

ØVELSER TIL `avenue.quark`

 **Indholdsfortegnelse**

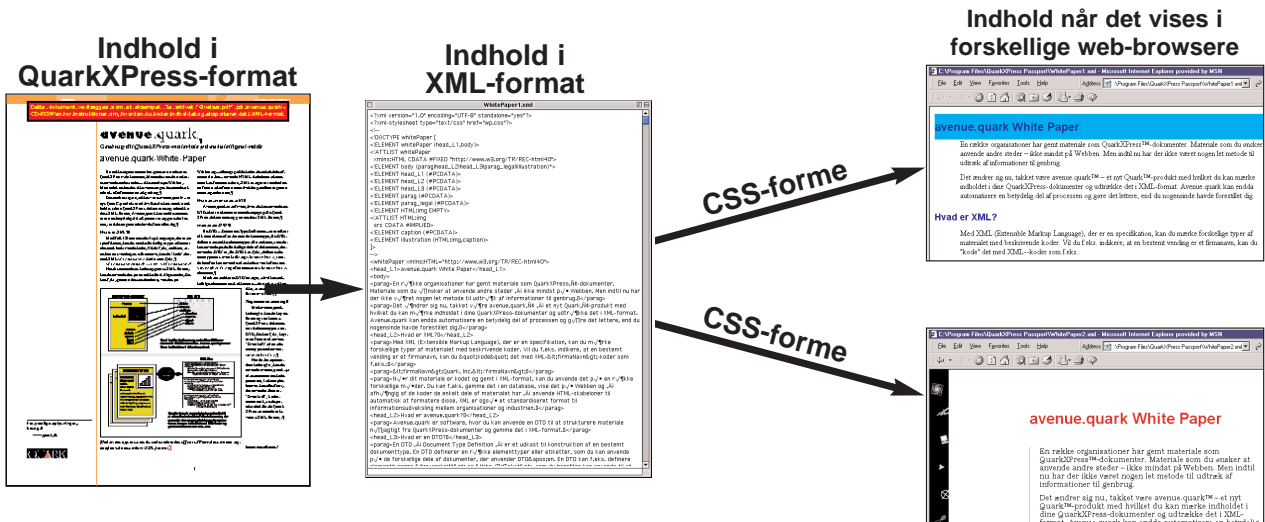
Indledning	3
Klargøring til brug af avenue.quark	5
Brug af regelbaseret kodning	6
Visning af udtrukket XML-indhold	13
Tilføjelse af en regel til et kodningsregelsæt	17
Kombinere tekstfelter til en sekvens	22
Kodning med sekvenser og en ny regel	25
Kodning af et billede og en billedtekst	27

Indledning

I disse øvelser skal du:

- 1 | Åbne et QuarkXPress Passport™-eksempeldokument.
- 2 | Bruge avenue.quark™ sammen med en vedhæftet DTD (Document Type Definition) til kodning af det meste af indholdet i eksempeldokumentet.
- 3 | Gemme det kodede indhold i XML-format (Extensible Markup Language).
- 4 | Vise indholdet i en web-browser med Cascading Style Sheets (kun Windows).
- 5 | Finpudse kodningsprocessen ved først at opdatere kodningsregelsættet til kodning af flere indholdstyper, og herefter kombinere flere tekstfelter til en sekvens, der kan kodes som en enhed.
- 6 | Kode et billede og en billedtekst.

Fordelen ved avenue.quark er, at du kan kode indhold der aktuelt er gemt i QuarkXPress Passport-format, eksportere indholdet i industristandard XML-format og herefter genbruge det på flere forskellige måder. I dette øvelsesdokument vil vi f.eks. tage de originale dokumenteksempler (til venstre i figuren herunder), eksportere indholdet i XML-format med avenue.quark og derefter vise indholdet i en web-browser med to helt forskellige sæt CSS-forme (Cascading Style Sheet-forme) (til højre i figuren herunder).



QuarkXPress Passport-indhold gemt som XML og vist i en web-browser med CSS

!!! Se kapitel 1 "Start på avenue.quark" i "En vejledning til avenue.quark" ("Vejledning til avenue.quark.pdf") for en dyberegående introduktion til XML, DTD'er og avenue.quark. Se kapitel 5 "Kodningsregelsæt" i det samme dokument for yderligere informationer om kodningsregelsæt.

Klargøring til brug af avenue.quark

Inden du starter på dette øvelsesdokument skal du udføre følgende trin:

- 1** | Kontroller at du har QuarkXPress Passport 4.04 eller senere.
- 2** | Kontroller at QuarkXTensions™-modulerne avenue.quark og Item Sequence findes i mappen "XTension" i QuarkXPress Passport-programmappen. Installation Instructions.pdf" findes installationsinstruktioner.
- 3** | Kontroller at filen "WhitePaper - Dansk.xmt" findes i mappen "Templates" i QuarkXPress Passport-programmappen.
- 4** | Kopiér mappen "avenue.quark Tutorial" til computerens Skrivebord, så du har let adgang til den.

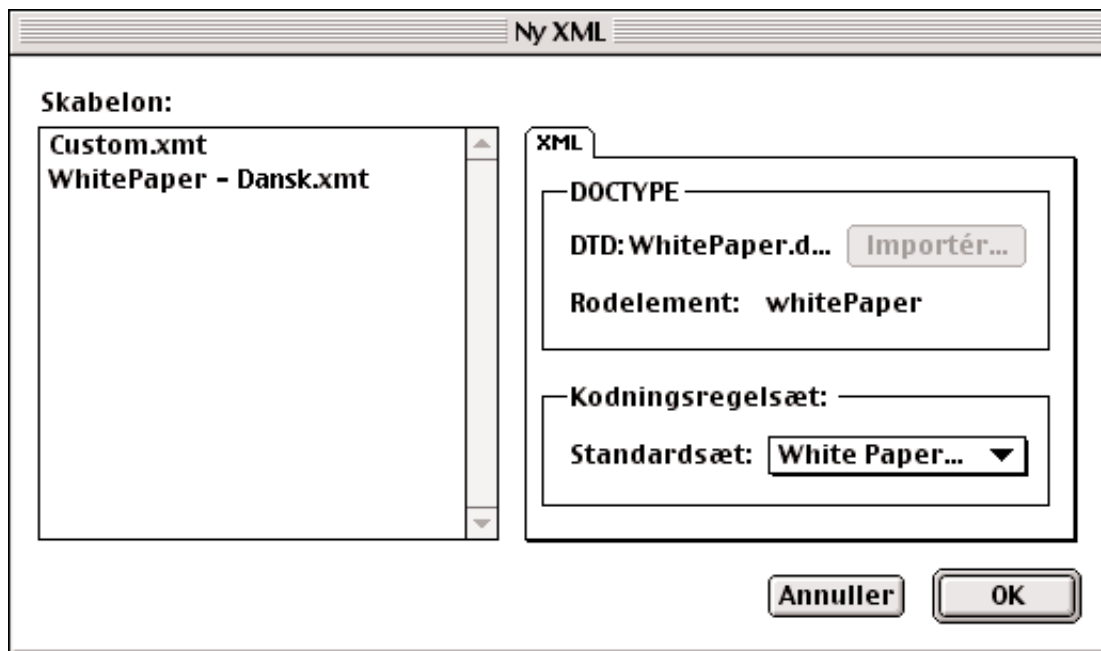
Det kan være en god idé at udskrive denne vejledning, så du let kan bruge den mens du gennemgår de enkelte trin.

Brug af regelbaseret kodning

I dette afsnit vises, hvordan man kan bruge en funktion, der kaldes for *regelbaseret kodning*, til automatisering af en stor del af XML-kodningsprocessen. Regelbaseret kodning er baseret på et *kodningsregelsæt*, der igen er en række retningslinier til automatisk kodning af QuarkXPress Passport-indhold, baseret på indholdets QuarkXPress Passport-forme og andre formateringsattributter.

Sådan bruger du et tidligere oprettet kodningsregelsæt til kodning af indholdet i et tekstfelt i et eksempeldokument:

- 1 Start QuarkXPress Passport.
- 2 Vælg **Arkiv** (Mac OS) eller **Fil** (Windows) → **Åbn**, find dokumentet med navnet "EksempelDok.qxt" i mappen "avenue.quark Tutorial" på Skrivebordet.
- 3 Vælg **Arkiv** (Mac OS) eller **Fil** (Windows) → **Ny** → **XML** for at oprette et nyt XML-dokument. Dialogen **Ny XML** vises.



Dialogen **Ny XML**



Hvis objektet **XML** ikke vises i undermenuen **Arkiv** (Mac OS) eller **Fil** (Windows) → **Ny** er avenue.quark sikkert ikke installeret korrekt. Se “Installation Instructions.pdf” for instruktioner om, hvordan du installerer softwaren.

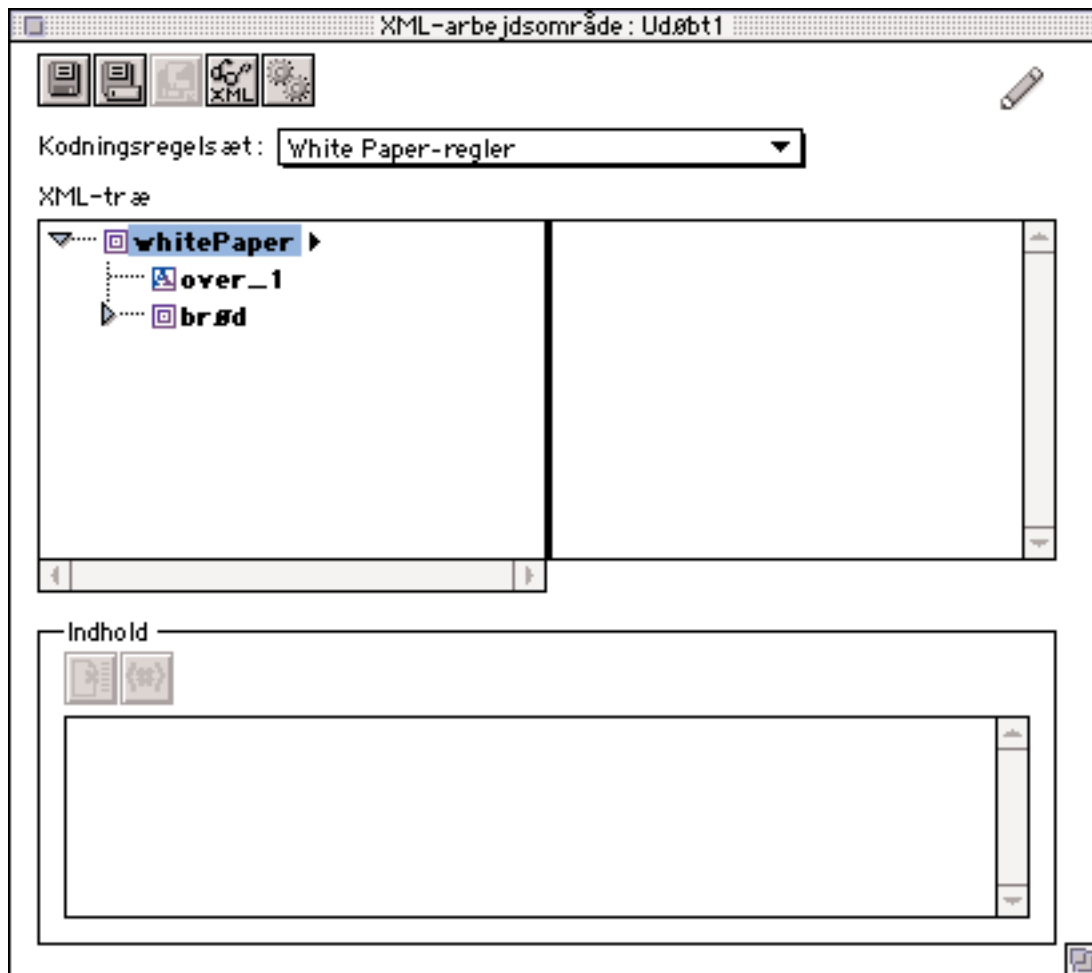
4

Vælg **WhitePaper - Dansk.xmt** på listen **Skabelon** for at angive, at du vil basere det nye XML-dokument på skabelonen “WhitePaper - Dansk.xmt”. I denne skabelon er angivet, at du vil bruge en DTD med navnet “WhitePaper.dtd” med `<whitePaper>` som kodelement, og at du vil bruge det foruddefinerede kodningsregelsæt “White Paper-regler”.



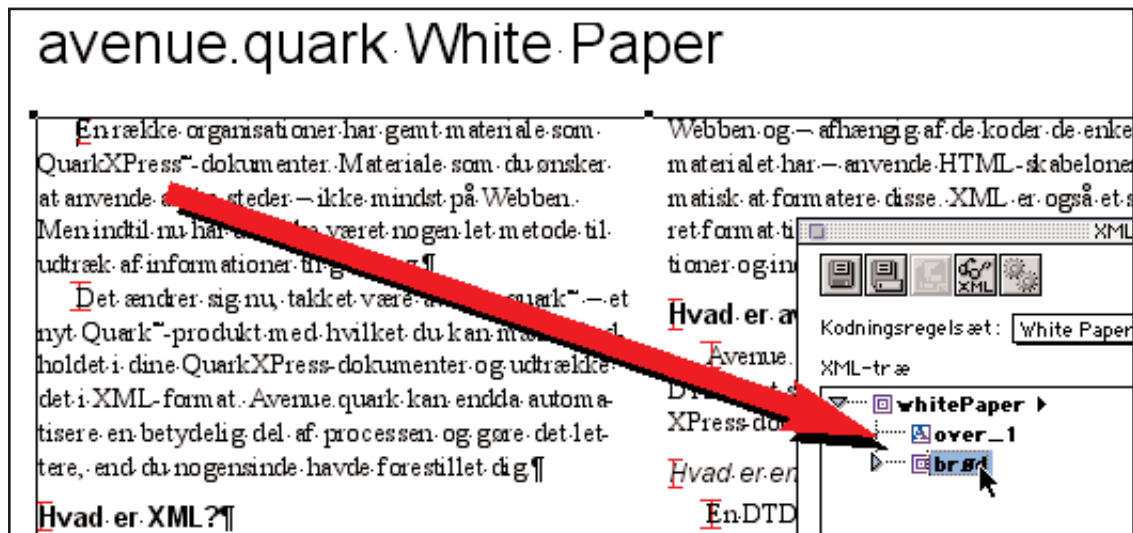
Hvad er en DTD? En DTD, eller Document Type Definition, definerer en række elementtyper (kodedefinitioner) og strukturregler for en bestemt dokumenttype. En DTD for en nyhedsartikel kan f.eks. angive, at hver artikel skal have nøjagtig én sektion med tekst, der er kodet som et `<overskrift>`-element, at hver artikel kan have et `<forfatter>`-element, at hver illustration umiddelbart skal efterfølges af et `<billedtekst>`-element osv. Se kapitel 1 “Start på avenue.quark” i “Vejledning til avenue.quark.pdf” for flere informationer om DTD’er.

- 5 Klik på OK. Paletten **XML-arbejdsområde** vises og alle obligatoriske elementer i WhitePaper-DTD'en (i dette tilfælde `<whitePaper>`, `<over_1>` og mindst ét `<brød>`) føjes automatisk til listen **XML-træ**.





Paletten **XML-arbejdsområde**

- 6 Markér det tekstfelt, hvori teksten begynder med ordene “En række organisationer har...”. Tryk herefter på ⌘ (Mac OS) eller Ctrl (Windows), og træk tekstfeltet over på elementet `<brød>` på listen XML-træ. (Elementnavne vises med fed skrift på listen XML-træ, f.eks. som **brød**). Avenue.quark koder automatisk indholdet i tekstfeltet og opretter nye elementer, hvis det er nødvendigt.



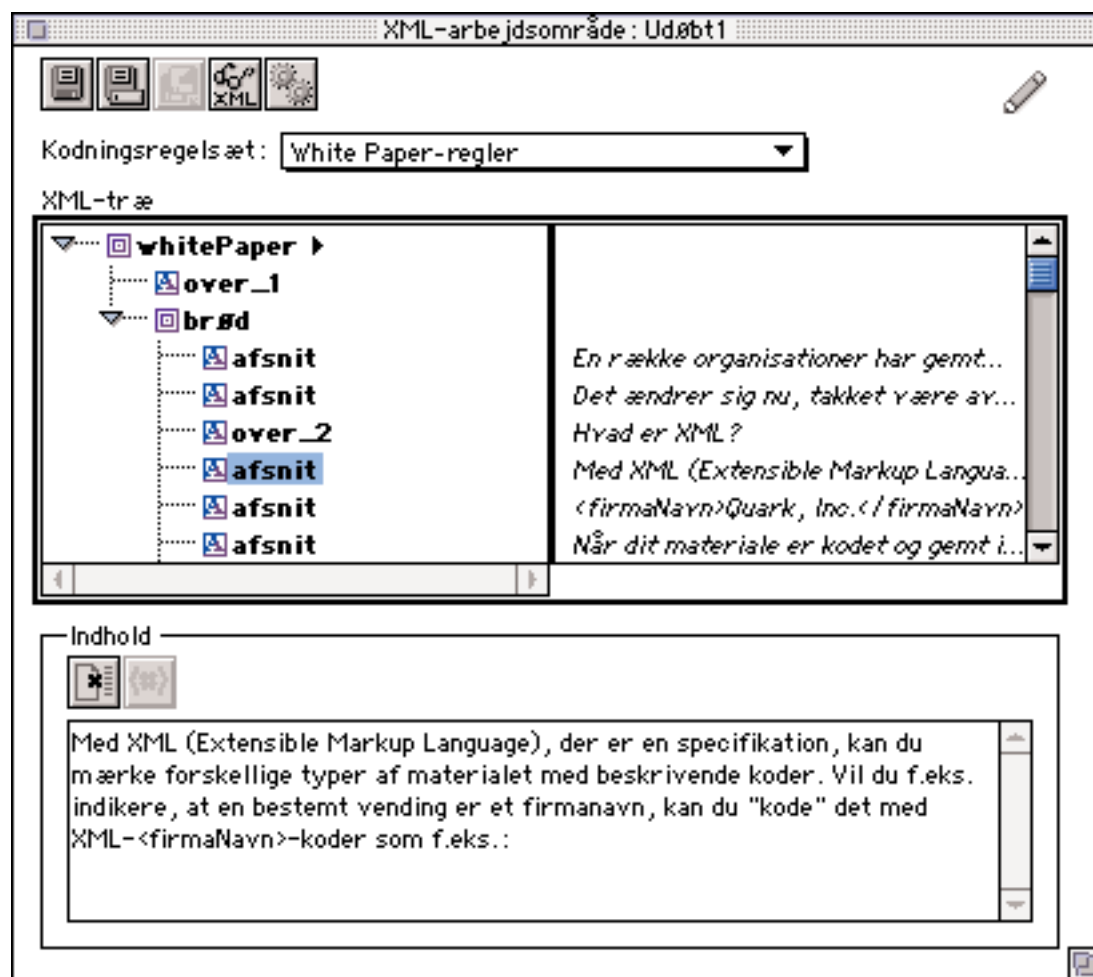
Regelbaseret kodning

 **Hvad skete der egentlig?** Avenue.quark anvendte kodningsregelsættet “White Paper-regler” til kodning af indholdet i det tekstfelt, du trak over på elementet `<brød>`. Du kan læse mere om kodningsregelsættet i afsnittet “Føj en regel til et kodningsregelsæt” i dette øvelsesdokument.

 **Stop, det fungerede ikke!** Hvis dialogen **Vælg regel/placering** vises når du trækker tekstfeltet over på elementet `<brød>`, skal du klikke på **Stop**. Vælg herefter **Redigér** → **Valg** → **avenue.quark**. Sørg for at feltet **Indsæt altid gentagende elementer i slutningen af den aktuelle gren** er afkrydset. Klik på **OK**, luk paletten **XML-arbejdsområde** og gå tilbage til trin 3.

Hvis der skete noget andet uventet da du trak tekstfeltet over på elementet `<brød>` på listen XML-træ, skal du ikke spekulere nærmere over dette. Luk blot paletten XML-arbejdsområde, og gå tilbage til trin 3. Det kan i dette tilfælde være en idé at undersøge, om filen "WhitePaper - Dansk.xmt" fra mappen "avenue.quark Tutorial" findes i mappen "Templates" i QuarkXPress Passport-programmappen.

7 Rul gennem listen XML-træ (på paletten XML-arbejdsområde) og se, hvordan indholdet i white paper nu er blevet kodet med de korrekte elementtyper. Underoverskrifter er kodet som `<over_2>`-elementer, og afsnit med brødtekst er kodet som `<afsnit>`-elementer. Du kan se hele indholdet i et element ved at klikke på elementets navn og se i feltet **Indhold**.



Paletten XML-arbejdsområde med feltet Indhold

!!! Der er fire ting, der *ikke blev* kodet: Overskriften (“avenue.quark White Paper”), billedet, billedteksten og den juridiske tekst, der udgør de to sidste afsnit på side to. Senere i dette øvelsesdokument kan du se, hvordan du lige så nemt kan kode disse tekster.

8 Klik på knappen **Vis XML**, øverst på paletten **XML-arbejdsområde**. Dette åbner dialogen **Vis XML**. Heri kan du se hvordan den eksporterede XML-fil vil se ud.



Knappen **Vis XML**

```

<?xml version="1.0"?>
<!DOCTYPE whitePaper SYSTEM "whitePaper.dtd" [
]>
<whitePaper xmlns:HTML="http://www.w3.org/TR/REC-html40">
<over_1></over_1>
<brød>
<afsnit>En række organisationer har gemt materiale som QuarkXPress™-dokumenter. Materiale som du ønsker at
anvende andre steder - ikke mindst på Webben. Men indtil nu har der ikke været nogen let metode til udtræk af
informationer til genbrug.
</afsnit>
<afsnit>Det ændrer sig nu, takket være avenue.quark™ - et nyt Quark™-produkt med hvilket du kan mærke
indholdet i dine QuarkXPress-dokumenter og udtrække det i XML-format. Avenue.quark kan endda automatisere en
betydelig del af processen og gøre det lettere, end du nogensinde havde forestillet dig.
</afsnit>
<over_2>Hvad er XML?
</over_2>
<afsnit>Med XML (Extensible Markup Language), der er en specifikation, kan du mærke forskellige typer af
materialet med beskrivende koder. Vil du f.eks. indikere, at en bestemt vending er et firmanavn, kan du
&quot;kode&quot; det med XML-&lt;f;irmaNavn&gt;-koder som f.eks.:
</afsnit>
<afsnit>&lt;f;irmaNavn&gt;Quark, Inc.&lt;/f;irmaNavn&gt;
</afsnit>
<afsnit>Når dit materiale er kodet og gemt i XML-format, kan du anvende det på en række forskellige måder. Du kan
f.eks. gemme det i en database, vise det på Webben og - afhængig af de koder de enkelt dele af materialet har -
anvende HTML-skabeloner til automatisk at formatere disse. XML er også et standardiseret format til
informationsudveksling mellem organisationer og industrien.

```

Dialogen **Vis XML**



Hvad er det, der vises? Dialogen **Vis XML** viser hvordan indholdet ser ud i XML-format. Hver informationselement i en XML-fil er omgivet af firkantede parenteser med koder der beskriver, hvad det pågældende indhold er. Øverst i figuren herover kan du f.eks. se, at det første afsnit i dokumentet er anbragt mellem en `<afsnit>`-åbningskode og en `</afsnit>`-lukkekode. Dette gør det nemt for en lang række programmer at genkende brødteksten og behandle den korrekt – f.eks. at vise den med en 12-punkt, Serif-skrift.

9 Luk dialogen **Vis XML** ved at klikke på **OK**.

10 Klik på knappen **Arkivér** (Mac OS) eller **Gem** (Windows) øverst på paletten **XML-arbejdsområde** og gem dokumentet under navnet “MitDok.xml” i mappen “avenue.quark Tutorial” (bibehold **UTF8** i lokalmenuen **Tegnkodning**, og afkryds **Gem XML som standalone**). I næste afsnit “Visning af udtrukket XML-indhold” skal du vise denne fil i en web-browser.





Knappen **Arkivér** (Mac OS) eller **Gem** (Windows)

11 Luk XML-dokumentet ved at klikke på luk-knappen på paletten **XML-arbejdsområde**.

Visning af udtrukket XML-indhold

Når du har udtrukket indholdet fra et QuarkXPress Passport-dokument og gemt det i en XML-fil, kan du anvende indholdet til en række forskellige ting. Du kan f.eks. anvende Cascading Style Sheets (CSS) til at vise det i en web-browser som et formateret dokument. I dette afsnit vil vi først se på et par XML-eksempelfiler med CSS-formatering og herefter vise, hvordan du kan føje CSS-formatering til det XML-dokument, du gemte i det forrige afsnit.

 **Hvad er CSS?** CSS er en metode til at knytte en form til kodet indhold. CSS er udviklet til brug sammen med HTML, men kan også anvendes sammen med XML. Det er praktisk, da du så kan ændre formateringen i et stort antal dokumenter ved at opdatere en enkelt CSS-fil.

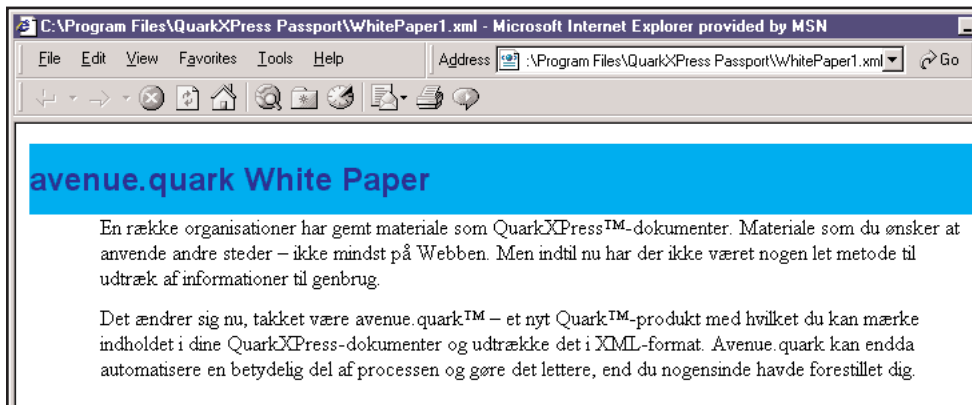
 I skrivende stund fungerer dette afsnit kun med Microsoft Internet Explorer 5.0, der kun er udviklet til Windows. Men når du læser dette findes der muligvis andre web-browsere (f.eks. Internet Explorer 5.0 til Mac OS), der kan vise XML med CSS-forme. Hvis du ikke bruger en browser der kan vise XML med CSS-forme, kan du gå videre til næste afsnit "Tilføjelse af en regel til et kodningsregelsæt".

Visning af en XML-fil med CSS-forme

For at gøre det nemt at se hvordan en XML-fil ser ud, når det vises med CSS, er der leveret to XML-eksempeldokumenter med dette øvelsesdokument. Disse dokumenter er baseret på det eksemplardokument, du arbejdede med i det forrige afsnit, og de er identiske med én undtagelse: De anvender forskellige CSS-filer til visningen. Sådan kan du vise og sammenligne filerne:

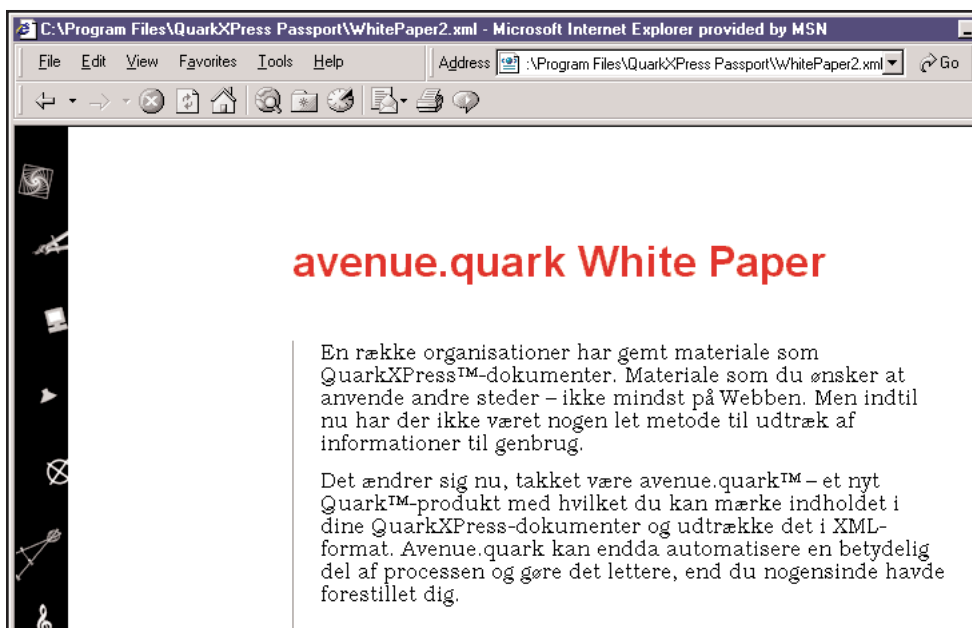
1 Start en web-browser der kan vise XML med CSS-forme, f.eks. Microsoft Internet Explorer 5.x

- 2** Åbn eksempeldokumentet "WhitePaper1.xml". Browseren viser XML-dokumentet og bruger CSS-formene i filen "wp.css". (Det eksporterede XML-dokument bør se ud på samme måde, når du viser det med denne CSS-fil i det næste underafsnit).



XML-versionen af white paper-dokumentet vises med CSS-formene i "wp.css"

- 3** Åbn eksempeldokumentet "WhitePaper2.xml" i web-browseren. Browseren viser det samme XML-indhold, men denne gang anvendes CSS-formene i filen "wp2.css". (Det eksporterede XML-dokument bør se ud på samme måde, når du viser det med denne CSS-fil i det næste underafsnit).



Det samme XML-dokument, vist med CSS-formene i "wp2.css"

Visning af XML-filen med CSS-forme

Sådan kan du vise den XML-fil, du oprettede i første afsnit:

1 Brug en teksteditor eller et andet tekstbehandlingsprogram end Microsoft Word og åbn "MitDok.xml" (det dokument, du gemte i trin 10 i første afsnit).

!!! **Hvorfor ikke bruge Word?** Nogle versioner af Microsoft Word erstatter automatisk to bindestreger efter hinanden med en lang bindestreg. Brug om muligt et tekstbehandlingsprogram med "almindelig tekst", f.eks. Simple Text (Mac OS) eller WordPad (Windows). Sådan kan du åbne filen i WordPad: Start WordPad, vælg **Filer** → **Åbn**, vælg **Alle dokumenter** på listen **Filtype**, vælg "MitDok.xml", og klik herefter på **Åbn**.

2 Tilføj nedenstående linie umiddelbart efter den første linie i XML-dokumentet. Denne linie angiver at XML-dokumentet skal vises med formen "wp.css". Sørg for, at du kopierer linien nøjagtigt.

```
<?xml-stylesheet type="text/css" href="wp.css"?>
```

3 Tilføj nedenstående linie umiddelbart efter den linie, du lige tilføjede. Denne linie angiver starten på en kommentar. Den bruges til at sætte den DTD, der er indlagt i filen, "ud af kraft". Dette er nødvendigt fordi visse web-browsere har problemer med at læse interne DTD'er.

```
<!--
```

4 Tilføj nedenstående linie umiddelbart efter linien "]>". Linien angiver slutningen på den DTD, du satte ud af kraft.

```
-->
```

5 Gem filen i ASCII-format (som almindelig tekst). Sørg for, at du ikke ændrer dokumentets navn.

6 Sørg for, at der findes en kopi af filen "wp.css" (leveret med Øvelser til avenue.quark) i samme mappe som "MitDok.xml".

7 Åbn "MitDok.xml" i web-browseren. De CSS-forme der er defineret i "wp.css" bliver anvendt på XML'en.

8 Hvis du vil undersøge, hvordan XML-filen ser ud når det vises med de CSS-forme der er defineret i "wp2.css", kan du åbne det igen i teksteditoren eller tekstbehandlingsprogrammet og ændre "wp.css" til "wp2.css" på følgende måde:

```
<?xml-stylesheet type="text/css" href="wp2.css"?>
```

9 Gem filen i ASCII-format (som almindelig tekst). Sørg for, at du ikke ændrer dokumentets navn.

10 Sørg for, at der findes en kopi af filen "wp2.css" (leveret med Øvelser til avenue.quark) i samme mappe som "MitDok.xml".

11 Klik på web-browserens opdateringsknap for at genindlæse XML-dokumentet. De CSS-forme der er defineret i "wp2.css" anvendes på XML'en. Bemærk forskellen og læg mærke til, hvordan den samme XML-fil bliver vist radikalt forskelligt, baseret på hvordan det præsenteres

12 Luk web-browseren og teksteditoren eller tekstbehandlingsprogrammet.

!!! Hvis du ser på WhitePaper-DTD'en i en teksteditor, og har lidt kendskab til hvordan DTD'er er konstrueret, kan du se, at `<whitePaper>`-elementet indeholder en attribut ved navn "xmlns:HTML", med den faste værdi "http://www.w3.org/TR/REC-html40". Denne attribut medtages automatisk i alle XML-filer baseret på denne DTD. Formålet hermed er at fortælle Microsoft Internet Explorer version 5.0 eller senere versioner, at dokumentet anvender HTML-namespace. Da dokumentet anvender HTML-namespace oversættes koden `<HTML:img>` til en HTML``-kode, og du kan derfor vise de billeder, som XML'en henviser til.

Tilføjelse af en regel til et kodningsregelsæt

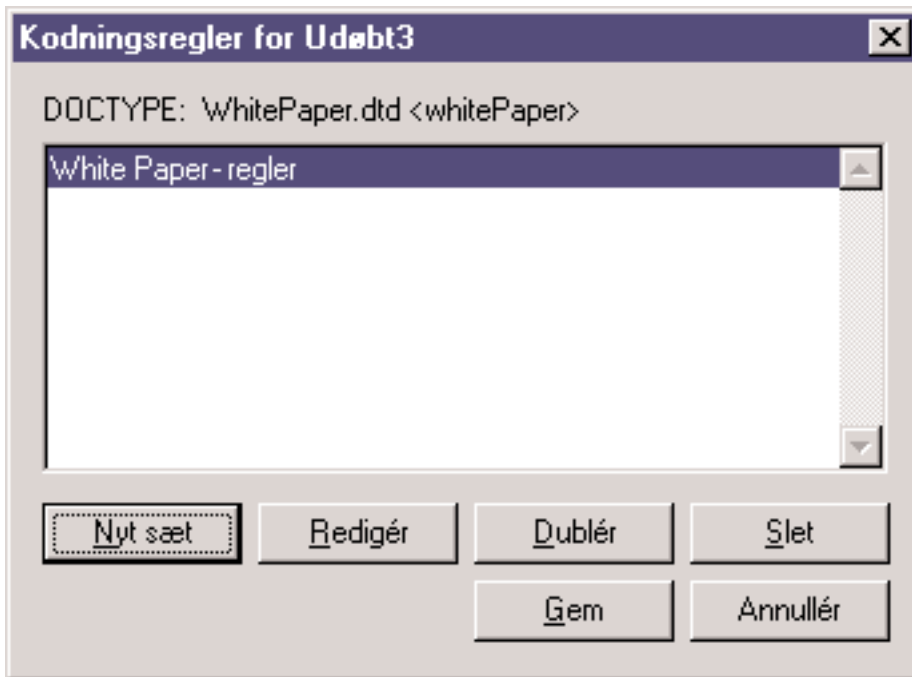
I første afsnit anvendtes et eksisterende kodningsregelsæt til automatisk kodning af det meste af et eksempeldokument. Men den juridiske tekst i slutningen af dokumentet blev *ikke* kodet. En af grundene til dette er, at det vedlagte kodningsregelsæt ikke indeholder en regel som håndterer tekst med afsnitsformen "Juridisk tekst". Dette afsnit viser, hvordan du kan løse problemet ved at tilføje en ny regel.

!!! Når vi i dette øvelsesdokument bruger udtrykket "afsnitsforme" henvises der til QuarkXPress Passport-forme, ikke til CSS-forme.

Sådan føjer du en ny regel til kodningsregelsættet "White Paper-regler":

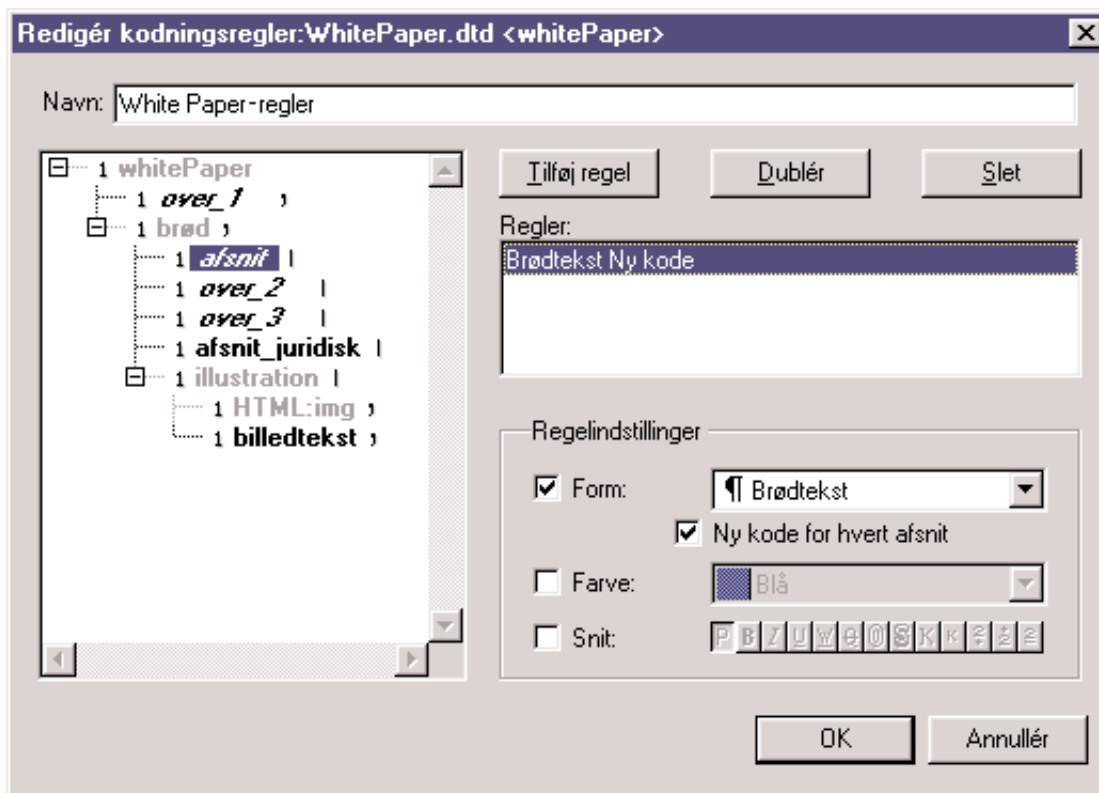
- 1 Start QuarkXPress Passport. Er QuarkXPress Passport allerede startet, skal du lukke alle åbne **XML-arbejdsområde**-paletter.
- 2 Åbn "EksempelDok.qxt", hvis det ikke allerede er åbent.
- 3 Vælg **Arkiv** (Mac OS) eller **Fil** (Windows) → **Ny** → **XML** for at oprette et nyt XML-dokument. Dialogen **Ny XML** vises. Markér **WhitePaper - Dansk** på listen **Skabelon**, og klik herefter på **OK**. Dette åbner paletten **XML-arbejdsområde**.

- 4 Vælg **Redigér** → **Kodningsregler**. Dialogen **Kodningsregler** vises.







Dialogen **Kodningsregler**

- 5 Vælg kodningsregelsættet “White Paper-regler”. (Det er det samme kodningsregelsæt som vi brugte i første afsnit). Klik herefter på **Redigér** for at redigere kodningsregelsættet ved hjælp af dialogen **Redigér Kodningsregler**.



Dialogen **Redigér kodningsregler**

- 6 Listen i dialogens venstre side viser WhitePaper-DTD'en i hierarkisk form. Du kan vise og skjule indholdet af containerelementer og attributter ved at klikke på symbolerne  og  (Mac OS) eller symbolerne  og  (Windows). Hvis du vil se de regler der tilsammen danner dette kodningsregelsæt, skal du klikke på elementnavnene “over_1”, “afsnit”, “over_2” og “over_3” på denne liste. De regler der gælder for de enkelte elementtyper vises på listen **Regler**, øverst til højre.



Hvad udfører kodningsregler? Med kodningsregler kan du fortælle avenue.quark, at indhold der er formateret på en bestemt måde skal kodes med en bestemt elementtype. Den ovenfor viste regel indikerer f.eks. at afsnit med afsnitsformen “Brødtekst” skal kodes som `<afsnit>`-elementer.

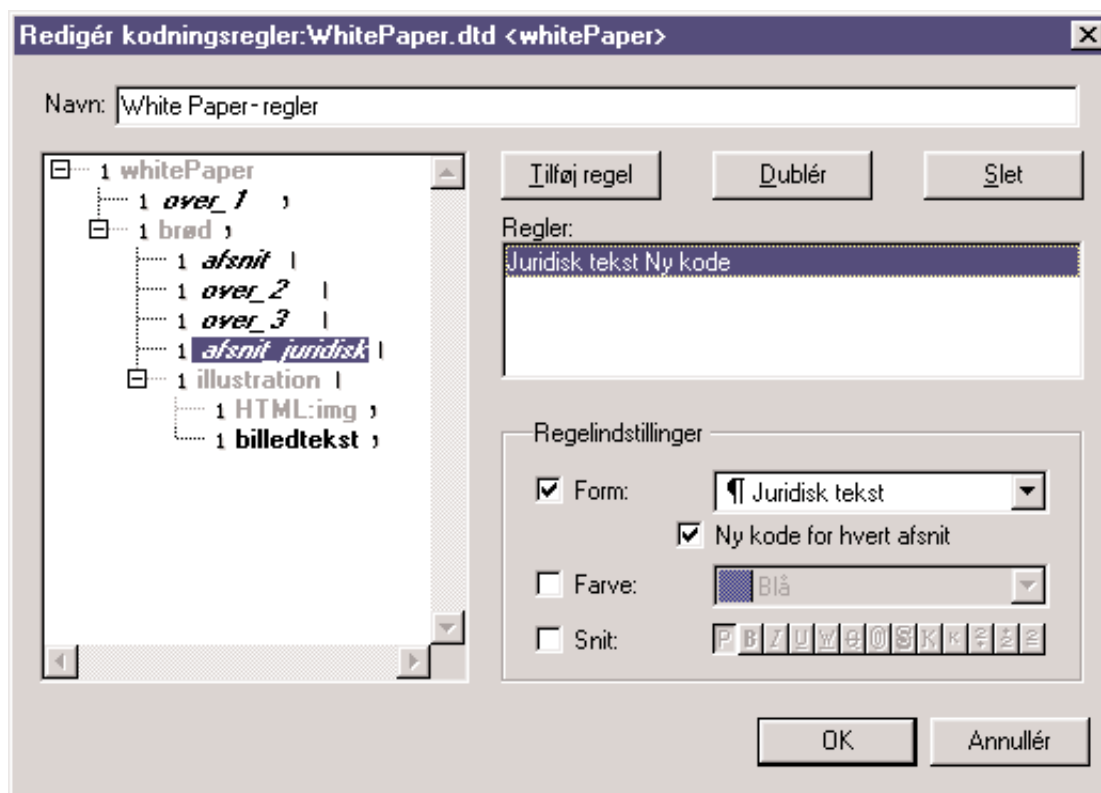
7

Klik på elementet `<afsnit_juridisk>` på listen til venstre, og klik herefter på **Tilføj regel** for at oprette en ny regel for denne elementtype. Dette indikerer at du vil oprette en ny kodningsregel for elementtypen `<afsnit_juridisk>`.



Hvorfor “afsnit_juridisk”? XML-koder kan – med få begrænsninger – navngives så forklarende, som du ønsker det. I denne DTD er et `<afsnit>`-element konstrueret til at indeholde et afsnit, og et `<afsnit_juridisk>`-element er konstrueret til at indeholde et afsnit med juridiske informationer.

- 8** Afkryds **Form**, og vælg **Juridisk tekst** i lokalmenuen **Form**. Afkryds **Ny kode for hvert afsnit**. Dette indikerer at du vil anvende elementtypen `<afsnit_juridisk>` på de afsnit der har tilknyttet afsnitsformen “Juridisk tekst”, og at hvert afsnit med formen “Juridisk tekst” skal kodes som separate elementer. Bemærk, at reglen på listen **Regler** (øverst til højre) opdateres, så den viser dine valg.



Dialogen **Redigér kodningsregler** med en ny regel for elementtypen `<afsnit_juridisk>`.

- 9** Klik på **OK** for at gemme de ændringer, du har foretaget i kodningsregelsættet “White Paper-regler”.

- 10** Luk dialogen **Kodningsregler** ved at klikke på **Gem**.

!!! Sørg for at den aktive **XML-arbejdsområde**-palet og det aktive QuarkXPress Passport-dokument er åbnet når du fortsætter – du får brug for dem i de følgende afsnit.

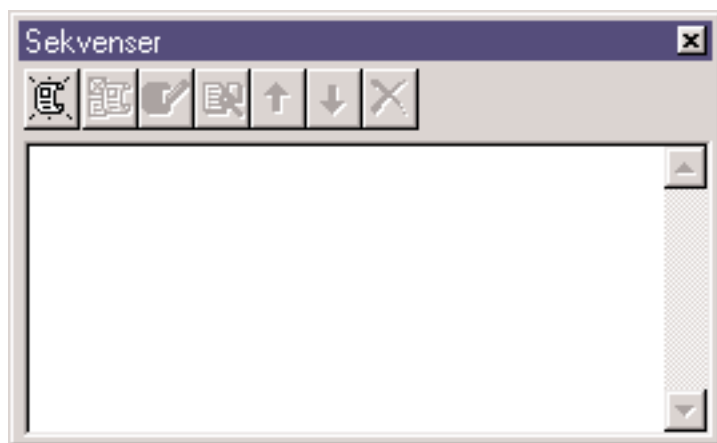
Kombinere tekstfelter til en sekvens

Du har nu allerede oplevet at avenue.quark kan hjælpe dig med hurtigt og nemt at kode og udtrække indholdet fra et enkelt tekstfelt. Vi skal nu gøre kodningsprocessen endnu hurtigere ved at kombinere et antal tekstfelter, der udgør brødtekstdelen i "white paper", til en *sekvens* og herefter kode denne sekvens som en enkelt enhed.

!!! Sekvenser er en del af funktionaliteten i Item Sequence QuarkXTensions-softwaren der leveres sammen med avenue.quark. Hvis du vil bruge sekvenser skal QuarkXTensions-softwaren Item Sequence være placeret i mappen "XTension" i QuarkXPress Passport-programmappen.

Sådan kombinerer du en et antal tekstfelter til en sekvens:

1 Åbn paletten **Sekvenser** ved at vælge **Syn** → **Vis sekvenser**.



Med paletten **Sekvenser** kan du oprette og navngive tekstfeltsekvenser.

2 Opret en ny sekvens ved at klikke på knappen **Ny sekvens**.

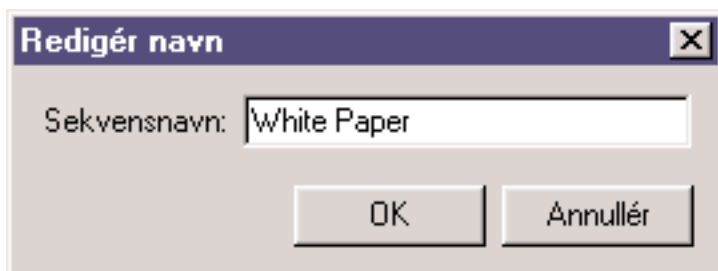


Knappen **Ny sekvens**

- 3** Klik på knappen **Redigér navn**. Dette åbner dialogen **Redigér navn**.
Skriv **White Paper** i feltet **Navn**, og klik herefter på **OK**.



Knappen **Redigér navn**



Med dialogen **Redigér navn** kan du omdøbe den markerede sekvens.

- 4** Marker det tekstfelt der indeholder overskriften (“avenue.quark
White Paper”) ved at klikke på det.
- 5** Føj tekstfeltet til sekvensen “White Paper” ved at klikke på knappen
Tilføj objekt.



Knappen **Tilføj objekt**

- 6** Markér det tekstfelt der starter med ordene “En række organisationer
har...”. Klik herefter på knappen **Tilføj objekt** for at føje det til sekvensen, umiddelbart efter overskriftsfeltet.

- 7 Flyt til side 2 og markér tekstfeltet nederst til højre på siden. Dette tekstfelt indeholder dokumentets juridiske tekst. Klik på knappen **Tilføj objekt** for at føje dette felt til sekvensen, umiddelbart efter det felt der rummer hovedparten af indholdet i white paper.



Paletten **Sekvenser** viser en sekvens som en række navngivne felter.

De tekstfelter der udgør "white paper" er nu kombineret, i den rigtige rækkefølge, til en enkelt sekvens ved navn "White Paper". Dette gør det lettere at arbejde med dem som en enhed. I næste afsnit skal du eksportere sekvensen i XML-format.

- !!! Sørg for at paletten **Sekvenser**, den aktive **XML-arbejdsområde**-palet samt det aktive QuarkXPress Passport-dokument er åbnet, når du fortsætter – du får brug for dem i det følgende afsnit.

Kodning med sekvenser og en ny regel

Nu, hvor du har tilføjet en ny kodningsregel for den juridiske tekst i slutningen af “white paper” og kombineret alle de tekstfelter der udgør “white paper”, til en enkelt sekvens, vil du kunne se, hvor nemt det er at kode hele den vigtige tekst i dokumentet i én arbejdsgang.

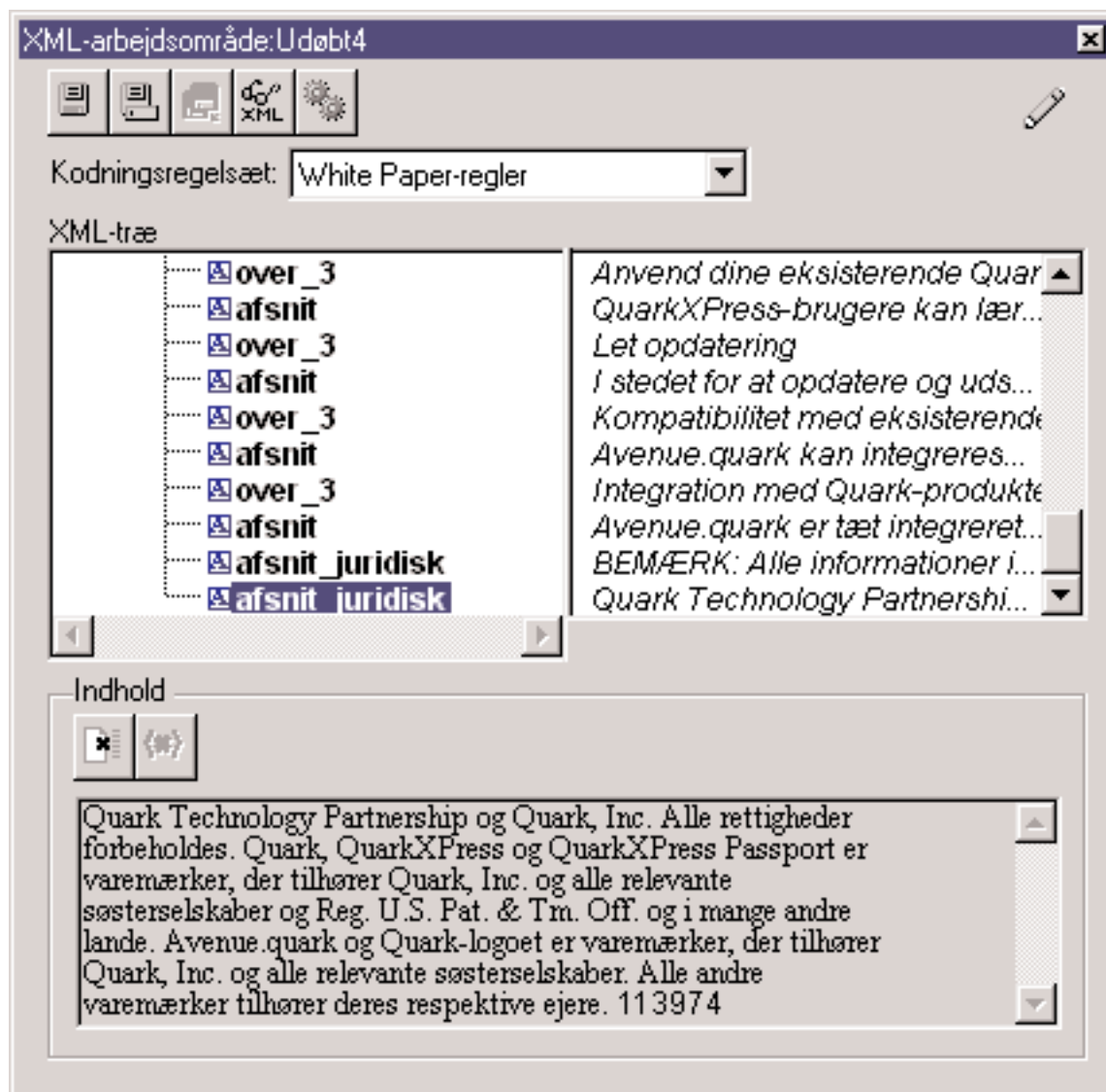
Sådan koder du sekvensen “White Paper”:

1 Vælg sekvensen “White Paper”.

2 Tryk på ⌘ (Mac OS) eller Ctrl (Windows), og træk sekvensen til elementet `<whitePaper>` på paletten **XML-arbejdsområde**. Avenue.quark koder automatisk indholdet i alle tre tekstfelter og opretter nye elementer, hvis det er nødvendigt.

 **Stop, det fungerede ikke!** Hvis der skete noget uventet da du trak sekvensen over på elementet `<whitePaper>` på listen **XML-træ**, skal du ikke tænke på det. Du skal blot lukke paletten **XML-arbejdsområde** og gå tilbage til trin 1.

- 3 Rul gennem listen XML-træ, indtil du når bunden. Bemærk nu, at avenue.quark har kodet alle afsnit med afsnitsformen “Juridisk tekst”.



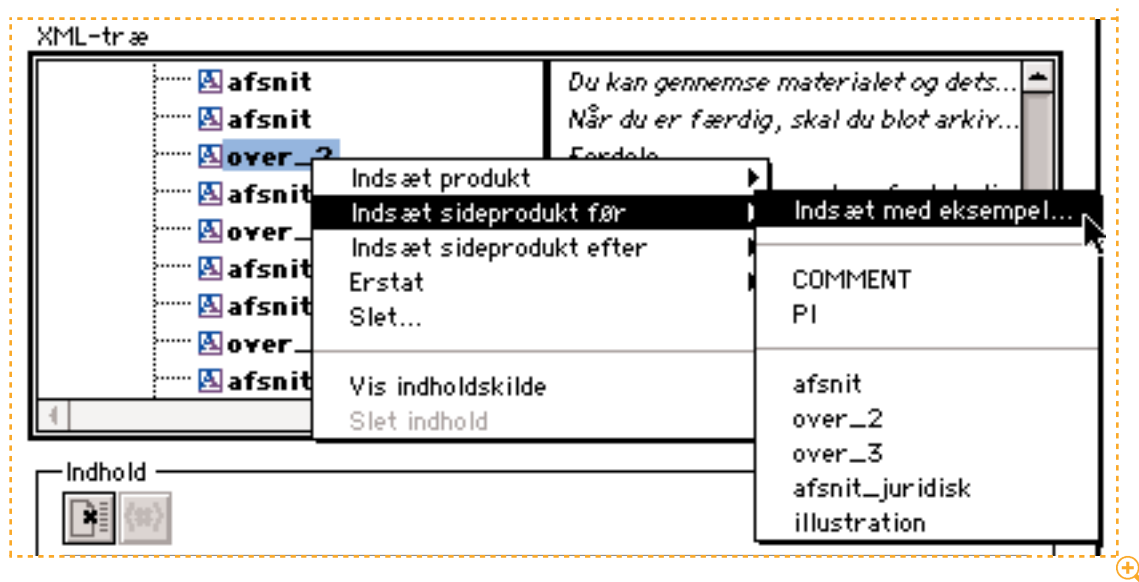
Et kodet dokument, hvor den juridiske tekst er kodet som <afsnit_juridisk>

- 4 Sørg for, at både QuarkXPress Passport-dokumentet “white paper” og det nye XML-dokument er åbnet. Du får brug for dem i det næste afsnit.

Kodning af et billede og en billedtekst

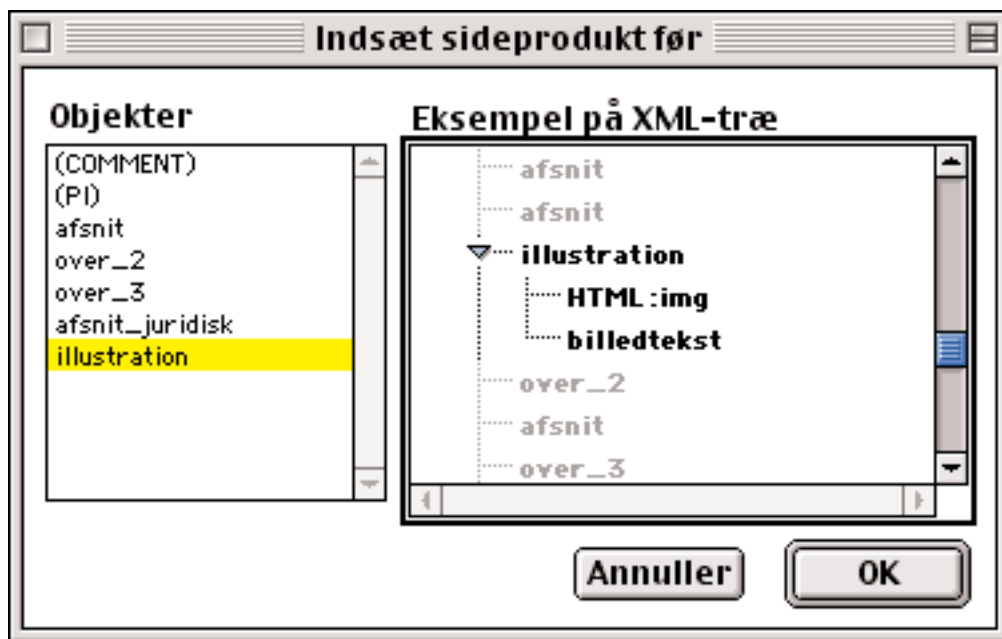
De eneste vigtige elementer i “white paper”, der endnu ikke har kodet, er billedet og billedteksten i det nederste venstre hjørne på side 1. I dette afsnit skal du føje et nyt `<illustration>`-element til XML-dokumentet og herefter føje billedet og dets billedtekst til dette elements produkter.

- 1 Rul gennem listen **XML-træ** på paletten **XML-arbejdsområde**, indtil du kan se elementet `<over_2>`, der indholder ordet “Fordele”.
- 2 Hold Ctrl nede, og klik (Mac OS), eller højreklik (Windows) på dette element for at vise en lokalmenu. Vælg **Indsæt sideprodukt før** for at vise en undermenu, og vælg herefter **Indsæt med eksempel**. Dialogen **Indsæt sideprodukt før** vises.



I lokalmenuen **XML-træ** kan du indsætte et objekt før det element der er markeret på listen **XML-træ**.

- 3** Vælg **illustration** på listen **Objekter** for at angive at du vil indsætte et `<illustration>`-element. Listen **Eksempel på XML-træ** viser et eksempel på, hvordan den aktuelle gren i XML-dokumentet kommer til at se ud, når du har indsat elementet `<illustration>`. Rul gennem listen **Eksempel på XML-træ**, indtil du kan se elementet `<illustration>` med fed skrift.

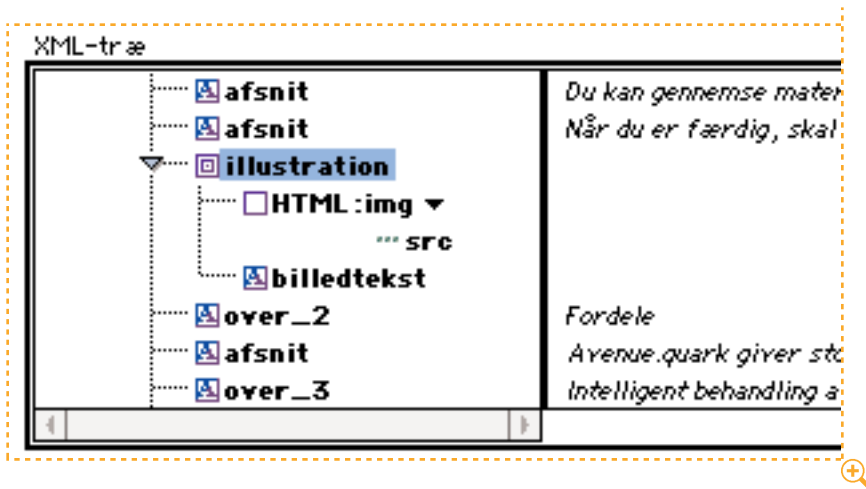


I dialogen **Indsæt sideprodukt før** kan du vise alle obligatoriske produkter af det objekt, du indsætter.

På listen **Eksempel på XML-træ** kan du vise alle obligatoriske produkter af det element, du vil indsætte. De obligatoriske produkter af elementet `<illustration>` er et `<HTML:img>`-element og et `<billedtekst>`-element.

- 4** Klik på **OK**. Elementet `<illustration>` tilføjes før det markerede `<over_2>`-element.

- 5 Klik på ▶ ud for elementet `<HTML:img>` på listen **XML-træ** for at vise den `src`-attribut, der er knyttet til det pågældende element.



Det netop indsatte `<illustration>`-element har to produkter: et `<HTML:img>`-element og et `<billedtekst>`-element. `<HTML:img>`-elementet har en attribut, der hedder `src`.

- 6 Vælg tekstfeltet under billedet nederst til venstre på side 1. Vælg herefter værktøjet **Objekt**, tryk på ⌘ (Mac OS) eller Ctrl (Windows), og træk tekstfeltet over på elementet `<billedtekst>` på listen **XML-træ**. Billedteksten kopieres til elementet `<billedtekst>`.
- 7 Vælg billedfeltet over billedtekstfeltet. Brug herefter værktøjet **Objekt** til at trække billedfeltet til attributten med navnet "src" på listen **XML-træ**. Billedfilens navn kopieres til `<HTML:img>`-elementets `src`-attribut.



Hvad var det, der skete? Du har lige bedt avenue.quark om at anbringe billedfilens navn i `<HTML:img>`-elementets `src`-attribut. Kun filens navn indsættes – ikke selve billedet. Det betyder, at billedfilen skal befinde sig i samme mappe som XML-filen, når denne vises, ellers kan billedet ikke ses i browseren.

8 Klik på knappen **Arkivér** (Mac OS) eller **Gem** (Windows) på paletten **XML-arbejdsområde**, og gem dokumentet i mappen “avenue.quark Øvelser” på harddisken under navnet “MitDok2.xml”. (igen skal du bibeholde **UTF8** i lokalmenuen **Tegnkodning**, og afkrydse **Gem XML som standalone**.)



Knappen **Arkivér** (Mac OS) eller **Gem** (Windows)

9 Brug den fremgangsmåde, der er forklaret i “Vis en XML-fil med CSS-forme” (i afsnittet “Visning af udtrukket XML-indhold”) for at vise “MitDok2.xml”. Sørg for at filen “diagram1.gif” findes i samme mappe som “MitDok2.xml”, hvis du vil vise billedet.



Du har nu set nogle få af de ting, som avenue.quark kan udføre. Læs “En vejledning til avenue.quark” (“En vejledning til avenue.quark.pdf”), der leveres sammen med Øvelser til avenue.quark hvis du ønsker flere informationer.

©2000 Quark Technology Partnership og Quark, Inc mht. indholdet og arrangementet af dette materiale.
Alle rettigheder forbeholdes.

©2000 Quark Technology Partnership og Quark, Inc mht. teknologien. Alle rettigheder forbeholdes.
Patenter er under udarbejdelse i USA og andre lande.

Quark, QuarkXPress, QuarkXPress Passport og QuarkXTensions er varemærker, der tilhører Quark, Inc.
og alle relevante tilknyttede selskaber, Reg. U.S. Pat. & Tm. Off. og i mange andre lande. Avenue.quark
og Quark-logoet er varemærker, der tilhører Quark, Inc. og alle relevante tilknyttede selskaber.

Microsoft er et registreret varemærke, der tilhører Microsoft Corporation.

Alle andre varemærker tilhører deres respektive ejere.