

XML-basierte Graphikstandards für interaktive Systeme: Erleichterter Zugang zu graphischer Information

Martin Rotard
Institut für Visualisierung und Interaktive Systeme
Universität Stuttgart
70569 Stuttgart
Tel.: ++49-(0)711-7816-269
Fax: ++49-(0)711-7816-340
E-Mail: rotard@vis.uni-stuttgart.de

Abstract: Graphiken werden in vielen Bereichen, von der Visualisierung bis zur Darstellung von Benutzungsoberflächen eingesetzt. Die XML-basierten Standards bzw. Empfehlungen Scalable Vector Graphics, Extensible 3D und Mathematical Markup Language haben viele Vorzüge gegenüber herkömmlichen Bitmapgraphiken. Dieser Artikel stellt Einsatzbereiche und technischen Möglichkeiten bei der Verwendung dieser neuer Graphikstandards vor und demonstriert Verfahren, die blinden Menschen den Zugang zu diesen Graphiken erlauben.

Stichworte: XML-basierte Graphikstandards, SVG, X3D, MathML, Accessibility

1 Motivation

Graphiken spielen in Benutzungsschnittstellen eine zentrale Rolle. Durch Graphiken wird den Anwendern von Benutzungsoberflächen auf direkte und einfache Weise ermöglicht, mit einem Computersystem zu interagieren. Darüber hinaus können Graphiken textuelle Inhalte ergänzen, einen Überblick über komplexe Sachverhalte bieten und dazu die Aufmerksamkeit des Betrachters wecken. Neben den Fenstersystemen der Betriebssysteme hat sich im World Wide Web HTML als eigenständige Beschreibungssprache für graphische Benutzungsoberflächen etabliert, die mit Browsern dargestellt wird. Die ersten Dokumente des Webs waren rein textbasiert, aber schon bald darauf wurde die Einbettung von Rastergraphiken ermöglicht. Um weitere Interaktionsmöglichkeiten zu bieten, wurden die Browser von Ihren Herstellern um immer mehr Funktionalität und PlugIn-Schnittstellen ergänzt. Dadurch begann eine Phase, in der Standards nicht von einem Konsortium verabschiedet und dann umgesetzt wurden, sondern die einen Wildwuchs von proprietären Quasistandards hervorbrachte. Die hierdurch entstandenen Kompatibilitätsprobleme erhöhen den Aufwand und die Kosten für die Erstellung von einheitlichen Benutzungsoberflächen im Web, da diese speziell für jeden Browser und dazu noch abhängig von dessen Version entwickelt werden müssen. Durch diesen Umstand wurden das World Wide Web Consortium (W3C) und das Web3D Consortium aktiv, deren Ziel es ist einheitliche Standards zu entwickeln.

Die aktuellen Standards bzw. Empfehlungen, dieser Konsortien im Bereich Graphik bieten neue Möglichkeiten zur Entwicklung flexibler Benutzungsoberflächen, so dass diese nicht nur auf Browseranwendungen begrenzt sind, sondern auch allgemein im Bereich Fenstersysteme Anwendung finden können. Ziele dieser Arbeit sind die technischen Möglichkeiten bei der Verwendung dieser neuer Standards in der Graphik aufzuzeigen und Verfahren für die Zugänglichkeit für blinde Menschen zu entwickeln. In die Betrachtung der Arbeit werden ausschließlich auf XML basierende Standards einbezogen, da diese einheitlich über definierte Schnittstellen erzeugt und manipuliert werden können.

2 XML-basierte Standards im Bereich Graphik

Die Standards bzw. Empfehlungen, die für diese Arbeit betrachtet wurden, sind Scalable Vector Graphics (SVG), Extensible 3D (X3D) und Mathematical Markup Language (MathML). Diese werden im Folgenden kurz vorgestellt, eine ausführlichere Darstellung ist in [RoScEr2002] zu finden. Anschließend werden der Aufbau und die Eigenschaften von XML-basierten Graphiken verdeutlicht und an einem Beispiel illustriert.

Scalable Vector Graphics (SVG): SVG ist eine Empfehlung des W3C für die Beschreibung interaktiver zweidimensionaler Vektorgraphiken in XML. Einfache geometrische Formen wie Rechtecke, Ellipsen, Linien, Polygone, Pfade (Kurven) etc. werden dabei Attribute wie Füllfarbe, Deckkraft, Strichstärke, etc. zugewiesen. Zusätzlich können Filter-Effekte wie Farb-, Überblend-, Wisch-, Misch- und Beleuchtungseffekte angewendet werden, wie sie in gängigen Bildbearbeitungsprogrammen eingesetzt werden. SVG verfügt über eine Scripting- und Animationschnittstelle, die über DOM (Document Object Model) angesteuert werden kann.

Extensible 3D (X3D): Für die Beschreibung interaktiver dreidimensionaler Vektorgraphikszene hat das Web3d Consortium mit Unterstützung des W3C diesen Standard als Nachfolger der Virtual Reality Modeling Language (VRML) etabliert. In X3D können dreidimensionale Geometrie und deren Ausleuchtungen, Animationen und Interaktionsmöglichkeiten beschrieben werden. Die Ansteuerung der Szene ist über das Scene Access Interface (SAI) oder über DOM möglich.

Mathematical Markup Language (MathML): MathML ist eine Empfehlung des W3C für die Beschreibung von mathematischen Inhalten. Diese wurden bisher fast ausschließlich als Rastergraphik dargestellt. Mit MathML kann sowohl die Darstellung eines mathematischen Ausdrucks, als auch dessen Semantik beschrieben werden. Diese beiden Ebenen werden durch die MathML Presentation Markup und die MathML Content Markup repräsentiert. Die Ausdrücke können über die DOM-Schnittstelle manipuliert werden.

XML-basierte Graphiken besitzen einen Wurzelknoten, der als Kindknoten jeweils Graphikprimitive, Gruppen, Definitionen, etc. enthält. Oft ist das Hinzufügen von Metadaten wie Titel, Be-

phen, Produktvisualisierungen, etc. genutzt werden. Andererseits können sie in graphischen Benutzungsoberflächen für die Darstellung von Widgets, Icons, Menüs, etc. verwendet werden [BuRoWa2004]. Auf die Einsatzbereiche E-Learning und Benutzungsoberflächen wird in den beiden folgenden Abschnitten vertieft eingegangen. An diesen beiden Bereichen können anschaulich die Vorteile XML-basierter Graphik dargestellt werden.

3.1 XML-basierte Graphik in Lehrmaterialien

Lehrmaterialien werden heutzutage oft in proprietären Dokumentenformaten gespeichert und dann als HTML- oder PDF-Dokumente im Web publiziert. Neuere Verfahren setzen auf XML-basierte Beschreibungssprachen (EML, Targeteam, CDF, LMML, etc.), um Lehrinhalte auszuzeichnen. Ein ausführlicher Vergleich ist in [RVKRL2002] zu finden. Der Vorteil liegt dabei in der Trennung von Lehrinhalt, dessen Layout und der Struktur. Damit können Lehrmaterialien in Lernplattformen speziell den Bedürfnissen und Vorlieben der Lernenden und dem Kontext angepasst werden. Der Grad der Flexibilität bei der Generierung der Ausgabe kann noch durch die Einbeziehung XML-basierter Graphiken gesteigert werden. Dadurch lässt sich die Graphik speziell an den Kontext und dessen Formatierung anpassen. Neben einer Anpassung des Farbschemas und der Zeichensätze in der Graphik lassen sich der Textinhalt, die Graphikobjekte und die Betrachtungsparameter verändern. Darüber hinaus ist eine Transformation der Graphikinhalte in eine für blinde Menschen erfassbare Form möglich (vgl. Abschnitt 4).

In vielen Bereichen der Wissenschaft verändern sich die Forschungsinhalte rapide. Dabei müssen auch die Lehrmaterialien in diesen Bereichen an die Neuerungen angepasst werden. Für die Autoren der Materialien sind daher die Aspekte Wiederverwendung von existierenden Materialien und nachhaltige Erstellung besonders wichtig. Im Projekt Information Technology Online (ITO) wurde ein ITO-Dokumentenformat und eine Arbeitsumgebung entwickelt. Durch die Möglichkeit der Transformation in das Ausgabeformat lässt sich dieses besonders flexibel einsetzen [ITO, RoFi2003]. Einige Projekte aus dem Bereich E-Learning haben in den letzten Jahren auch ihr eigenes Dokumentenformat definiert. Bei vielen war für die Erstellung eines Lehrmoduls die native Bearbeitung des XML-Quelltexts notwendig. Dies wurde von den Autoren der Inhalte als sehr umständlich und nicht intuitiv aufgefasst. Außerdem ist die Integration existierender Materialien schwierig. Das ITO-Dokumentenformat besteht aus modularen Komponenten eines XML-Schemas, mit denen sich Lehrmodule auszeichnen lassen [FiRo2002]. Dabei ist es möglich, neben herkömmlichen Graphikformaten auch XML-basierte Graphiken einzubinden. Um eine plattformunabhängige Erstellung von Lehrmodulen zu ermöglichen, wurde die freie Software OpenOffice.org verwendet. In diesem Rahmen wurde eine Arbeitsumgebung geschaffen, die den Autoren in vielen Aspekten wie Erstellung, Gliederung, Validierung, Metadatenpflege, Importierung existierender Materialien und Exportierung in das ITO-Dokumentenformat Unterstützung bietet. Die umfangreichen Import- und Exportfilter und das XML-basierte ITO-Dokumentenformat ermöglichen eine Konvertierung existierender Materialien vieler Formate. Auch in anderen Projekten haben Arbeiten in diesem Bereich stattgefunden, wie dem FIGARO Projekt, das auch OpenOffice.org als Autorenwerkzeug verwendet [Oelt2003], und der entwi-

ckelten Sprache XDiML (Dissertation Markup Language) an der Humboldt Universität in Berlin [HeScVo2003].

Wenn man ein konventionelles Autorenwerkzeug wie OpenOffice.org für die Erstellung von Lehrmaterialien einsetzt, kann man Absätze und Zeichenketten mit Formatvorlagen auszeichnen und anschließend zu Knoten der XML-basierten Beschreibungssprache zuordnen [FiRo2003]. Für einige Anwendungsfälle, wie das Exportieren eines Lehrmoduls in das ITO-Dokumentenformat und das Editieren der Metadaten in Standard LOM (Learning Object Metadata), wurden Assistenten entwickelt. Beim Export eines Lehrmoduls besteht die Möglichkeit, die Ausgabe der Zielgruppe anzupassen. Um Versionen ohne XML-basierte Graphiken zu erzeugen, können dabei die integrierten XML-basierten Graphiken automatisch durch Bitmapgraphiken ersetzt werden. Das Ausgabeformat kann dabei HTML oder ein anderes selbst definiertes Format sein. Der Vorteil bei der Verwendung von XML-basierten Graphiken in Lehrmaterialien ist darüber hinaus die Wiederverwendbarkeit der Graphiken durch Co-Dozenten und durch Lernende. Auch Übungsaufgaben, in denen abgebildete Diagramme, Schaubilder, allgemeine Formeln, graphische Szenen, etc. modifiziert oder ergänzt werden sollen, sind damit möglich. Der hier beschriebene Ansatz wird im Projekt metacoon-services eingesetzt und ist direkt in die Lernplattform metacoon integriert [Metacoon]. Der Ansatz ist auch nicht auf das Autorenwerkzeug OpenOffice.org beschränkt, sondern lässt sich auf viele verfügbare Systeme erweitern [RoFi2004].

3.2 XML-basierte Graphik in Benutzungsoberflächen

Graphische Benutzungsoberflächen sind die am häufigsten verwendete Schnittstelle zwischen Mensch und Computer. XML-basierte Graphiken werden seit kurzem in immer mehr graphische Benutzungsoberflächen integriert. In den Projekten KDE und GNOME werden bereits Icons mit SVG beschrieben und gerendert. Weitere Bestrebungen sind die gesamte Benutzungsoberfläche mit XML und die Widgets mit XML-basierten Graphiken zu beschreiben, Beispiele hierfür sind die User Interface Markup Language und das Custom Graphics User Interface [UIML, CGUI]. Auch die Benutzungsoberfläche der nächsten Generation des Windows-Betriebssystems (Code-name Longhorn), die Avalon genannt wird, setzt auf XML-basierte Graphik, die ähnlich der Beschreibung von SVG ist. Der Hauptvorteil von XML-basierten Graphiken in Benutzungsoberflächen liegt in der Skalierbarkeit der Widgets. Dabei werden nicht nur die Widgets in der Größe angepasst, sondern auch die Darstellung der Inhalte. Ein weiterer Vorteil ist die einfache Weiterverarbeitung der XML-basierten Beschreibung. Speziell dieser Aspekt wurde in einem Projekt an der Universität Stuttgart erforscht und soll im Folgenden beschrieben werden.

Im Client/Server-Bereich gibt es eine Vielzahl von Ansätzen/Protokollen, die es ermöglichen die graphische Benutzungsoberfläche und Anwendung getrennt zu betreiben und zwischen beiden Systemen die notwendigen Informationen über das Netzwerk auszutauschen. Vertreter dieser Klasse sind X11, Virtual Network Computing (VNC), Microsofts Remote Desktop Protocol (RDP), etc. Dabei werden vom Client zum Server die Events des Benutzers und in die Gegen-

richtung die Steuerbefehle bzw. ganze Graphiken der aktuellen Arbeitsoberfläche übertragen. Dazu werden vorwiegend spezielle Ports verwendet, die jedoch häufig in den Firewalls wegen Sicherheitsaspekten geschlossen werden. Die Kommunikation zwischen Client und Server sollte daher über HTTP und Port 80 erfolgen, dies ist aber bei vielen Protokollen nicht möglich. Anforderungen dieser Art finden Anwendung im Rapid-Prototyping, wie beispielsweise in Client/Server-Systemen zur Simulation von Geräteprototypen wie Handys, Kameras, medizinischen Geräten, etc. Bei einem solchen Client/Server-System visualisiert die Client-Anwendung die Bedienoberfläche des Geräts und die Server-Anwendung simuliert dessen Funktionsweise. Das Spektrum der hier betrachteten simulierten Geräte ist damit nicht auf die Rechenleistung des Client-Computers beschränkt. Auf der Seite des Servers kann gegebenenfalls auch ein Cluster die Berechnung der Graphiken übernehmen. Damit können beispielsweise bei medizinischen Geräten realistisch wirkende Visualisierungen berechnet werden.

Mit dem Protokoll SOAP (Simple Object Access Protocol) kann die Kommunikation zwischen Client und Server über HTTP realisiert werden. Die Schnittstelle des Servers kann dabei als Webservice verwendet werden. Bei manchen Geräten ist eine dreidimensionale Visualisierung sinnvoll, da dabei ein möglichst realistischer Eindruck von der Bedienung entsteht. Als Beschreibungssprache für die Geometrie der 2D-Modelle bietet sich SVG und für die der 3D-Modelle X3D an. Damit ist es möglich, die veränderten Geometriedaten strukturiert über das Netzwerk zu übertragen und beim Client in die Visualisierung zu integrieren. Durch die einheitliche Schnittstelle kann über DOM eine von den Dimensionen der Visualisierung des Geräts unabhängige Handhabung erfolgen. Der Server und der Client arbeiten beide mit einer Instanz der graphischen Bedienoberfläche des Modells, die synchron gehalten werden müssen. Als Reaktion auf die vom Client an den Server gemeldeten aufgetretenen Maus- und Tastatur-Events, sendet der Server die Änderungen in der Visualisierung der Bedienoberfläche zurück. Diese Nachricht besteht aus einer Liste der zu modifizierenden Knoten und der Information über die Veränderungen der Attribute und der neuen Kindknoten. Der Client decodiert die Nachricht und modifiziert das Modell der Visualisierung entsprechend einheitlich über die Schnittstelle DOM. Da es nicht sinnvoll ist, jedes aufgetretene Event vom Client an den Server zu schicken, kann der Server für jeden Knoten festlegen, welche Events er bekommen möchte. Da SVG und X3D auf XML basieren, kann dazu jedem Knoten ein Kindknoten eines speziell dafür vorgesehenen Namensraumes hinzugefügt werden. Dieser wird vom Server eingefügt und vom Client interpretiert. Mit den Attributen dieses spezifischen Eventknotens können die spezifischen Eigenschaften für die Handhabung des Clients für die verschiedenen Events konfiguriert werden. Ein Beispiel für die Notwendigkeit des selektiven Übertragens von Events ist ein Schieberegler. Erst bei der gewählten Endposition ist eine Neuberechnung des Modells notwendig. Ein wichtiger Aspekt, um die Kommunikation von Client und Server zu reduzieren, ist die selektive Verarbeitung von Events auf dem Client zu ermöglichen. Beispielsweise benötigt es keiner komplexeren Verarbeitung, wenn neben einem Eingabefeld für numerische Werte der Button für die Erhöhung des Wertes gedrückt wird, sondern kann mit einem einfachen Skript auf dem Client geschehen. Erst wenn der Benutzer zum Beispiel den Button für die Bestätigung der Eingabe drückt, ist eine Übertragung der entsprechenden Werte der Eingabefelder notwendig. Um dies zu ermöglichen,

muss der Client die Information besitzen, welche Werte er an den Server übertragen muss. Diese Konfiguration kann im oben beschriebenen spezifischen Eventknoten, der dem Button für die Bestätigung zugeordnet ist, vorgenommen werden.

4 Zugang zu XML-basierten Graphiken

Viele Informationen wie Lehrmaterialien, Bücher, Verzeichnisse, etc. liegen heute bereits in digitaler Form vor. Dadurch ist die Aufbereitung dieser Materialien für blinde Menschen möglich. Text lässt sich für diese Nutzergruppe über die Sprachausgabe oder die Braillezeile ausgeben. Graphiken liegen meist als Bitmapgraphik vor, daher werden für die Aufbereitung Optical Character Recognition (OCR) Verfahren verwendet, um Text zu extrahieren. Die Erkennung von geometrischen Formen mit diesen Verfahren ist ein nicht triviales Problem, welche ungenaue Ergebnisse liefern kann [Schw1996]. Auch mathematische Formeln werden meist als Bitmapgraphik dargestellt. Dadurch wird die Vermittlung von Lehrinhalten im schulischen und wissenschaftlichen Bereich für blinde Menschen erschwert. In den folgenden Abschnitten wird auf die Aufbereitung von XML-basierten Graphiken in Form von Formeln, 2D-Graphiken und 3D-Graphiken für blinde Menschen eingegangen. Dies ist nur deshalb möglich, da sich die XML-basierte Beschreibung einfach weiterverarbeiten lässt.

4.1 Mathematische Formeln

Mathematische Formeln besitzen eine starke Aussagekraft und sind die kompakteste Form Wissen darzustellen. Auf die Beschreibung der mathematischen Aussage in Worten wird deshalb meistens verzichtet. Von blinden Lernenden kann der beschreibende Text, aber nicht die für das Verständnis wichtigen Aussagen der Formeln erfasst werden, wenn diese als Bitmapgraphik eingebettet sind. Werden die Formeln in MathML beschrieben, dann ist die textuelle Beschreibung für Blinde zugänglich [RoScEr2004]. Jedoch ist es schwierig, die Aussage aus der umfangreichen XML-Beschreibung zu interpretieren. Deshalb bietet sich eine Transformation der Formelbeschreibung in MathML in eine spezielle mathematische Notation für Blinde an. Vorarbeiten und aktuelle Forschungen in diesem Bereich sind in [Gard2002, KGMPG2001, Lambda] zu finden. Im Folgenden wird die Transformation in SMFB (Stuttgart Mathematical Notation For the Blind) beschrieben, die an der Universität Stuttgart seit 1980 entwickelt und weiterentwickelt wird [Schw1980]. SMFB basiert auf der Marburger Mathematikschrift, die 6 Punkte zur Darstellung von mathematischen Formeln verwendet [ScWiZe1930]. SMFB dagegen ist eine 8-Punkt-Schrift, was die darstellbaren Zeichen von 64 auf 256 erhöht. Dies ermöglicht eine einfachere Handhabung, da die in Formeln verwendeten Zeichen wie beispielsweise Buchstaben aus dem griechischen Alphabet direkter dargestellt werden können. In Abb. 2 ist eine SMFB-Formel in Brailleschrift dargestellt. Für SMFB gibt es dazu eine Repräsentation für Sehende, in der jedes Zeichen direkt auf ein Sonderzeichen abgebildet wird.

The image shows three representations of the mathematical formula $\sqrt{\frac{x}{y}}$. From left to right: 1. A Braille representation consisting of seven vertical columns of dots. 2. A LaTeX representation of the formula $\sqrt{\frac{x}{y}}$ in a serif font. 3. The LaTeX source code `$\sqrt{\frac{x}{y}}$` in a monospace font.

Abbildung 2: Mathematische Formel in Brailleschrift, in deren Repräsentation für Sehende und in LaTeX

Die Transformation einer mit MathML beschriebenen Formel in SMFB basiert auf XSTL-Stylesheets. Für die Darstellung von SMFB werden Zeichen des Unicode-Zeichensatzes verwendet. In der Transformation wird jedes MathML-Element durch ein entsprechendes SMFB-Zeichen ersetzt, das anschließend mit XHTML positioniert und formatiert wird. Diese Ersetzungsregeln sind für jedes MathML-Element einzeln in einem XSLT-Template beschrieben [RBSE2003]. Durch die Verwendung von XSLT ist es möglich, die Transformation in eine virtuelle Lernumgebung auf dem Server einzubinden. Neben der Transformation der Formeldarstellung kann auch das Layout der Lehrmodule angepasst werden. Eine blindengerechte Darstellung von Lehrmodulen kann dann durch die Ausführung des XSLT-Stylesheets auf dem Server generiert werden, ohne Installation von Software auf den Rechnern der Benutzer. Der Benutzer selbst kann in seinem Profil der Lernumgebung wählen, auf welche Weise die Formeln dargestellt werden [RoFi2003]. Die Auswahl kann um eine Vielzahl von Formeldarstellungen erweitert werden, indem weitere XSLT-Stylesheets hinzugefügt werden. Damit kann jedem Benutzer individuell der Zugang zu Formeln ermöglicht werden. Als Demonstration dieser Fähigkeiten wurde in unsere prototypisch entwickelte Lernplattform ein weiteres XSLT-Stylesheet eingebunden, das Formeln, die in MathML beschrieben sind, in LaTeX-Formeln transformiert [XSLTML] (vgl. Abb. 2).

4.2 2D-Graphiken

Graphiken sind für blinde Menschen nur schwer zugänglich. Mit taktilen Ausgabegeräten können speziell aufbereitete Graphiken für blinde Menschen dargestellt werden. Diese Ausgabegeräte besitzen meist eine sehr niedrige Auflösung (wie 16x16, 120x60) und können nur monochrome Graphiken anzeigen, da jeder Stift des Geräts nur zwei Zustände hat (erhaben/nicht erhaben). Deshalb müssen die Graphiken kleiner skaliert und in der Farbenzahl auf die Hinter- und Vordergrundfarbe reduziert werden. Die Aufbereitung fällt bei Bitmapgraphiken mit OCR-Verfahren schwer, da im Allgemeinen aus dem Dateiformat keine Metainformationen wie Titel, Beschreibung, textueller Inhalt, enthaltene geometrische Formen, Liniendicken, etc. gespeichert werden. Vektorgraphiken haben für diesen Anwendungsfall Vorzüge, da eine verlustfreie Skalierung möglich ist und deren Metainformationen zur gesamten Graphik bzw. zu den einzelnen Komponenten direkt ausgelesen und bei der Aufbereitung verwendet werden können. Graphiken, die in SVG beschrieben sind, lassen sich auf diese Weise auf taktilen Ausgabegeräten interaktiv darstellen. Drei Aspekte sind dabei wichtig: die Ausgabe der Metainformation auf der Braillezeile/Sprachausgabe, die Navigation der Graphik und das Anwenden von Filtern [RoOtEr2004]. Im Accessibility-Bereich gab es bereits Aktivitäten der Web Accessibility Initiative [Jack2003] des W3Cs. [McKo2002] stellt Richtlinien auf, wie Graphiken in SVG erzeugt werden müssen, um zugänglich für sensorisch Behinderte zu sein. Weitere Veröffentlichungen beschreiben das Aus-

lesen von Metainformationen [HeDa2002], die Darstellung von Karten [CBMS2003] und das Drucken von SVG mit Blindendruckern [GaBu2001].

Durch den direkten Zugriff auf die graphischen Elemente kann ein schrittweiser Aufbau realisiert werden (Abb. 3). Zum jeweils aktuellen Element wird der Typ, evtl. der textuelle Inhalt, die Farben, etc. ausgegeben. Zur Orientierung gibt es einen Überblicksmodus und eine Umrissdarstellung. Wenn es sich beim aktuellen Element um eine Gruppe handelt, kann diese betreten und auch wieder verlassen werden. Das Aufheben der Verdeckungsberechnung ermöglicht dem Benutzer Objekte als Ganzes zu ertasten, die von anderen Objekten verdeckt werden. Um die kleine Auflösung des taktilen Ausgabegeräts besser auszunutzen, ist ein zoomen und scrollen der Ausgabe möglich. Die wichtigste Information für blinde Menschen ist der Text in der Graphik. Dieser lässt sich mit einem Textfilter auslesen und ermöglicht das Navigieren zur nächsten Textstelle. Die taktile Ausgabe zeigt die Textposition durch Blinken an. Durch die Beschränkung auf die monochrome Darstellung, können Fülleffekte wie Gradienten und Muster zu Kanten in der aufbereiteten Graphik führen. Dies kann durch Anwendung eines speziellen Filters vermieden werden. Um alle Graphikelemente einer Farbgruppe separat darzustellen, kann ein Farbfilter verwendet werden. Dazu wurden als Standardwerte die elf unterscheidbaren Farbgruppen von Brent Berlin und Paul Kay verwendet [BeKa1969]. Diese lassen sich aber für Sehgeschädigte zur Falschfarbendarstellung anpassen. Im L^*a^*b Farbraum werden die Farben der Graphik einer oder mehreren Farben der “Basic Color Terms” zugeordnet. Die in SVG definierten Farbnamen werden verwendet, um die Farbinformationen auf der Braillezeile oder als Sprachausgabe zu jedem Graphikelement auszugeben [RoEr2004]. Erste Benutzertests haben ergeben, dass die Navigation der Graphik mit Hilfe der Metainformationen einfach handhabbar ist. Die aufbereitete Graphik auf dem taktilen Ausgabegerät ist von guter Qualität und das Anwenden von Filtern erscheint logisch und steuerbar.

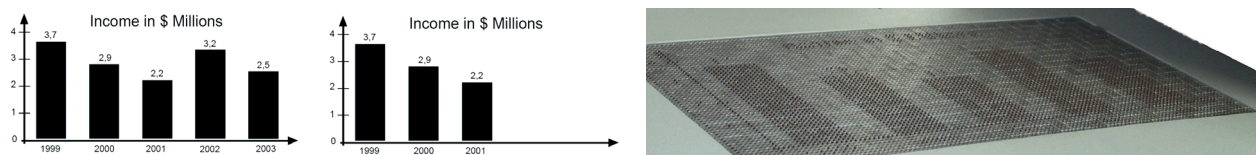


Abbildung 3: Original, schrittweiser Aufbau der Graphik und Ausgabe auf einer taktilen Matrix

4.3 3D-Graphiken

Blinde Menschen den Zugang zu 3D-Graphiken zu ermöglichen, ist eine große Herausforderung. Basierend auf dem in Abschnitt 4.2 vorgestellten System lässt sich dies auf einfache Weise umsetzen. [DiGöEr2004] beschreibt einen Ansatz, in dem 3D-Graphiken auf einem Server als 2D-Liniengraphik gerendert, anschließend zu einem Mobile Gerät übertragen und dort in Echtzeit angezeigt werden. Für die 2D-Liniengraphik wird ein spezielles Verfahren verwendet, dass die Silhouette, die Kamm- und Tallinien und die Flächengrenzen in Linienzüge umwandelt. Dabei gehen allerdings gefüllte Flächen und Schattierungen verloren. Die Events des Clients werden an den Server übertragen und entsprechend die neue Graphik gerendert. Basierend auf die-

sem Verfahren wurde das System aus Abschnitt 4.2 um die Möglichkeit erweitert, die Liniengraphiken, die vom Server berechnet werden, in XML-basierte Graphiken im SVG-Format umzuwandeln. Diese lassen sich dann mit dem herkömmlichen Verfahren als SVG auf einem taktilen Ausgabegerät für blinde Menschen anzeigen. Für die interaktive Steuerung ist ein spezieller 3D-Modus notwendig, der Rotationen um die drei Achsen und das Zoomen ermöglicht. Abb. 4 zeigt den Utah Teapot als 3D-Graphik gerendert, in der 2D-Liniendarstellung und als tastbare 2D-Graphik, die für blinde Menschen aufbereitet wurde. Weitere Forschungen in diesem Bereich sind in [KaTo1996, Kurz1997] zu finden.



Abbildung 4: 3D-Rendering, 2D-Liniendarstellung und tastbare Aufbereitung des Utah Teapots

5 Ergebnisse

Die hier vorgestellten Anwendungen und Verfahren hatten zum Ziel die technischen Möglichkeiten bei der Verwendung XML-basierter Graphiken aufzuzeigen. Dazu wurden die Einsatzbereiche von XML-basierten Graphiken in Lehrmaterialien und in Benutzungsoberflächen exemplarisch betrachtet. Zum einen wurde ein Prozess dargestellt, mit dem sich lernplattformunabhängige Lehrmaterialien mit XML-basierten Graphiken erstellen und anschließend weiterverarbeiten lassen, und zum anderen ein Client/Server-System, mit dem sich 2D- und 3D-Benutzungsoberflächen zur Gerätevisualisierung für das Rapid Prototyping darstellen lassen. Dabei können alle Arten von Graphiken wie Formeln, Diagramme, Landkarten, Bäume, Graphen, Produktvisualisierungen, etc. mit XML-basierten Graphiken modelliert werden. Als Nachteil von XML-basierten Graphiken muss die langsamere Verarbeitungsgeschwindigkeit und der bisher noch fehlende Schutz der Quelltextinformationen genannt werden. Der Einsatz von XML-basierter Graphik bietet sich für Bereiche besonders an, bei denen die Eigenschaften Skalierbarkeit, Wiederverwendbarkeit, Metainformationen, einheitliche Schnittstelle und die algorithmische Weiterverarbeitung oder Generierung eine wichtige Rolle spielen. Für den Zugang von blinden Menschen zu den Graphiken spielen die Eigenschaft der algorithmischen Weiterverarbeitung und das direkte Auslesen von Metainformationen eine ausschlaggebende Rolle. Diese beiden Eigenschaften sind besonders wichtig für die Umsetzung der vorgestellten Verfahren zur Transformation von mathematischen Formeln in eine speziell mathematische Mathematikschrift für Blinde und zur interaktiven taktilen Aufbearbeitung von 2D- und 3D-Graphiken.

6 Literatur

- [BeKa1969] Berlin, B.; Kay, P.: Basic Color Terms: Their Universality and Evolution, Berkeley and Los Angeles, University of California Press, 1969
- [BuRoWa2004] Burger, C.; Rotard, M.; Wacker, A.: Zukünftige Trends online-gestützten Lehrens und Lernens in informationstechnischen Berufen, In: Peter Göhner (Hrsg.), Information Technology Online, S. 247-254, Waxmann Verlag, 2004
- [CBMS2003] Campin, M. B.; Brunet, L. L.; McCurdy, W. A.; Siekierska, E. M.: SVG Mapping for People with Visual Impairment, SVG Open 2003
- [CGUI] Lewis, C.: CGUI - Custom Graphics User Interface
- [DiGöEr2004] Diepstraten, J.; Görke, M.; Ertl, T.: Remote Line Rendering for Mobile Devices, Proceedings of IEEE Computer Graphics International, 2004
- [FiRo2002] Finsterle, L.; Rotard, M.: Mit konventionellen Autorensystemen zum E-Learning Portal, Tagungsband der 10. Leipziger Informatik Tage, 2002
- [FiRo2003] Finsterle, L.; Rotard, M.: Automated Format Transformation for Courseware, OpenOffice.org, Conference 2003
- [GaBu2001] Gardner, J. A.; Bulatov, V.: Smart Figures, SVG, and Accessible Web Graphics, Technology And Persons With Disabilities Conference, 2001
- [Gard2002] Gardner, J. A.: Access by Blind Students and Professionals to Mainstream Math and Science, Oregon State University Corvallis, 2002
- [HeScVo2003] Henneberger, S.; Schulz, M.; Voss, J.: Save as XDiML, Writing and Converting digital Theses and Dissertations using OpenOffice.org, OpenOffice.org Conference OOoCon 2003
- [HeDa2002] Herman, I.; Dardailler, D.: SVG Linearization and Accessibility, Proceedings of the SVG Open Conference 2002
- [ITO] Information Technology Online (ITO): Information Technology Online
- [Lambda] Lambda Project: Linear Access to Mathematic for Braille Device and Audio-synthesis (LAMBDA), <http://www.lambdaproject.org>
- [Jack2003] Jackson, D.: Making Accessible SVG, SVG Open 2003
- [KGMPG2001] Karshmer, A.; Gupta, G.; Miesenberger, K.; Pontelli, E.; Guo, H-F.: The Development of a Tool to Enhance Communication Between Blind and Sighted Mathematicians, Proceedings of the 9th International Conference on Human-Computer Interaction, Lawrence Erlbaum Associates, 2001
- [KaTo1996] Kawai, Y.; Tomita, F.: Interactive tactile display system: a support system for the visually disabled to recognize 3D objects, Proceedings of the second annual ACM conference on Assistive technologies, 1996

- [Kurz1997] Kurze, M.: Interaktion Blinder mit virtuellen Welten auf der Basis von zwei-dimensionalen taktilen Darstellungen, Proceedings of the Software-Ergonomie 97, 1997
- [McKo2002] McCathieNevile, C.; Koivunen, M.-R.: Accessibility Features of SVG, 2002
- [Metacoon] Metacoon-Services: metacoon-services homepage, <http://www.metacoon.de/>
- [Oelt2003] Oeltjen, W.: OpenOffice.org as an Authoring Tool in the FIGARO Project, OpenOffice.org Conference OOoCon 2003
- [RVKRL2002] Rawlings, A.; van Rosmalen, P.; Koper, R.; Rodríguez-Artacho, M.; Lefrere, P.: Survey of Educational Modelling Languages (EMLs), CEN/ISSS WS/LT Learning Technologies Workshop, 2002
- [RoScEr2002] Rotard, M.; Schweikhardt, W.; Ertl, T.: Verwendung von Scalable Vector Graphics und MathML in web-basierten Lernumgebungen, Mensch und Computer, Workshop Computer Support for Collaborative Learning, 2002
- [RBSE2003] Rotard, M.; Bosse, K.; Schweikhardt, W.; Ertl, T.: Access to Mathematical Expressions in MathML for the Blind, Universal Access in Human Computer Interaction, HCI International Conference, Lawrence Erlbaum Assoc., 2003
- [RoFi2003] Rotard, M.; Finsterle, L.: Creating and Publishing Courseware Independent of the Authoring System, Technology Enhanced Learning Conference, 2003
- [RoOter2004] Rotard, M.; Otte, K.; Ertl, T.: Exploring Scalable Vector Graphics for Visually Impaired Users, Proceedings of the 9th International Conference on Computers Helping People with Special Needs, 2004
- [RoEr2004] Rotard, M.; Ertl, T.: Tactile Access to Scalable Vector Graphics for People with Visual Impairment, Proceedings of the SVG Open Conference, 2004
- [RoFi2004] Rotard, M.; Finsterle, L.: Erstellung lernplattformunabhängiger Lehrmodule, In: Peter Göhner (Hrsg.), Information Technology Online , Waxmann, 2004
- [RoScEr2004] Rotard, M.; Schweikhardt, W.; Ertl, T.: Aufbereitung von Lehrmaterialien für sensorisch behinderte Menschen, In: Peter Göhner (Hrsg.), Information Technology Online, Waxmann, 2004
- [ScWiZe1930] Scheid, F. M.; Windau, W.; Zehme, G., System der Mathematik- und Chemie-schrift für Blinde, Deutsche Blindenstudienanstalt, Marburg a.d. Lahn, 1930
- [Schw1996] Schweikhardt, W.: Interactive Exploring of Printed Documents by Visually impaired People, Interdisciplinary Aspects on Computers Helping People with Special Needs, 1996
- [Schw1980] Schweikhardt, W.: A Computer Based Education System for the Blind, In: Lavington, S. H. (Hrsg.): Information Processing 80, Holland Publishing, 1980
- [UIML] UIML.org: User Interface Markup Language, <http://www.uiml.org>
- [XSLTML] Golovin, P.; Yaroshevich, V. I.; Hertz, T.: XSLT MathML Library, <http://sourceforge.net/projects/xsltml/>