

# Technische documentatie Edubox 2.0 Webplayer

Citation for published version (APA):

Van der Vegt, W., & Brouns, F. (2001). *Technische documentatie Edubox 2.0 Webplayer*.

## Document status and date:

Published: 01/06/2001

## Document Version:

Peer reviewed version

## Please check the document version of this publication:

- A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
- The final author version and the galley proof are versions of the publication after peer review.
- The final published version features the final layout of the paper including the volume, issue and page numbers.

[Link to publication](#)

## General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license above, please follow below link for the End User Agreement:

<https://www.ou.nl/taverne-agreement>

## Take down policy

If you believe that this document breaches copyright please contact us at:

[pure-support@ou.nl](mailto:pure-support@ou.nl)

providing details and we will investigate your claim.

Downloaded from <https://research.ou.nl/> on date: 16 Jul. 2023

Open Universiteit  
[www.ou.nl](http://www.ou.nl)



**Onderwijstechnologisch expertisecentrum OTEC  
Open Universiteit Nederland**

# **Technische Documentatie Edubox 2.0 Webplayer**

**Webplayer**

**OTEC 2001/9**



## **OTEC werkdocumenten**

De Open Universiteit Nederland ontwikkelt en verzorgt open hoger afstandsonderwijs en is tevens een centrale partner in het consortium voor vernieuwing van het Hoger Onderwijs. Onderwijstechnologische vernieuwingen krijgen daarbij speciale aandacht. Binnen de Open Universiteit Nederland is de onderwijskundige en onderwijstechnologische expertise samengebracht in het Onderwijstechnologisch expertisecentrum (OTEC). Dit centrum vervult taken in het kader van ontwikkeling, vernieuwing, onderzoek en evaluatie van het onderwijs van de Open Universiteit Nederland en haar consortiumpartners. Deze taken worden veelal uitgevoerd in nauwe samenwerking met directoraten en faculteiten van de OUNL en/of samenwerkingspartners in het consortium.

De werkzaamheden van het OTEC leiden regelmatig tot producten, zoals voorlopige onderzoeksresultaten, strategische stellingnames en overwegingen op projektniveau, functionele specificaties van informatiesystemen, en dergelijke. Deze worden door het OTEC vastgelegd en onder de aandacht gebracht in een reeks werkdocumenten waarvan het voorliggende deel uitmaakt.

Naast deze reeks werkdocumenten geeft het OTEC een reeks rapporten uit met een meer geformaliseerd en/of afgerond karakter.

De OTEC rapporten en sommige werkdocumenten kunnen worden besteld bij:

Open Universiteit Nederland  
secretariaat OTEC  
Postbus 2960  
6401 DL Heerlen  
Tel. 045-5762406 of 5762942  
Fax. 045-5762802

of opgehaald via Internet: <http://www.ou.nl/OTEC>

Onderwijstechnologisch expertisecentrum (OTEC)  
Open Universiteit Nederland

**Technische documentatie Edubox 2.0 Webplayer**  
Webplayer

## Colofon

Titel:	Technische documentatie Edubox 2.0 Webplayer
Subtitel:	Webplayer
Auteurs:	Wim van der Vegt, Francis Brouns
Projectleiding:	Ghislain Rodenburg
Projectondersteuning:	Mieke Haemers
Uitgifte:	OTEC
Datum druk:	24 maart 2004

© 2004, Onderwijstechnologisch expertisecentrum,  
Open Universiteit Nederland, Heerlen.

Behoudens uitzonderingen door de wet  
gesteld mag zonder schriftelijke toestemming  
van de rechthebbende(n) op het auteursrecht  
niets uit deze uitgave worden verveelvoudigd  
en/of openbaar gemaakt door middel van  
druk, fotokopie, microfilm of anderszins,  
hetgeen ook van toepassing is op de gehele of  
gedeeltelijke bewerking.

# Inhoudsopgave

<b>Inleiding .....</b>	<b>7</b>
<b>Functies van de Tree component.....</b>	<b>7</b>
Server-zijde XSL Transformatie.....	8
Lokalisatie .....	9
Functies van de Perftool component .....	10
<b>Stijlen in de Edubox afspeler .....</b>	<b>11</b>
Cascading stylesheets.....	11
<i>CSS eigenschappen .....</i>	<i>11</i>
<i>Selectoren .....</i>	<i>11</i>
<i>Toekenning van waardes, inheritance en cascade.....</i>	<i>12</i>
<i>HTML elementen .....</i>	<i>12</i>
<i>Positionering .....</i>	<i>13</i>
Webdocumenten .....	13
Schermopbouw .....	14
Opmaak inhoud .....	15
<b>Bijlage 1. Files.....</b>	<b>17</b>
<b>Bijlage 2. Architectuur player. ....</b>	<b>18</b>
<b>Bijlage 3. Statische publicatie.....</b>	<b>19</b>
<b>Bijlage 4. Elementen die dynamisch worden vormgegeven .....</b>	<b>22</b>





## Inleiding

De Edubox player bestaat uit een aantal asp pagina's, en (voorlopig) een tweetal Com Objecten. Verder wordt er gebruik gemaakt van een aantal commerciële Com Objecten als AspUpload en de Microsoft Scriptcontrol en een java applet ten behoeve van remote scripting. De zelf ontwikkelde Com Objecten zijn tree.dll en perftools.dll

Tree.dll wordt aan de serverzijde aangeroepen vanuit een aantal asp pagina's en aan de clientzijde vanuit het player.js script (genereerd en gepersonaliseerd door player.asp). Deze laatste aanroepen lopen niet direct maar altijd via remote scripting (o.a. remote.asp), zodat de feitelijke aanroep van de component weer op de web server zelf plaatsvindt.

Perftools.dll is een ComObject dat serverzijde in asp scripts gebruikt kan worden voor accurate timing ten behoeve van performance analyse.

Voor een schema van de objecten en hun onderlinge interactie zie appendix 2.

Tree.dll en Perftools.dll kennen beiden geen user-interface.

Installatie gebeurt doormiddel van het toevoegen van de com objecten aan een package van de transactie server. Door nog steeds onbekende oorzaak is het nodig dat tree.dll onder een admin account werkt.

In het hoofdstuk stijlen in de Edubox afspeler wordt beschreven op welke wijze de vormgeving gekoppeld wordt in de Edubox player middels CSS.

## Functies van de Tree component

Het bevat een aantal functies die in een volgende versie van de Edubox player wellicht uit elkaar gerafeld moeten worden in kleinere functionele blokken. Met name locatisatie, debugging en klant specifieke instellingen zorgen voor wat vervuiling qua object model.

### **Laden XML files.**

XML/XSL files worden rechstreeks vanaf het filesystem geladen, relatief ten opzichte van de aanroepende webpagina's locatie. Dit gebeurt door het converteren van een web-relatief path naar een disk absoluut path.

Voor het parsen van XML/XHTML wordt gebruik gemaakt van het Microsoft DOM Object. Dit object leest XML files en zet deze om in een hiërarchische object structuur die gemakkelijk doorlopen kan worden.

Zowel content als menu's zijn in XML gecodeerd. Hierdoor is automatische verwerking gemakkelijk.

De naamgeving van de menu's is als volgt:

Naam	Functie
Activity.xml	Activiteiten menu
Preferences.xml	Tools menu
Env_*.xml	Environment menu's

**Tabel 1: Menu naamgeving**

### **Personaliseren van XML files door middel van JScript.**

Tree.dll gebruikt een scriptcontrol object als toegang tot Jscript. Dit JScript (personalize.js) bevat functies die de database toegang afschermt. In tree.dll wordt nergens de database rechtstreeks aangeroepen.

Personalisatie vindt plaats door het wijzigen van de waarden van een aantal variabelen in de script engine waardoor het personalize.js script beschikt over onder andere de rundeelnameID van de gebruiker.

De personalisatie is verschillend voor menu's en content.

Voor de menu XML files zijn er drie verwerkingslagen nodig, elk werkend op een van de attributen van de treenodes. Twee verwerkingslagen worden achterwaarts uitgevoerd en een voorwaarts. Waar mogelijk worden zo groot mogelijke stukken xml verwijderd als de expressies daartoe aanleiding geven. Zodoende krijgt de student nooit meer informatie dan hij/zij mag zien.

De output van deze translatie moet valide XHTML zijn.

**\*\*DTD\*\***

Het personaliseren van content XML files gebeurt in een enkele pass en wordt gecombineerd met conversie naar XHTML. Hiervoor wordt de objectboom doorlopen en on-the-fly vertaald naar XHTML tags. Voor die tags die personalisatie behoeven worden functies uit de ScriptControl gebruikt. Alle toegang tot de database vindt vanuit het Personalize.js Script in de ScriptControl plaats.

**\*\*DTD\*\***

### **Server-zijde XSL Transformatie**

De Tree Component heeft de mogelijkheid om ingelezen Menu XML files geladen in een Dom Object te transformeren met behulp van een xsl tansformatie naar XHTML. Hierdoor kunnen menu's worden aangepast zonder dat de Tree Component opnieuw vertaald hoeft te worden. Let op dat voor content dit te complex was en dat dit nu hardcoded in de Tree Component gebeurt.

Omdat de server-zijde conversie wordt aangestuurd vanuit de client-zijde code in player.js en inbox.js (altijd via remote scripting) van de web-pagina's kan er dynamisch een andere translatie gekozen worden. Op die manier kan het menu eenvoudig op verschillende wijzen getoond worden (in Edubox 2.0 versie als boom en als lijst).

## Lokalisatie

Localisatie is een service die zowel binnen de component wordt gebruikt als naar buiten toe wordt aangeboden. Dit gebeurt door het inlezen van een language.ini file met vertaalinstructies. Deze ini file staat in dezelfde directory als de opgevraagde content. Hierdoor kan er per run een taal ondersteund worden.

De Language.ini is specifiek voor elke publicatie en kan maar een taal tegelijk bevatten. De **[Tree]** sectie bevat telkens items met de string id gevolgd door de tekst van string. Deze sectie wordt alleen binnen de tree component gebruikt voor on-the-fly vertaling. De **[Scripts]** sectie bevat paren string id's en waarden voor vertaling van scripts en asp-pagina's. Deze strings zijn via een functie van de tree component extern te benaderen.

```
[Tree]
;delphi com object
nvd=geen waarde gedel nieerd
err=niet beschikbaar
search=zoek
name=naam

[Scripts]
;index.asp
btn_information=Informatie
btn_activities=Activiteiten
btn_communication=Communicatie
btn_preferences=Opties
```

**Figuur 1. Language.ini sample.**

### Client specifieke links.

De Tree.ini file met klant specifieke settings staat in de directory waar de dll zelf staat. Deze file wordt automatisch aangemaakt/ge-update voor nieuwe klanten (lees virtual websites), maar de waarden van de strings moeten voor elke klant aangepast worden. Een virtuele webserver wordt geïdentificeerd and de hand van de metabase id. Per klant/virtuele website is het nodig om de database connectstring aan te passen en de url van de portal in te geven.

```
[/LM/W3SVC/4]
database=Dsn=dossier;User ID=ounl_dos20;Password=ounl_dos20;Initial
Catalog=ounl_dos20;
portal=http://int.portal.ou.nl/studie-alg-werkplek/
```

**Figuur 2. Tree.ini sample**

## Debugging

Via een functie kan het tracing level van de Tree component ingegeven worden. Standaard wordt er erg veel gelogd in de Windows nt eventlog. Dit is nodig voor debuggen van code en traceren van fouten in EML.

Omdat de Tree component logt in de Windows eventlog is het nodig de component te registreren als eventlog message source en tijdens de compilatie een message-tabel toe te voegen aan de executable. Eventlog Message Source Registratie gebeurt tijdens het gewone registreren van de component b.v. met regsrv32.exe. Voor dit registreren is een administratief account nodig.

## **Functies van de Perftool component**

Toegang geven tot de performancecounters van Windows, waarmee sub-milliseconden timing mogelijk is. De component bevat naast de tijdmetingen zelf ook een aantal functies voor het converteren van de resultaten van de performancecounters naar bruikbare eenheden als milli-seconden.

## Stijlen in de Edubox afspeler

De Edubox afspeler maakt gebruik van cascading stylesheets (CSS). Hiermee kunnen gestructureerde documenten vormgegeven worden.

### Cascading stylesheets

Cascading stylesheets (CSS) is een taal om opmaak te koppelen aan gestructureerde documenten (bijv HTML pagina's, XML applicaties). Het is ontwikkeld door W3C (The World Wide Web Consortium) en CSS2 is sinds 12 mei 1998 geaccepteerd als een W3C Recommendation.

CSS geeft een veel grotere controle over de opmaak van webpagina's dan met HTML mogelijk is, doordat veel meer eigenschappen ingesteld kunnen worden. Bovendien kan via CSS opmaak gescheiden worden van inhoud. CSS kan gebruikt worden voor alle gestructureerde documenten (dus niet alleen HTML, maar ook bijv XML).

Edubox hanteert de CSS2 Specificatie. Het grootste verschil met CSS1 bestaat uit een uitbreiding van de functionaliteit (met name adressering van meerdere media, positionering en een grotere controle over het visual formatting model) en enkele correcties van fouten in CSS1. De volledige specificaties staan op de website van W3C (<http://www.w3.org/TR/REC-CSS2/>). Zie <http://www.w3.org/Style/> voor algemene informatie over stylesheets en <http://www.w3.org/Style/CSS/> voor algemene informatie over CSS.

### CSS eigenschappen

In de CSS Specificatie zijn CSS eigenschappen gedefinieerd. Van iedere eigenschap is de naam vastgelegd, plus de volgende kenmerken:

- toegestane waarde en de syntax
- initiële waarde
- de elementen waarop deze eigenschap van toepassing is
- of het overerfd wordt
- hoe procentuele waardes geïnterpreteerd worden
- op welke media de eigenschap van toepassing is.

Het is mogelijk om meerdere eigenschappen in een korte notatie weer te geven, waarmee meerdere eigenschappen in een keer gezet kunnen worden. Eigenschappen die niet expliciet genoemd zijn in de korte notatie gebruiken de initiële waarde zoals die bij de respectievelijke eigenschappen is beschreven; bijv font is een korte notatie voor font-style, font-variant, font-weight, font-size, line-height en font-family.

### Selectoren

Door middel van de selectoren en eigenschappen kunnen diverse delen van een document of 'user agent' geadresseerd worden:

- elementen in de document boom en bepaalde relaties tussen die elementen
- attributen van elementen in de boom en de waardes van die attributen

- enkele delen van de inhoud van een element (bijv eerste regel)
- elementen in een bepaalde staat
- aspecten van het canvas waarop het document afgebeeld wordt
- enige systeem informatie (bijv cursor).

De opmaak wordt vastgelegd in CSS regels. Deze regels kunnen in een extern 'stylesheet' staan en dan aan meerdere HTML documenten gekoppeld worden; in een stijlblok in het begin van een HTML document staan; of als inline stijl toegevoegd aan een (HTML) element.

Externe stylesheets bieden het voordeel dat ze aan meerdere documenten gekoppeld kunnen worden en dat vormgeving gewijzigd kan worden zonder de HTML documenten aan te hoeven passen.

Een CSS regel bestaat uit twee gedeelten: een selector en een declaratie. De declaratie bestaat ook weer uit twee delen: de eigenschap en de waarde.

Een selector refereert naar elementen in de document boom en kan bestaan uit een enkel element of een context-afhankelijke specificatie.

Het is ook mogelijk te refereren naar een specifieke waarde van een bepaald attribuut. In HTML is dit vaak het attribuut class=. Ook kan gerefereerd worden naar attributen van type ID. Deze identifier moet uniek zijn in het document.

Verder biedt CSS de mogelijkheid om content te benaderen die niet vastgelegd is in een element, zogenaamde pseudo-elementen en pseudo-classes (bijv. de eerste regel, eerste woord, eerste afstammeling (first-child)).

Ook zijn er pseudo-classes en dynamic pseudo-classes voor hyperlinks.

## Toekenning van waardes, inheritance en cascade

De waarde die een eigenschap uiteindelijk krijgt wordt bepaald door verschillende factoren. Waardes kunnen berekend zijn, of gespecificeerd zijn. Sommige waardes worden geërfd (inheritance). Ook de cascade heeft invloed op de waarde.

Stijlregels kunnen geïmporteerd worden vanuit andere stylesheets. Stylesheets kunnen afkomstig zijn van de auteur, de gebruiker of de user agent (webbrowser). Al deze stylesheets zullen overlappen.

Er zijn regels om te bepalen welke eigenschap voorrang krijgt. Auteurs- en gebruikers-stylesheets hebben altijd voorrang boven die van de browser. In beginsel hebben auteurs-stylesheet voorrang boven stylesheets van de gebruiker, tenzij de gebruiker een !important declaratie gebruikt in het stylesheet.

Eigenschappen uit geïmporteerde regels worden toegekend, afhankelijk van volgorde van importeren. Regels uit het stylesheet dat een ander stylesheet importeert, hebben altijd voorrang.

## HTML elementen

De interpretatie van de CSS eigenschappen hangt o.a. af van het type element. HTML kent element op blokniveau en inline elementen. Elementen op blokniveau worden gescheiden door witruimte (bijv p, div, ul, ol, li, table, form). Inline elementen blijven op dezelfde regel en worden niet gescheiden door witruimte (bijv, a, img, span). Een aparte klasse van elementen zijn de replaced elementen. Dit zijn elementen die vervangen worden door de inhoud waarnaar het element verwijst (bij img, input, textarea, select).

## Positionering

Ieder element wordt omgeven door een rechthoekig gebied, de box. Deze box bestaat uit de inhoud, padding, border en margin. Blok-elementen kennen slechts 1 box, inline-elementen worden gesplitst over meerdere blokken indien deze niet op een regel passen.

Normaal worden blok-elementen onder elkaar en inline-elementen naast elkaar geplaatst (normal flow). De verticale margin bij blok-elementen schuiven daarbij in elkaar. De positie van de elementen is dan relatief ten opzichte van elkaar.

Indien elementen gepositioneerd worden, kan het betekenen dat ze uit de 'normal flow' gehaald worden. Bij relatieve positionering wordt de positie bepaald ten opzichte van de gebruikelijke positie van het element. Bij absolute positionering is er geen relatie meer met de normale positie van het element. De positie wordt dan bepaald ten opzichte van het meest dichtsbijzijnde bovenliggende element dat gepositioneerd is (uiteindelijk Body, indien geen ander voorgaand element gepositioneerd is).

Hierdoor kan het gebeuren dat elementen elkaar gaan overlappen. De normale volgorde is van onder naar boven, dit is de volgorde waarin ze in een document verschijnen. De z-index eigenschap kan gebruikt worden om deze volgorde te beïnvloeden.

## Webdocumenten

De pagina-indeling voor de afspeler is vastgelegd in de openingspagina index.asp. Hierin zijn gebieden gedefinieerd voor de verschillende componenten, zoals de diverse menu's (activiteiten, omgeving, opties, inbox), bovenste iconenbalk, de inhoud en 'popup' informatie. Deze gebieden worden gepositioneerd door middel van CSS stijlen. De diverse EML elementen worden getoond in het relevante gebied. Opmaak van de inhoud is vastgelegd in een tweede CSS stylesheet. In index.asp worden beide CSS stylesheets (player.css en content.css) aangeroepen.

Bij het publiceren van EML documenten worden een aantal EML elementen omgezet in HTML elementen en wordt eventueel het attribuut class gezet. Deze statische opmaak wordt toegekend via het 'Omnimark stylesheet'. Het betreft de Metadata elementen, Figure, Formula (plus Figure-source/Formula-source), lijsten, de 'achtergrond' van alle componenten, de kopjes voor de elementen binnen Activity-description, Emphasis, EML-ref, en de elementen die refereren naar een bron op het internet.

Dit Omnimark stylesheet is klantspecifiek. Momenteel bestaan er twee implementaties: voor de OUNL en voor de HHM. Er zijn geen structurele verschillen tussen beide implementaties die invloed hebben op de CSS stylesheet. Voor de HHM wordt een plaatje toegevoegd bij bepaalde objecten.

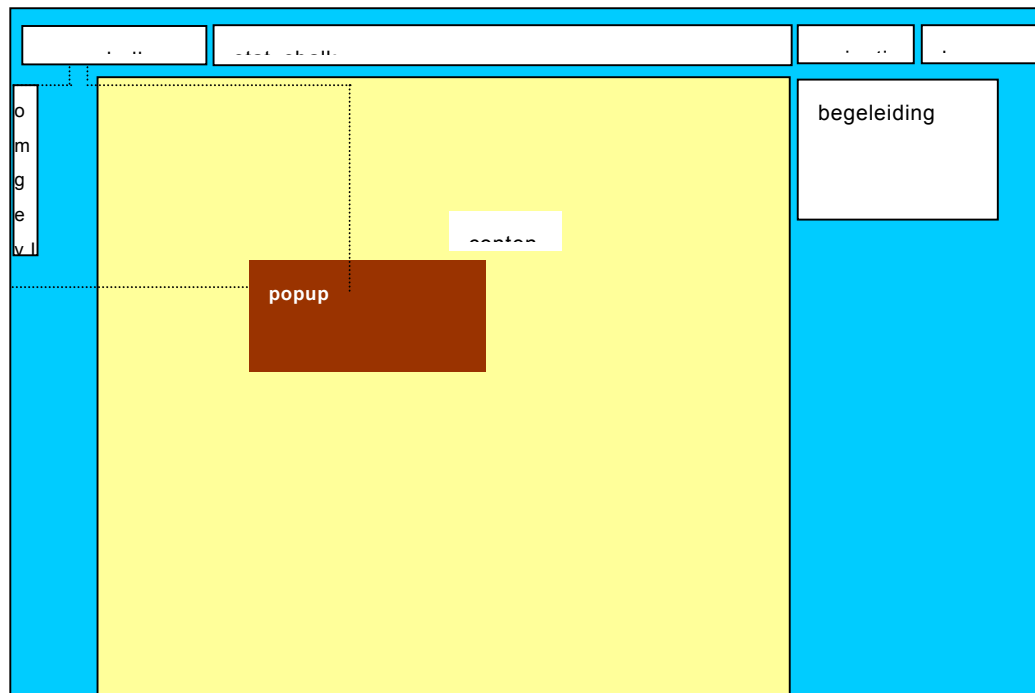
Alle andere EML elementen worden dynamisch afgehandeld en CSS eigenschappen worden toegekend en/of gewijzigd tijdens het ophalen van de pagina.

Er zijn aparte CSS stylesheets voor de pagina-indeling, voor de opmaak van de feitelijke inhoud, voor de 'Inbox' en voor de 'Viewswitch tool'.



## Schermbouw

Er wordt uitgegaan van een scherm van maximaal 800x600px. Dit is inclusief eventuele menubalken en scrollbars van de browser. Effectief is het scherm 784px breed omdat er vrijwel altijd een scrollbar aanwezig zal zijn aan de rechterkant van het scherm en ongeveer 570px hoog.



In het scherm zijn enkele functionele gebieden te onderscheiden.

Het grootste gebied is natuurlijk het deel waar de inhoud (content) komt te staan. De inhoud is afhankelijk van de keuze die gemaakt wordt in het menu.

Bovenin staat een balk. Het eerste deel van de balk bevat de diverse menu-opties, het tweede deel fungeert als statusbalk, het derde deel bevat navigatieknoppen, en het vierde deel het Edubox logo.

Menu's worden geactiveerd middels knoppen in de menubalk, met uitzondering van het omgevingsmenu dat links staat. Het omgevingsmenu is slechts beschikbaar nadat een activiteit gekozen is. Het feitelijke menu (in de figuur is dit gebied aangegeven met een stippellijn) valt dan deels over het inhoudsdeel heen. De inhoud bij het gekozen menu wordt getoond in het 'content' deel.

Het rechterdeel wordt momenteel slechts gebruikt voor het interface voor docenten/begeleiders. Hier moet een docent/begeleider de te begeleiden studenten kiezen.

Verder wordt een 'popup' gebruikt voor het tonen van metadata, ingangseisen en leerdoelen. Deze popup valt over de content.

Deze gebieden worden gespecificeerd in index.asp en gepositioneerd en vormgegeven middels CSS eigenschappen. De CSS regels voor de functionele gebieden staan in een apart stylesheet (player.css).

Behalve de positionering en opmaak van de diverse functionele gebieden, wordt ook vastgelegd wat de opmaak is van de menu's (kleur van de hyperlinks, regelafstand van de lijsten, opmaak van de tabel die gebruikt wordt voor de menubalk) en de basisopmaak voor het element Body.

Het Body element is de basis voor een webpagina. Alle andere elementen zijn child elementen van Body. Daarom is de basis opmaak voor Body reeds in player.css gespecificeerd.

Er is een achtergrondkleur gespecificeerd waarop alle functionele gebieden getekend worden. De tekstkleur is afgestemd op de achtergrondkleur. De tekst wordt weergegeven in Verdana. Arial of Helvetica zijn gespecificeerd als fonts die gebruikt moeten worden als Verdana niet beschikbaar is. Fontgrootte is in een relatieve eenheid (em) gespecificeerd, zodat gebruikers de mogelijkheid hebben dit naar believen aan te passen. De regelafstand is ruimer gesteld om de leesbaarheid te vergroten.

De overige functionele gebieden vallen hierover heen en opmaak hiervan is relatief ten opzichte van Body.

De grootte van de meeste gebieden wordt dynamisch aangepast aan de grootte van het browservenster.

Geen van de gebruikte stijlblokken mag verwijderd worden uit het stylesheet. Ook de stijlregels kunnen niet aangepast worden zonder consequenties voor het interface en onderliggende code. Wel kunnen enkele CSS stijleigenschappen aangepast worden. Kleur, font (soort, grootte, kleur) kunnen zonder meer aangepast worden. Grootte en positie van de gebieden kan in beperkte mate aangepast worden, rekening houdend met de afhankelijkheden tussen de verschillende gebieden.

### **Opmaak inhoud**

In content.css wordt de opmaak van alle elementen van de inhoud bepaald. Dit betreft de standaard HTML elementen als H1-H6, de hyperlinks (A), PRE, alle instanties van elementen waarvoor het class attribuut is gezet tijdens de publicatie, als de elementen waarvan het class attribuut dynamisch wordt toegekend.

Ook hier geldt dat geen van de gedefinieerde stijlblokken of eigenschappen verwijderd kunnen worden.

Eigenschappen als font, fontgrootte, font-stijl, tekstkleur, achtergrondkleur, kunnen zonder problemen aangepast worden.

Eigenschappen als width, height, display, visibility mogen niet uit de stijlblokken verwijderd worden, omdat deze soms dynamisch aangepast worden, of omdat ze gebruikt worden om de pagina aan te passen aan de grootte van het browservenster.

Kleine wijzigingen in maat en marges zijn toegestaan, maar substantiële wijzigingen kunnen de indeling van de pagina beïnvloeden.

Tabellen worden door veel browser als aparte objecten gezien. Opmaak van tabellen moet dan ook expliciet aangegeven worden. In Edubox worden tabellen niet alleen gebruikt om auteurstabellen te tonen, maar ook om bepaalde objecten op de pagina te positioneren. Hier

moet rekening mee gehouden worden in het CSS stylesheet. De volgorde van de stijlregels voor de diverse tabellen is bepalend voor de uiteindelijke vormgeving.

De beide klantimplementaties (OUNL en HHM) gebruiken dezelfde stijlregels, met andere waarden voor enkele van de eigenschappen. Voor de HHM zijn de kleuren voor de achtergronden en de tekst, het font voor kopjes en titels aangepast aan de huisstijl van de Hoge Hotelschool Maastricht.

## Bijlage 1. Files

Naam	Bevat
tree.dpr tree.res tree.cfg	Delphi Project file
tree_tlb.pas tree.tlb	Object definitie
unit1.pas	Object Implementatie
Edubox_event.mc	Eventlog message definitie

**Tabel 2. Delphi 5.0 sources van tree.dll.**

Deze dll maakt gebruik van ExcMagic, een tool om bij exceptions een gedetailleerde foutmelding inclusief callstack te verkrijgen. Deze tool vereist dat Turbo Debugger Debugging informatie in de dll aanwezig is. Omdat deze informatie meer bevat dan noodzakelijk en de dll erdoor in grootte nogal toeneemt, is het mogelijk om met behulp van een utility tdspace, na compilatie, alle niet gebruikte debug informatie uit de dll te halen. Hierdoor krimpt de dll van rond de 2 MB naar rond de 500KB.

Naam	Bevat
perftools.dpr perftools.res perftools.cfg	Delphi Project file
tools_tlb.pas tools.tlb	Object definitie
tools.pas	Object Implementatie

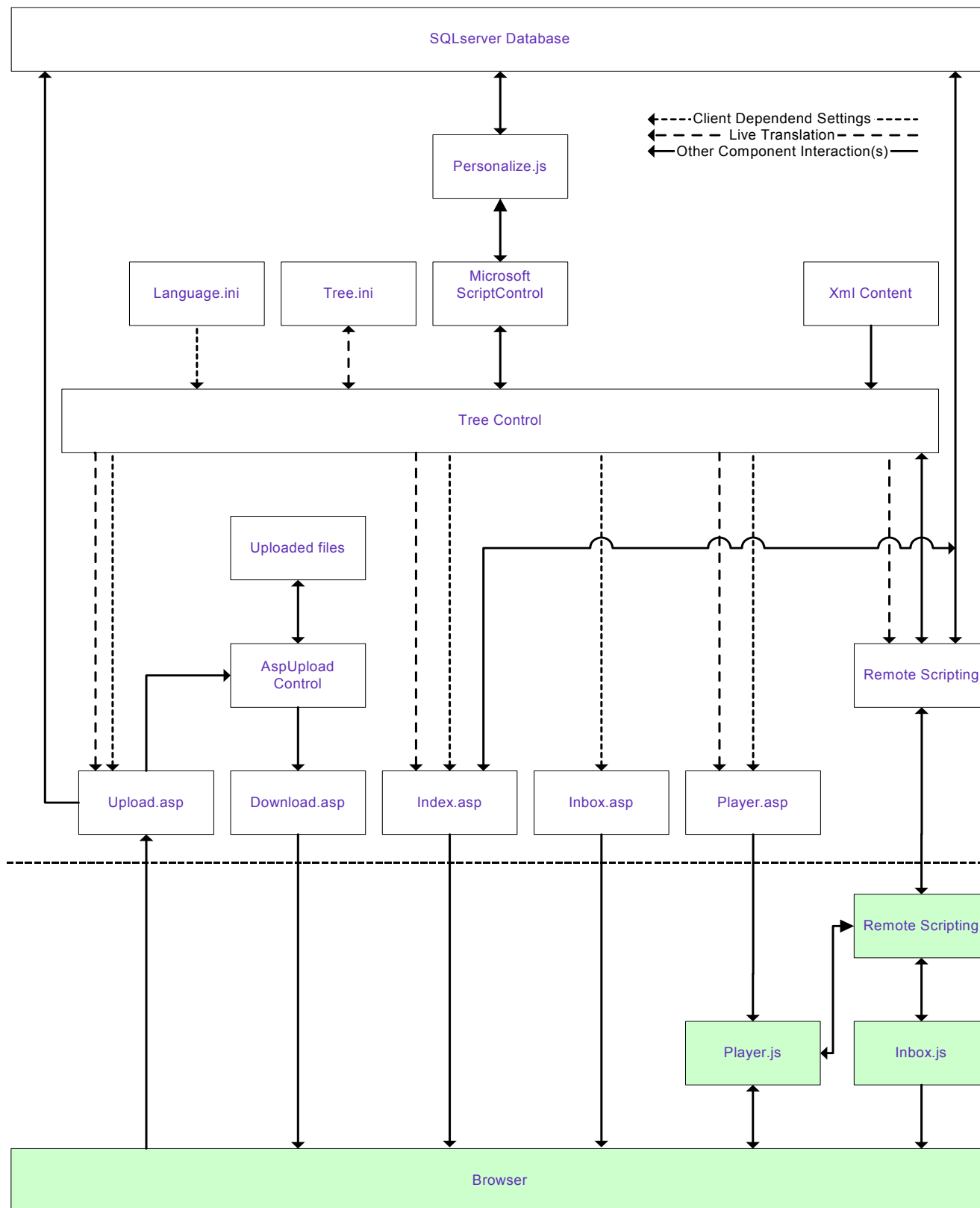
**Tabel 3. Delphi 5.0 sources van perftools.dll.**

Naast tree.dll is er ook nog een Tools Object dat gebruikt kan worden voor performance metingen op microseconden basis.

Naam	Bevat
Index.asp	
Inbox.asp	
Remote.asp	
Inbox.js	
Upload.asp	
Download.asp	
Player.asp	Niet gepersonaliseerde Player.js

**Tabel 4. Web Files.**

## Bijlage 2. Architectuur player.



Figuur 1. Samenhang Componenten Edubox 2.0 Player.

### Bijlage 3. Statische publicatie

In het 'Omnimark stylesheet' worden enkele EML elementen vertaald in HTML elementen en wordt de waarde van het class attribuut gezet volgens onderstaande tabel.

CSS stijl, class=	HTML element	EML element
	 	Chapter special-new-flow Metadata Copyright-statement Structured-source Activity What How With-whom When Why Hint Information Questionnaire-object Knowledge-object Index-search-object index Role-information-object Communication-object Announcement-object Tool-object Environment
	i	Volume
	img	Figure-source Formula-source Metadata-link Objectives-link Prerequisite-link
	ol	List (Ordered=Yes)
	ul	List (Ordered=No)
	li	Li
	<p />	P Only-for-learner
	pre	Code-line
border="0"	table	Unit-of-study Figure Formula Copyright-set Only-for-learner Support-role
	tr, td	Unit-of-study Figure Formula

<b>CSS stijl, class=</b>	<b>HTML element</b>	<b>EML element</b>
		Copyright-item Only-for-learner Support-role
	td	role-ref-entry
activity	div	Activity
activity-title	h1	Activity
eml-ref	a	EML-ref
Emph-1	span	Emphasis
Emph-2	span	Emphasis
Emph-3	span	Emphasis
Emph-4	span	Emphasis
Emph-5	span	Emphasis
generalfeedback	div	General-feedback (binnen Activity-description)
Heading	div	Creator Description Keywords Copyright Studyload Contributor-type Contributor Status Creation-date Date-last-change Min-completion-time Max-completion-time Meta Term
Heading	p	Extra-meta
Heading	span	Introduction What How Who When Why Hint (Activity-description) General-feedback (Activity-description) Information Objective-description Prerequisite-description Other-objective
Hintact	div	Hint (binnen Activity-description)
How	div	How
Information	div	Information
internet-ref	a	Internet-ref Internet-source External-questionnaire External-program
Introduction	div	Introduction
Lemma	div	Lemma

<b>CSS stijl, class=</b>	<b>HTML element</b>	<b>EML element</b>
Metadata	div	Metadata
metadata-item	div	alle metadata elementen
metadata-title	h1	Title in Metadata
Subtitle	div	Subtitle
Term	div	Term
Title	div	for Title in: Section Lemma Multiple-choice-question True-false-question Multiple-response-question Question-answer Subactivity Learning-objective Prerequisite Questionnaire-object Knowledge-object Index-search-object Role-information-object Communication-object Announcement-object Tool-object Environment
unit-of-study-title	h1	Unit-of-study
What	div	What
When	div	When
Who	div	Who
Why	div	Why



## **Bijlage 4. Elementen die dynamisch worden vormgegeven**

CSS eigenschappen worden dynamisch toegekend voor de EML-elementen:

- Special/Special-inline
  - classes voor collapse/expand
  - classes voor het attribuut Position
- de diverse vraagtypes inclusief de afhandeling daarvan
- het zetten en tonen van properties
- index/zoeken
- Send-message/Receive-message
- Announcement-object en afhandeling daarvan.

Deze elementen worden vertaald naar diverse HTML elementen (DIV, SPAN, INPUT, FORM, SELECT, OPTION, TEXTAREA, TABLE, TR, TD) waarvan het class attribuut gezet of gewijzigd wordt, of waarvan een CSS eigenschap dynamisch gezet of gewijzigd wordt.