p" xml:id="skt02" corresp="skakt.xml#skt02"/>
  .
</argument>
```

Forstil os fx at ⇕ angiver en skakt mellem de to udgaver:

Det var en mørk og stormfuld nat.	Det var en mørk og stormfuld kat.
⇕	⇕
Himmel og jord stod i ét	Jord og himmel stod i ét

Typisk vil vi gerne have kollationeret "nat" mod "kat", "Himmel" mod "Jord" og omvendt. Og det giver ingen mening hvis ikke selve skaktpunktet er invariant tekst. Hvis dette nu ikke tilfældig- eller lykkeligvis er tilfældet pga. den aktuelle tekst eller kollationeringsalgoritmes beskaffenhed, kan vi tvinge det frem ved at foretage en selvstændig kollation med skaktens som udgangspunkt.

1 Den semantiske betydning af `<argument>` er næppe helt i overensstemmelse med denne brug. De mulige kandidater til en rent syntaktisk 'neutral' mærkning som er desværre få. `<div>` tillader fx ikke efterfølgende `<p>`. `<ab>` må ikke indeholde `<p>`, osv.

2 `<milestone>` tager en obligat `@unit`. Her er valgt det element "p" som i andre versioner svarer til klippet. Andre muligheder er "chapter", "movedText", eller simpelthen "shaftTarget", attributværdien anvendes p.t. ikke.

3 »Tekstkritiske retningslinier for *Søren Kierkegaards Skrifter*, Trykte skrifter«, 4.1.1, [sks.dk/red/retningslinier\\_ts.pdf](https://sks.dk/red/retningslinier_ts.pdf)

4 Textual collation on the web, <http://www.juxtacommons.org>, specielt 'paparallel segmentation'

5 *CollateX* – Software for Collating Textual Sources, *CollateX*, fra *The Interedition Development Group*, [collatex.net](http://collatex.net).

Ideen er mao. at bruge skakt-kodningen som udgangspunkt for kollationeringen<sup>6</sup>.

En særlig udfordring udgør flyttet tekst, her fx afsnittet "Om historien". Hvor i de to versioner skal vi anbringe afsnittet? Umiddelbart vil skaktmålene blive kollationeret og anbragt efter den rækkefølge de har i skaktens `<linkGrp>`, og hvis skakten er mærket `@type="movedText"` mærket som sådan. Bemærk dog at hvis imidlertid skaktmålet tillige indgår i et andet skaktmål, vil det også optræde to gange. I det flg. eksempel er der valgt den inkonsekvente løsning at lade den flyttede tekst være indlejret i den version hvor placeringen afviger fra de andre versioner, og hvor den derfor optræder to gange. Den endelige kodning af skaktmål *skt01* i *ex2* bliver dermed:

```
<argument xml:id="skt01" corresp="skakt.xml#skt01">
  <p>Det var en mørk</p>
  <p>og stormfuld kat.</p>
  <milestone unit="p" xml:id="skt02" corresp="skakt.xml#skt02"/>
  <p xml:id="skt01.1" corresp="skakt.xml#skt01.1">Det var en værre historie</p>
</argument>
```

Vi kunne have gjort det samme med de to andre versioner, der ville så bare være tekstgentagelse i alle tre spalter. Det kunne imidlertid løses ved at henvise flyttet tekst til selvstændig visning i egen fil, nemlig ved at flytte skaktens `<link>` til sin egen `<linkGrp>`, fx *skakt.xml*:

```
<linkGrp type="alignment" n="ex">
  <link xml:id="skt01" target="ex1.xml#skt01 ex2.xml#skt01 ex3.xml#skt01"/>
  <link xml:id="skt02" target="ex1.xml#skt02 ex2.xml#skt02 ex3.xml#skt02"/>
</linkGrp>
<linkGrp type="alignment" n="ex.skt01.1">
  <link xml:id="skt01.1" type="movedText" target="ex1.xml#skt01.1 ..." n="Om historien"/>
</linkGrp>
```

<sup>6</sup> AUH. Kollationering af filer, KK 2017.12.15.

Et særligt problem udgør anførselstegn i forskellige versioner. Her vil forskellige udgaver af et værk ofte bruge forskellige typer, som man imidlertid ikke ønsker med i en sammenligning. Til den ende kan man med fordel anvende TEIkodning af anførselstegn, fx *ex3.xml*:

```
<p xml:id="skt02" corresp="skakt.xml#skt02"><q>Jord og himmel stod i ét</q>, kunne man  
sige</p>
```

For imidlertid overhovedet at kunne se anførslerne i tekstsammenligning (og evt. variation af dem) kan man indsætte en standardtype inden selve sammenligningen. Det er hvad det er gjort her, og derfor har de en anden facon end i originalen.