

7 Editor

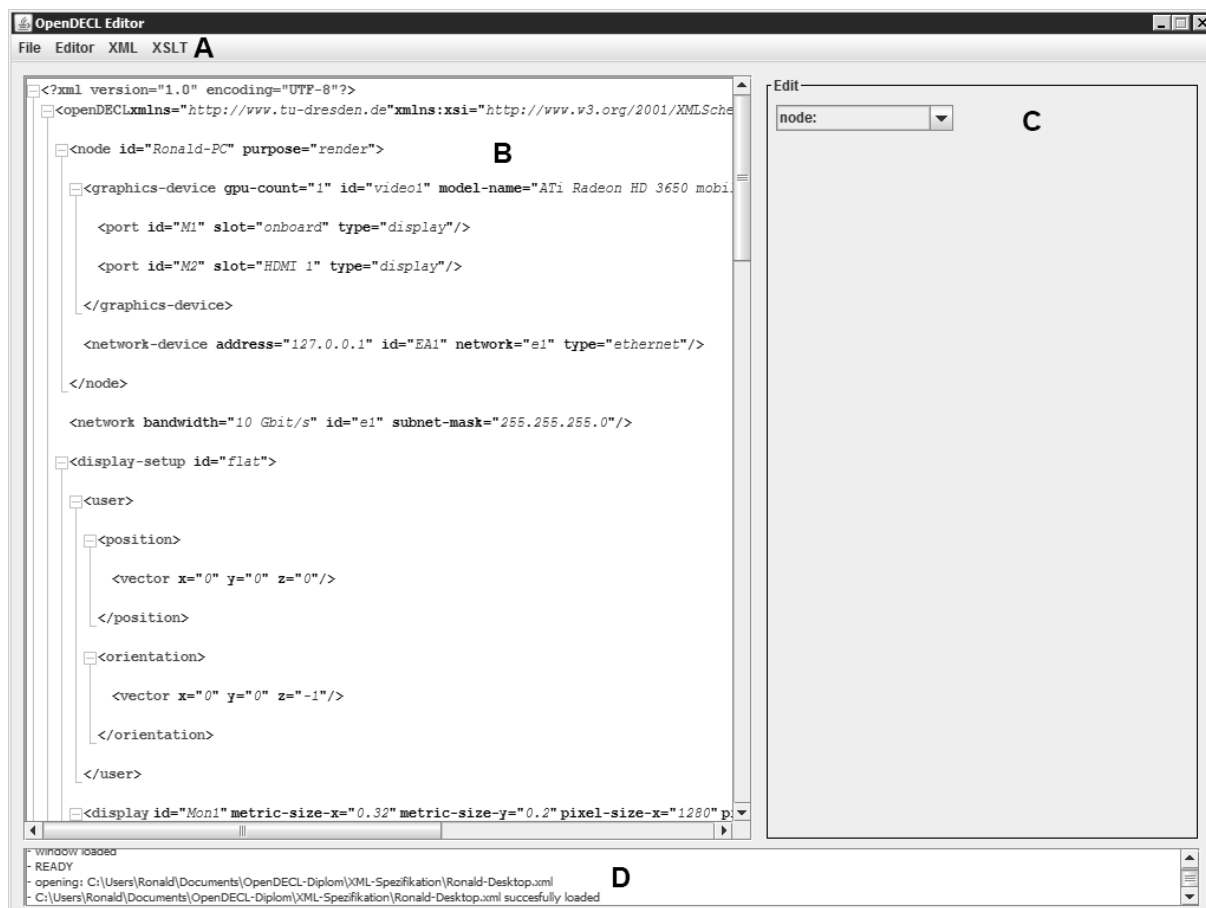


Abbildung 7.1: Editor zum Bearbeiten von OpenDECL-Dokumenten A, Menüleiste; B, Ansicht des aktuellen Dokumentes; C, Interface zum Bearbeiten des Dokumentes; D, Statuszeile

Bei dem Editor handelt es sich um einen Prototypen zum Editieren von OpenDECL-Dokumenten (Abb. 7.1). Es werden Funktionen zum editieren aller in der Spezifikation vorgestellten Elemente und Attribute zur Verfügung gestellt. Außerdem kann das Dokument auf Wohlgeformtheit getestet und anhand einer XML-Schema Datei validiert werden. Für den Export von Konfigurationen steht die Funktion zum anwenden einer XSLT-Datei zur Verfügung. Während der Bearbeitung kann das Dokument in einen Status kommen, in der es nicht mehr valide bezüglich des OpenDECL-Schema ist. Daher sollte vor dem Speichern immer eine Validierung durchgeführt werden.

7.1 Funktionen

Um ein OpenDECL-Dokument zu bearbeiten, muss zunächst ein gespeichertes Dokument geladen werden oder ein neues angelegt werden, wobei dabei ein Standarddokument mit minimalem Inhalt geladen wird. Beides kann über File in der Menüleiste gemacht werden (Abb. 7.1 A). Das Dokument erscheint in der Ansicht links (Abb. 7.1 B) wobei die XML-Syntax hervorgehoben wird und die einzelnen Elemente

verkleinert oder expandiert werden können.

Zum Bearbeiten wählt man rechts eines der drei Hauptelemente (*node*, *network* oder *display-setup*) aus dem Dropdown-Menü (Abb. 7.1 C) aus. Dadurch verändert sich entsprechend der Auswahl das Interface zum Bearbeiten. Es erscheinen Optionen zum Bearbeiten der entsprechenden Unterelemente und Attribute. Für Elemente gibt es Dropdown-Menüs, in denen diese mit ihren IDs aufgeführt. Zusätzlich gibt es mit **add new** immer die Option ein neues Element anzulegen beziehungsweise mit dem *x* hinter dem Dropdown-Menü das ausgewählte Element zu löschen. Unter den Dropdown-Menüs befinden sich Textfelder zum Editieren der Attribute. Bei bereits vorhandenen Elementen werden diese mit den Werten der Attribute vorbelegt. Pflichtattribute sind mit einem ***-Zeichen gekennzeichnet. Unter den Textfeldern für Attribute gibt es einen Set-Button. Mit einem Klick auf diesen Set-Button, werden die editierten Attribute des Elementes im aktuellen Dokument verändert beziehungsweise ein neues Element mit den angegebenen Attributen im aktuellen Dokument angelegt. Dabei wird kontrolliert ob alle Pflichtattribute gesetzt sind und im Falle von IDs ob diese eindeutig sind.

Möchte man den Text des aktuellen Dokumentes direkt bearbeiten, kann über Editor in der Menüleiste die Dokumentenansicht umgeschaltet werden. Dabei kann zwischen XML-Highlight-Ansicht (Standardansicht) und Text-Editor Ansicht umgeschaltet werden. In der Text-Editor-Ansicht lässt sich der Text des Dokumentes direkt bearbeiten und so beispielsweise größere Teile kopieren oder Elemente eines anderen Namespaces einfügen. In der Text-Editor-Ansicht stehen die zuvor beschriebenen Funktionen zum bearbeiten des Dokumentes nicht zur Verfügung. Wechselt man zurück in die XML-Highlight-Ansicht, wird zunächst überprüft, ob der editierte Text einem wohlgeformten XML-Dokument entspricht. Ist dies nicht der Fall, wird ein Fehler mit entsprechenden Informationen ausgegeben, und es wird nicht in die XML-Highlight-Ansicht gewechselt. Sollte man nicht in der Lage sein, das Dokument so zu bearbeiten, dass es Wohlgeformt ist, kann man im Menü File über Load Last Documentstate, die letzte zwischengespeicherte Version des aktuellen Dokumentes laden und kommt zurück in die XML-Highlight-Ansicht. Über den Menüpunkt XML kann man testen, ob das aktuelle Dokument wohlgeformt ist und ob es valide bezüglich eines XML-Schemas ist. Beim Validieren muss rechts eine XML-Schema Datei geladen werden wogegen das aktuelle Dokument validiert werden soll. Ähnlich verhält es sich mit dem Export von Konfigurationen via XSLT. Diese Funktion erreicht man im Menüpunkt XSLT. Dabei muss man rechts zunächst die XSLT-Datei laden, die für den Export verwendet werden soll und darunter die Zieldatei, in die das Ergebnis der Transformation gespeichert werden soll.

Bei allen Funktionen die das Dokument verändern sowie bei den XML Tests und XSLT Funktionen, wird in der Statuszeile unten (Abb. 7.1 D) ausgegeben, ob die Funktion erfolgreich durchgeführt wurde oder ob es einen Fehler gab. Im Falle eines Fehlers werden dort zusätzlich Informationen zum Fehler angegeben.

7.2 Umsetzung

Der Editor wurde in JAVA umgesetzt und liegt als ausführbare JAR-Datei vor. Die Oberfläche wurde dabei mit Elementen des Swing Paketes von JAVA umgesetzt. Der Kern des Editors ist die *OpenDECLdoc* Klasse. Diese repräsentiert eine OpenDECL Dokument und stellt nach außen hin Funktionen zum Bearbeiten zur Verfügung. Innerhalb der Klasse wird das XML-Dokument durch ein DOM-Document-Objekt repräsentiert. Dieses wird beim erstellen einer Instanz von *OpenDECLdoc* erstellt, wobei eine XML-Datei geladen wird, deren Pfad dem Konstruktor mit übergeben wird. Zum Bearbeiten dieses XML-Dokumentes gibt es nichtöffentliche Funktionen, die beispielsweise ein neues Element einfügen oder Attribute bearbeiten. Nach außen hin werden Funktionen zur Verfügung gestellt, die speziell für OpenDECL Elemente und Attribute konzipiert sind, beispielsweise *addGraphicsDevice* zum hinzufügen eines neuen *graphics-device* Elementes oder *setDisplayPortref* um das *portref* Attribut eines *display* Elementes zu setzen (weitere Funktionen Abb. 7.2). Diese öffentlichen Funktionen benutzen die nicht-öffentlichen Funktionen um das Dokument entsprechend zu bearbeiten beziehungsweise Informationen des Dokumentes zurückzugeben.

Im Editor wird eine Instanz der *OpenDECLdoc* Klasse gehalten. Die verschiedenen Interface Elemente rufen die entsprechenden öffentlichen Funktionen zum Bearbeiten des Dokumentes auf. Dabei wird nach jeder Bearbeitung eine temporäre Kopie des aktuellen Dokumentes unter dem Namen *temp.output.xml* lokal gespeichert. Die Funktionen zum Validieren beziehungsweise Testen auf Wohlgeformtheit und für den XSLT-Export, sind nicht Teil der *OpenDECLdoc* Klasse, sondern sind in den entsprechenden Interface Elementen umgesetzt.

Der Quellcode für den Editor befindet sich auf dem beigelegten Datenträger.

Abbildung 7.2: Klassendiagramm der *OpenDECLdoc* Klasse mit allen öffentlichen Funktionen