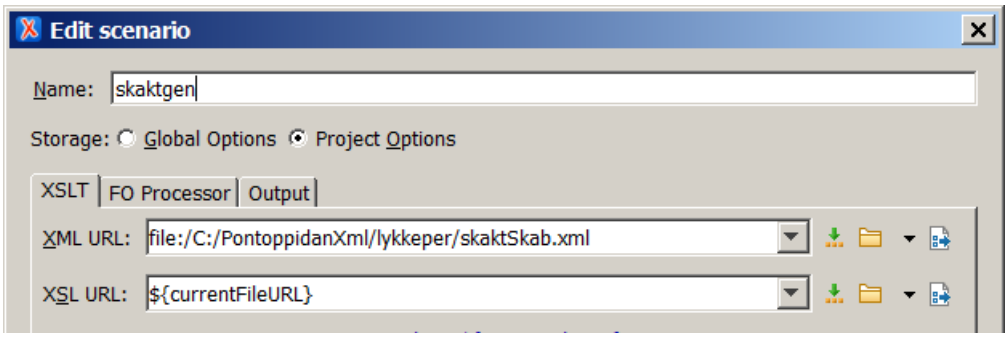
Elektronisk udgivelse af Henrik Pontoppidans tre store romaner
Automatisk generering af skakter
skaktgen.xsl vers. 02

Karsten Kynde

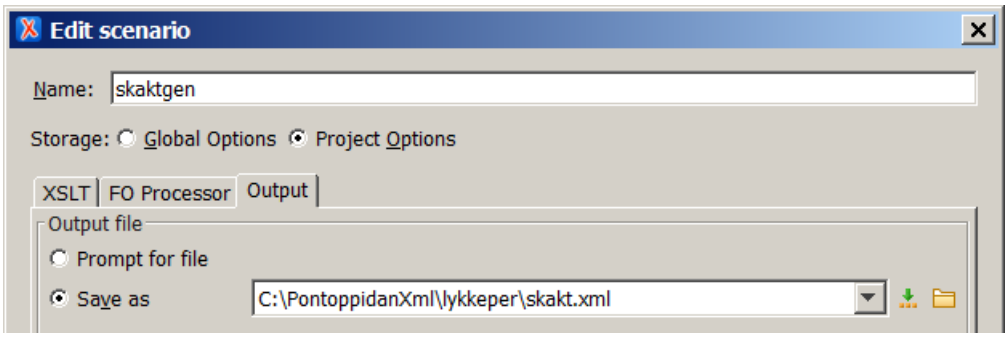
Kort vejledning

Skaktgeneratoren er en XSL transformation, findes på **etxt.dk/pontoppidan/xml/skaktgen.xsl**.

Transformationen kan foretages med Oxygen, *configure transformation scenario* (ctrl-skift-T):



og *Output*



eller fra kommandopromt (her fra en *Windows pc*)

```
C:\PontoppidanXml\lykkeper>altovaxml /xslt2 ..\skaktgen.xsl /in skaktSkab.xml /out skakt.xml
```

(idet de indgående filer formodes fordelt i `C:\PontoppidanXml\` på samme måde som på etxt.dk/pontoppidan/xml/)

Input for transformationen er en skakt*skabelon*, et TEI formatteret dokument bestående af en `<teiHeader>` og en `<text>`. *TeiHeader*'en kopieres uforandret over i den resulterende skakt, bortset fra at der til revisionsbeskrivelsen tilføjes en `<change>` hvoraf det fremgår at skakten er autogenereret, hvornår og af hvilken version.

`<text>` skal efter TEI indeholde en `<body>` som igen skal indeholde et `<p>`. Heri angives et antal linkgrupper; hver gruppe udpeger en række skaktmål som af kollationsprogrammet vil blive anbragt i separate filer. Gruppen kan have to attributter:

`<linkGrp>`

`@n="navn"`	obligat; navn som vil gå igen i filnavnet på en kollation af teksten
`@domains="#id1 #id2"`	valgfri; angiver to link-id; skakt-id'erne formodes valgt alfabetisk fortløbende i teksten, og links med id *inklusive og efter* `id1` *til men eksklusive* `id2` vil indgå i gruppen. id'erne kan anføres med eller uden foranstillet `#`. Hvis `@domains` ikke angives, antages `id1` = `@xml:id` for det første `<link>` i gruppen og `id2` = `@xml:id` for det første `<link>` i den følgende gruppe. Hvis `id2` hermed bliver mindre end `id1`, eller hvis der ikke er flere grupper, er der ingen øvre grænse.

I gruppen angives et antal *links*, dette er hvad der skal blive til de egentlige skakter. Første link har flg. attributter:

`<link>`

`@xml:id="id"`	obligat midmindre `@domains` ikke er angivet i jf. `<linkGrp>` ovenfor; id for den første skakt i denne gruppe.
`@target="fil1 fil2 ..."`	obligat for mindst ét `<link>` i skabelonen; filnavne adskilt med mellemrum på de filer hvori der vil blive søgt skaktmål. Der vil blive søgt i enhver fil som er anført i noget link i skabelonen. Et evt. # med tilhørende id efter filnavnet ignoreres. I den resulterende skaktfil vil filerne inkl. evt. foranstillet sti blive anbragt alfabetisk.

En særlig variant benyttes ved **flyttet tekst:**

`<linkGrp>`

`@n="navn"`	valgfri; som ovenfor; hvis `@n` ikke angives vælges et navn udfra linkets `@xml:id`

`<link type="movedText">`

`@xml:id="id"`	obligat;
`@type="movedText"`	obligat;
`@n="navn"`	obligat; betegnelse for indholdet af teksten tilhørende denne skakt, benyttes ved flyttet tekst

Selvom der ikke angives et link og en linkgruppe for en flyttet tekst, vil alle sådanne alligevel blive opdaget og noteret i den resulterende skakt, for så vidt de er omfattet af de andet steds anførte intervaller. De vil så blot ikke have nogen betegnelse, og kollationsfilen vil få et automatisk genereret navn. I den resulterende skakt vil et link der repræsenterer flyttet tekst altid have egen linkgruppe.

Eventuelle xml-kommentarer vil blive kopieret uændret til resultatfilen.

Output vil udgøre en valid skakt for tekstsammenligning. Ovenstående specifikation er således

indrettet at output tillige vil kunne fungere som input til samme transformation. Hvis filerne i øvrigt er urørt, vil en transformation hvis input er en forudgående transformations output, producere en uændret fil bortset fra at der vil være tilføjet endnu et `<change>` element i headeren.

Lidt længere vejledning

Skaktmarkeringen er organiseret som dobbeltrettede referencer, dvs. der henvises fra de enkelte skaktmål i tekstfilerne til skakten som igen viser tilbage til de sammenlignelige skaktmål i alle relevante filer. Dermed er der skabt redundans og dermed en sandsynlig fejlkilde. Det er derfor en god ide hvis skakterne kunne genereres automatisk ud fra de kodede filer.

Der mangler ganske få data for at det kan lade sig gøre. Bl.a. må man vide hvilke de relevante filer er. Af praktiske grunde skal kollationsfilerne ikke være for store. Det betyder at vi må angive hvordan vi ønsker dem opdelt og hvad vi vil kalde dem. Flyttet tekst skal behandles specielt og vi vil gerne give flytningerne et kaldenavn og placere dem i hver deres fil. Endelig skal skatfilen som alle andre TEI-filer have en *header* med oplysninger om tid, sted og titler og al det der.

Til den ende begynder vi den automatiske generering med en skabelon over skakterne som netop indeholder de ønskede oplysninger. Skabelonen er for genkendelighedens skyld designet i samme stil som den resulterende skaktfil. Faktisk er det gjort så smart at man kan bruge en fungerende skaktfil som skabelon for en ny version.

Eksempel
Følgende er en annoteret, minimal skaktskabelon for *Lykke-Per*:

```
<TEI xmlns="http://www.tei-c.org/ns/1.0">
  <teiHeader>
    <fileDesc>
      <titleStmt>
        <title>Skakter til Lykke-Per</title>
      </titleStmt>
      <publicationStmt>
        <authority>
          <name xml:id="KK">Karsten Kynde</name>
          <date>2020-03-28</date>
        </authority>
        <availability>
           <p>http://etxt.dk/pontoppidan/xml/lykkeper/skakt.xml</p>
        </availability>
      </publicationStmt>
      <sourceDesc>
        <p/>
      </sourceDesc>
    </fileDesc>
    <encodingDesc>
      <p>Shafts interconnecting variants in editions are coded as described
for &lt;linkGrp type=/&gt; in the TEI Guidelines chapter 16.4</p>
    </encodingDesc>
  </teiHeader>
```

Headeren kopieres uændret til den færdige skaktfil.

```
  <text>
    <body>
      <p>
        <linkGrp n="lp1">
          <link xml:id="skt01" target="1udg/lp1.xml 1udg/lp2.xml
1udg/lp3.xml 1udg/lp4.xml 1udg/lp5.xml 1udg/lp6.xml 1udg/lp7.xml
1udg/lp8.xml 2udg/lp.xml 4udg/lp.xml"/>
```

I første link anføres samtlige relevante filer, dvs. de otte bind af førsteudgaven og to samleudgaver.

```
        </linkGrp>
```

Denne gruppe omfatter intervallet skt01 til skt05 da der ikke er angivet nogen @domains, dvs. alle skakter med id der begyner med skt01, skt02, skt03, og skt04. Ved kollationsfilen for denne gruppe kommer jf. @n til at hedde ***lp1***.*col.xml*

```
        <linkGrp n="lp2">
          <link xml:id="skt05"/>
        </linkGrp>
        <linkGrp n="lp3">
          <link xml:id="skt08"/>
        </linkGrp>
        <linkGrp n="lp4">
          <link xml:id="skt11"/>
        </linkGrp>
        <linkGrp n="lp5">
          <link xml:id="skt14"/>
        </linkGrp>
        <linkGrp n="lp6">
          <link xml:id="skt17"/>
        </linkGrp>
        <linkGrp n="lp7">
          <link xml:id="skt19"/>
        </linkGrp>
        <linkGrp n="lp8">
          <link xml:id="skt23"/>
        </linkGrp>
```

Denne gruppe omfatter alle skakter med id højere end skt23 . Der er ikke noget loft over intervallet, fordi det er den sidste i rækken (ud over den flyttede tekst som ikke tæller med):

```
        <linkGrp n="flyt.02a">
          <link xml:id="skt02a" n="Projektet beskrives"/>
        </linkGrp>
        <linkGrp n="flyt.12a">
          <link xml:id="skt12a" n="Pers julelov"/>
        </linkGrp>
```

skt02a og skt12a er tekster som er flyttet. De har på denne måde fået et navn til at beskrive den flyttede tekst.

```
      </p>
    </body>
  </text>
</TEI>
```

Den resulterende skakt ser sådan ud:

```
<TEI:TEI xmlns="http://www.tei-c.org/ns/1.0" xmlns:TEI="http://www.tei-c.org/ns/1.0">
   <teiHeader>
      <fileDesc>
         <titleStmt>
            <title>Skakter til Lykke-Per</title>
         </titleStmt>
...
      <revisionDesc>
         <change who="auto" when="2020-04-06">Autogenereret med skaktgen.xsl,
                                              vers. 02 KK 2020-04-06</change>
      </revisionDesc>
   </teiHeader>
   <text>
      <body>
         <p>
            <linkGrp type="alignment" domains="#skt01 #skt05" n="lp1">
```

```
            <link xml:id="skt01"
                  target="1udg/lp1.xml#skt01 2udg/lp.xml#skt01 4udg/lp.xml#skt01"/>
            <link xml:id="skt02"
                  target="1udg/lp1.xml#skt02 2udg/lp.xml#skt02 4udg/lp.xml#skt02"/>
...
            <link xml:id="skt04.3"
                  target="1udg/lp1.xml#skt04.3 2udg/lp.xml#skt04.3 4udg/lp.xml#skt04.3"/>
         </linkGrp>
         <linkGrp type="alignment" n="flyt.02a">
            <link xml:id="skt02a"
                  type="movedText"
                  n="Projektet beskrives"
                  target="1udg/lp1.xml#skt02a 2udg/lp.xml#skt02a 4udg/lp.xml#skt02a"/>
         </linkGrp>
         <linkGrp type="alignment" n="flyt.02b">
            <link xml:id="skt02b"
                  type="movedText"
                  target="1udg/lp2.xml#skt02b 2udg/lp.xml#skt02b 4udg/lp.xml#skt02b"/>
         </linkGrp>
         <linkGrp type="alignment" domains="#skt05 #skt08" n="lp2">
...
         <linkGrp type="alignment" domains="#skt23 #~ubegrænset" n="lp8">
...
         </linkGrp>
      </p>
   </body>
</text>
</TEI:TEI>
```

Det viser sig at der var endnu et indlejret skaktmål `"skt02b"`, som ikke var opført i skabelonen. Bemærk at der alligevel er belevet genereret et link for dette sted, blot mangler der en @`n` til at betegne teksten. Den kunne man passende anføre nu, efter første generation af den første skaktfil. Den kan derefter bruges som skabelon for næste generation.

Bemærk også at @`domains` er blevet påført eksplicit. Undtagen for flyttet tekst: Sådanne grupper har altid netop ét link.

Et eksempel på brugen af @`domains` i skabelonen findes i DET FORJÆTTEDE LAND, hvor vi valgt at benytte deltitlerne i skakt-id'erne. Da vi gerne vil følge handlingskronologien i skaktfilen og da forf. ikke har være så hensynsfuld at ordne sine titler alfabetisk (tværtimod, faktisk), laver vi skabelon som følger:

```
<linkGrp type="alignment" n="dfland1">
  <link xml:id="m.i.skt01"
        target="0til/til2.xml 0til/til2.xml
                1udg/m.xml 1udg/dfl.xml 1udg/dd.xml
                2udg/m.xml 2udg/dfl.xml
                3udg/dfland.xml
                4udg/dfland.xml
                5udg/dfland.xml"/>
</linkGrp>

<linkGrp type="alignment" n="dfland2" domains="#dfl.i.skt01 #dfl.~">
  <link xml:id="dfl.i.skt01"/>
</linkGrp>

<linkGrp type="alignment" n="dfland3" domains="#dd.i.skt01 #dd.~">
  <link xml:id="dd.i.skt01"/>
</linkGrp>
```

Husk at intervallet er eksklusive sidst anførte id alfabetisk. Tegn sorteres før cifre, som sorteres før store bogstaver, som sorteres før små. Sidst kommer '~'. Romertal < ix sorteres også pænt.

Udover at opfylde syntaksen for xml og være valide TEI-filer, er er et par andre krav til skakter, som det vil være godt at tjekke: Der anvendes samme `xml:id` til linket, og til de enkelte steder i

tekstfilerne. Det kan tjekkes under indsamlingen af skaktmål, og skulle der være en forskel rapporteres det dels med en s.k. *message* under transformationen. Skulle opratøren overse denne, vil den være gentaget i skaktfilen, fx

```
<link xml:id="skt16.x" target="4udg/lp.xml#skt16.1">xml:id != @target #</link>
```

Da der ydermere ikke må være indhold i et link, forårsager fejlmeddelelsen en valideringsfejl og et rødt mærke i *Oxygen*. Fejlen skal her findes i fjerdeudgaven `4udg/lp.xml`.

Alle links i en gruppe bør have lige mange felter i `@target`. Her kan man pådrage sig flg. fejludskrift ved afslutningen af linkgruppen:

```
FEJL i dfl.i.skt02:
"1udg/dfl.xml#dfl.i.skt02 3udg/dfland.xml#dfl.i.skt02
 4udg/dfland.xml#dfl.i.skt02 5udg/dfland.xml#dfl.i.skt02":
4 henvisninger != 5</linkGrp>
```

der mangler en henvisning til `2udg/dfl.xml`, så der er kun 4 mod de andres 5 henvisninger.

Al tekst bør være inkluderet i et eller flere skaktmål. Hvis de ikke er, kan man opleve en udskrift i slutningen af skaktfilen:

```
3udg/dfland.xml: Flg. tekst er ikke dækket af en skakt:
                 ...</body>
```