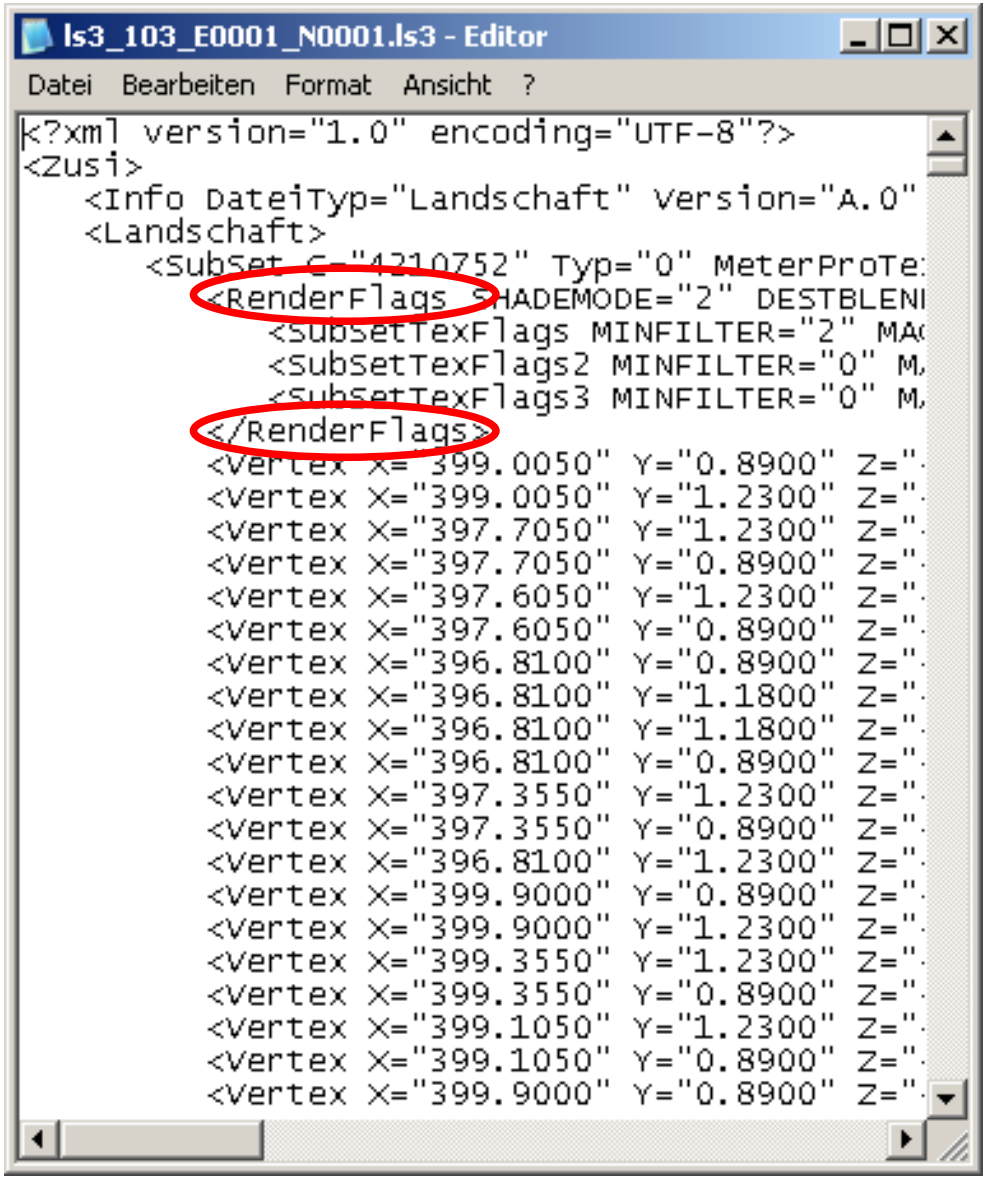
XML-Format

- „xml“: extensible markup language
- Datenformat aus Knoten und Attributen in einer Hierarchie (wie z.B. Dateien im Explorer)
- Format ist selbstbeschreibend
- Daten werden über einen Text-Bezeichner identifiziert (Name=“Wert“, z. B. XKoordinate = “36.34“)
- Dateiformat ist ASCII, also mit jedem Texteditor zu öffnen
- xml-Editoren erleichtern die Handhabung durch strukturierte Darstellung
- xml-Editoren lesen die Syntax, ohne den Inhalt „zu verstehen“
- Verschiedene Editorkonzepte werden jetzt kurz vorgestellt

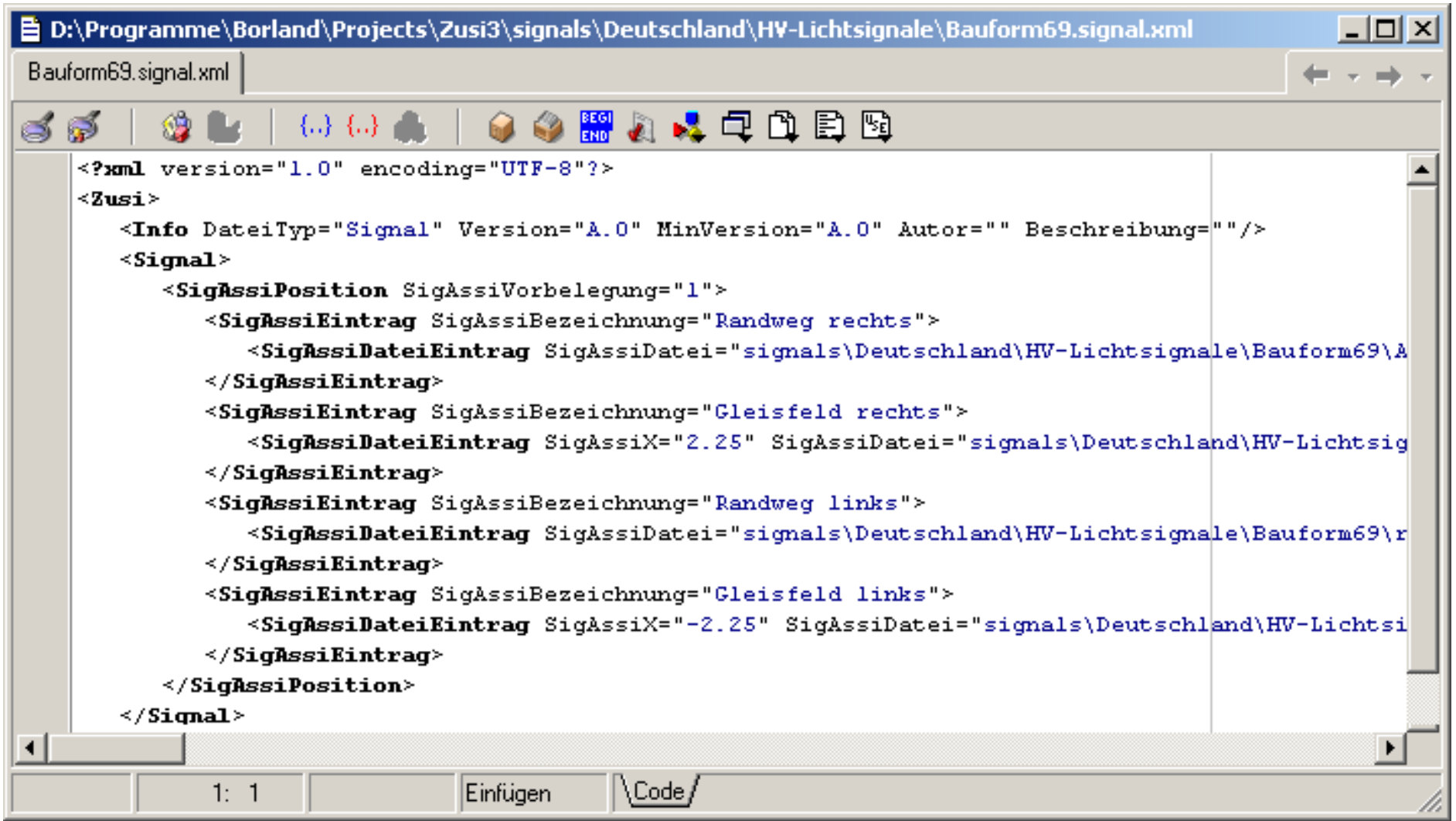
xml-Datei: Quelltext

- Darstellung: MS-Notepad
- Knoten nur mit Attributen definiert durch:
  <name [Attribute...] />
- Knoten mit Unterknoten:
  <name> [Attribute...]
  [Unterknoten...]
  </name>
- Attribute definiert mit Name = “Wert“, sind einem Knoten zugeordnet
- Einrückung und Zeilenumbrüche formal nicht nötig, erhöhen aber die Lesbarkeit im Texteditor



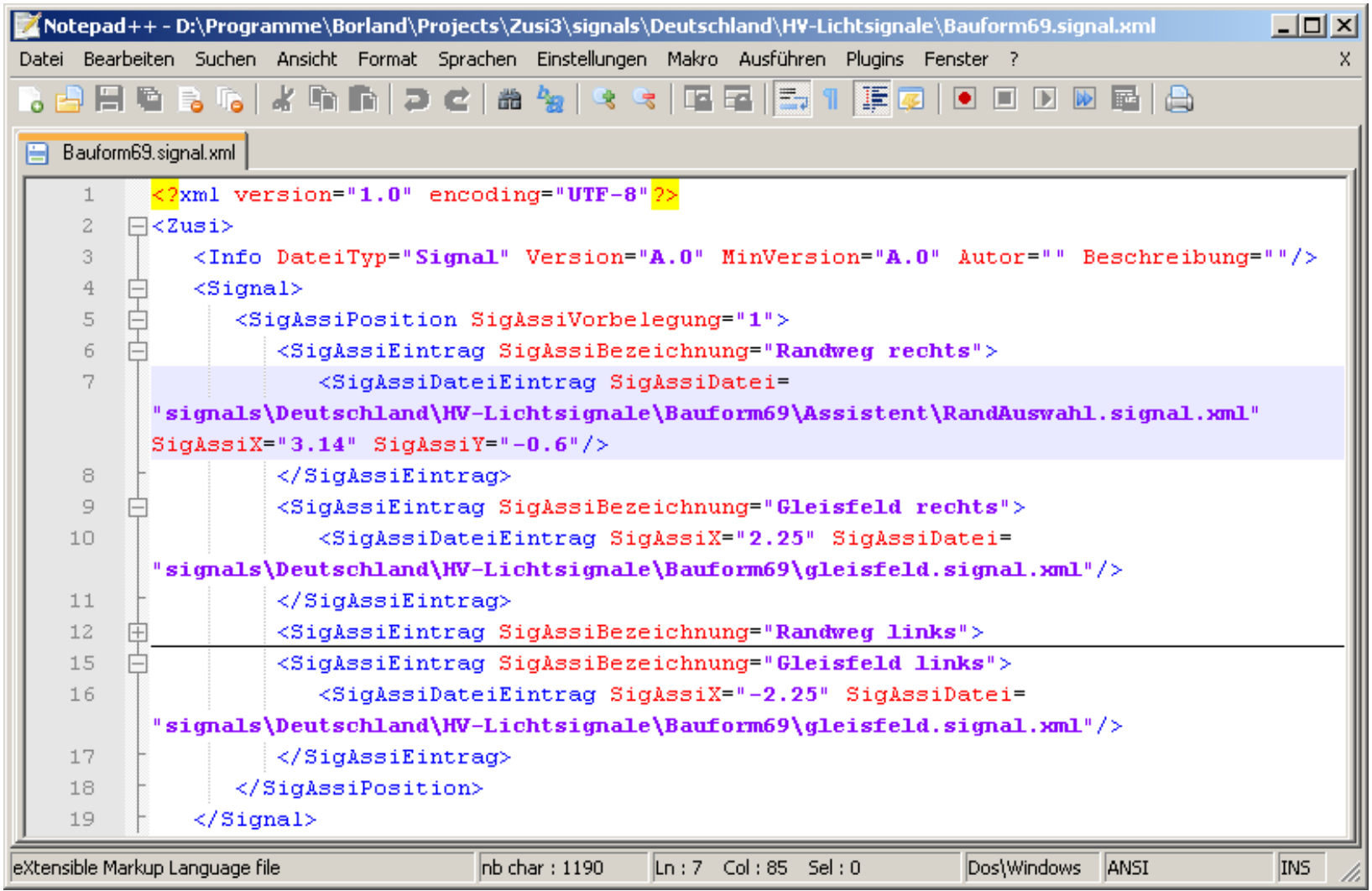
xml-Editor mit Syntaxhervorhebung

- Datei wird als Quelltext dargestellt, aber die Attribute, Werte und Knoten werden farblich hervorgehoben. Beispiel Delphi-IDE:



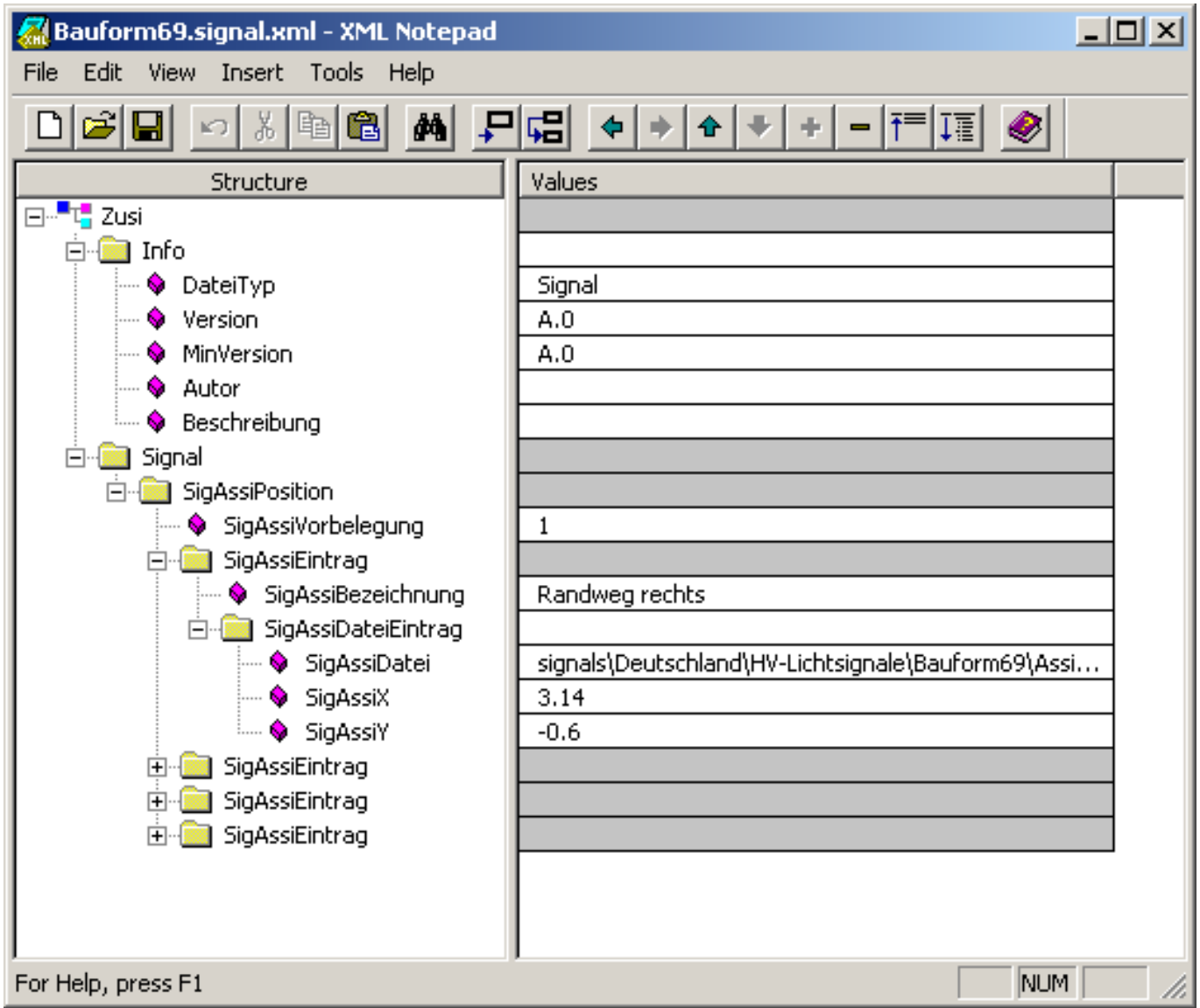
xml-Quelltexteditor mit Hierarchie

- Daten werden als Quelltext mit farblicher Kennung angezeigt und können wie im Explorer reduziert werden. Beispiel Notepad++:



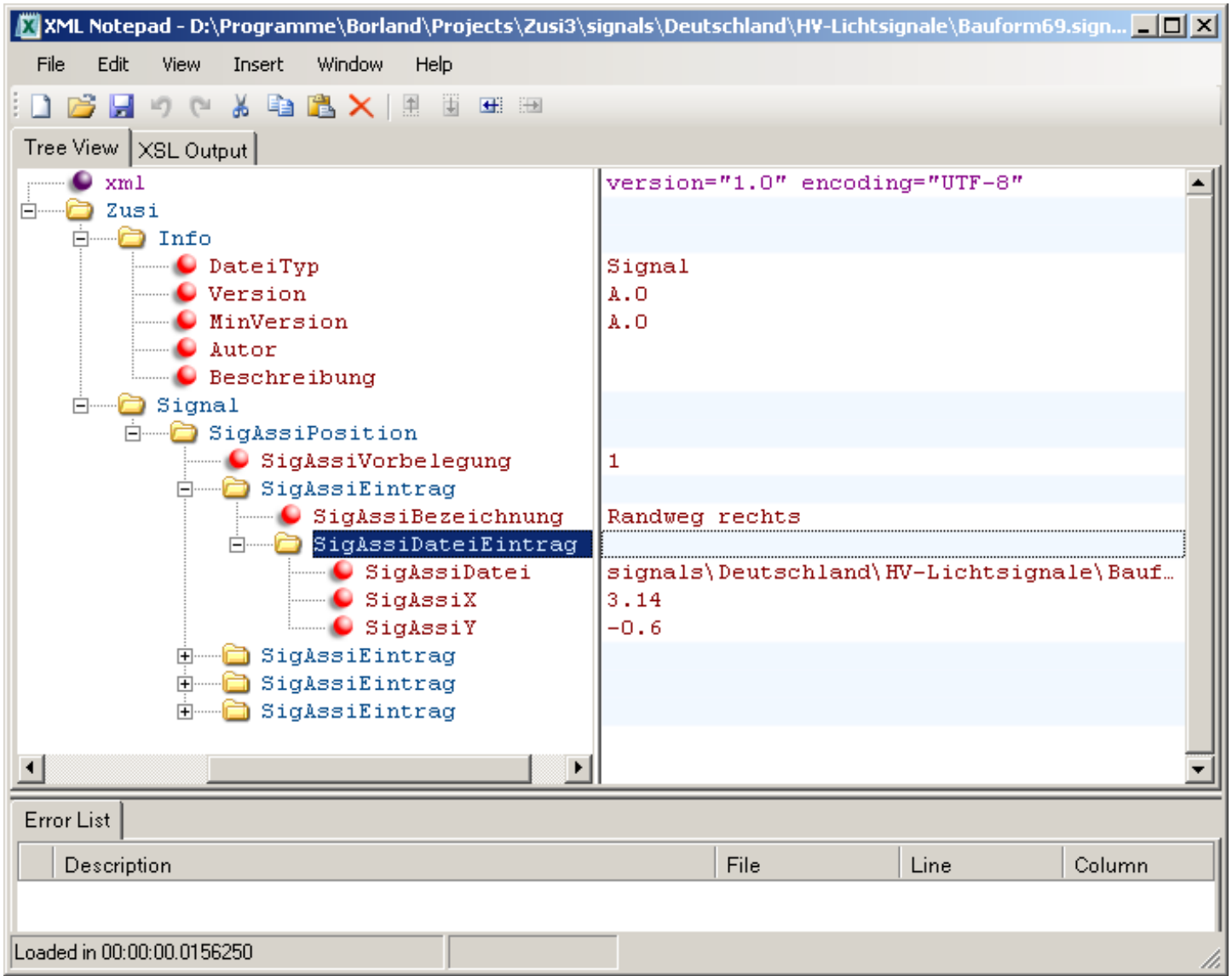
xml-Editor mit Eingabeoberfläche

- Im Quellcode muß nicht mehr editiert werden. Beispiel: MS-xml-Notepad 1.5 beta (1998):



Aktuelles MS-xml-Notepad

- 2007 erschien dann die endgültige MS-Notepad-Version als xml-Notepad 2007:

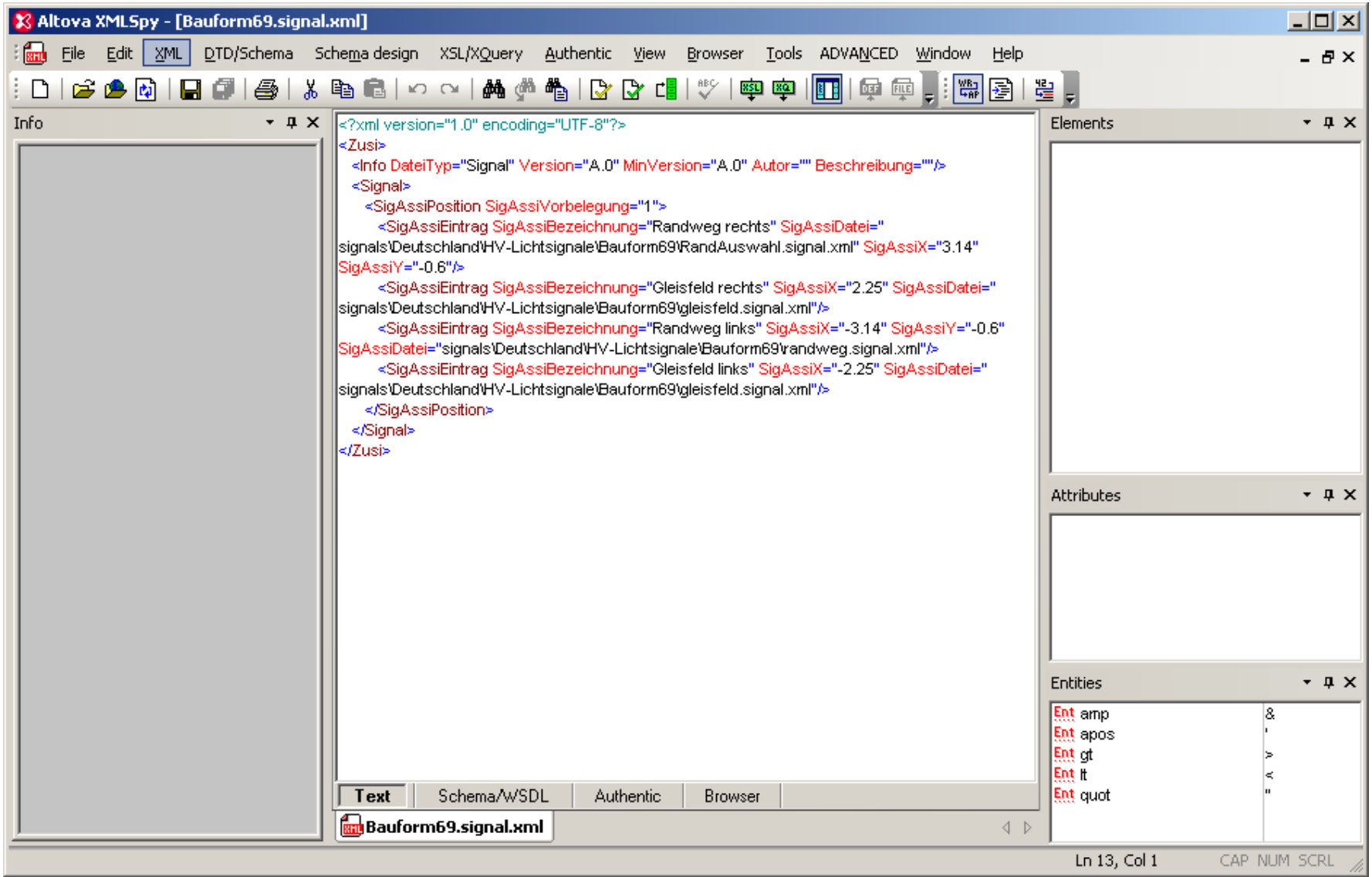


xml-Vorteile

- Dateiformat ist standardisiert: Für Einlesen und Speichern müssen keine eigenen Routinen programmiert werden (z.B. Zusi-Tools)
- Format ist ohne Konflikte erweiterbar, so daß sogar neuere (eigentlich unbekannte) Dateiformate gelesen werden können, wobei nur die neuen Attribute ignoriert werden
- Kaum Dokumentation nötig, da das Format selbstbeschreibend ist
- Es kann ein Schema des spezifischen (Zusi-)Formats erstellt werden, womit Editoren (wie Notepad 2007) sogar den Inhalt einer xml-Datei auf korrekte Werte überprüfen können. Es wird dabei in der Schema-Datei für jeden Knoten definiert, welche Unterknoten, Attribute und Werttypen zulässig sind

Profi-Editor

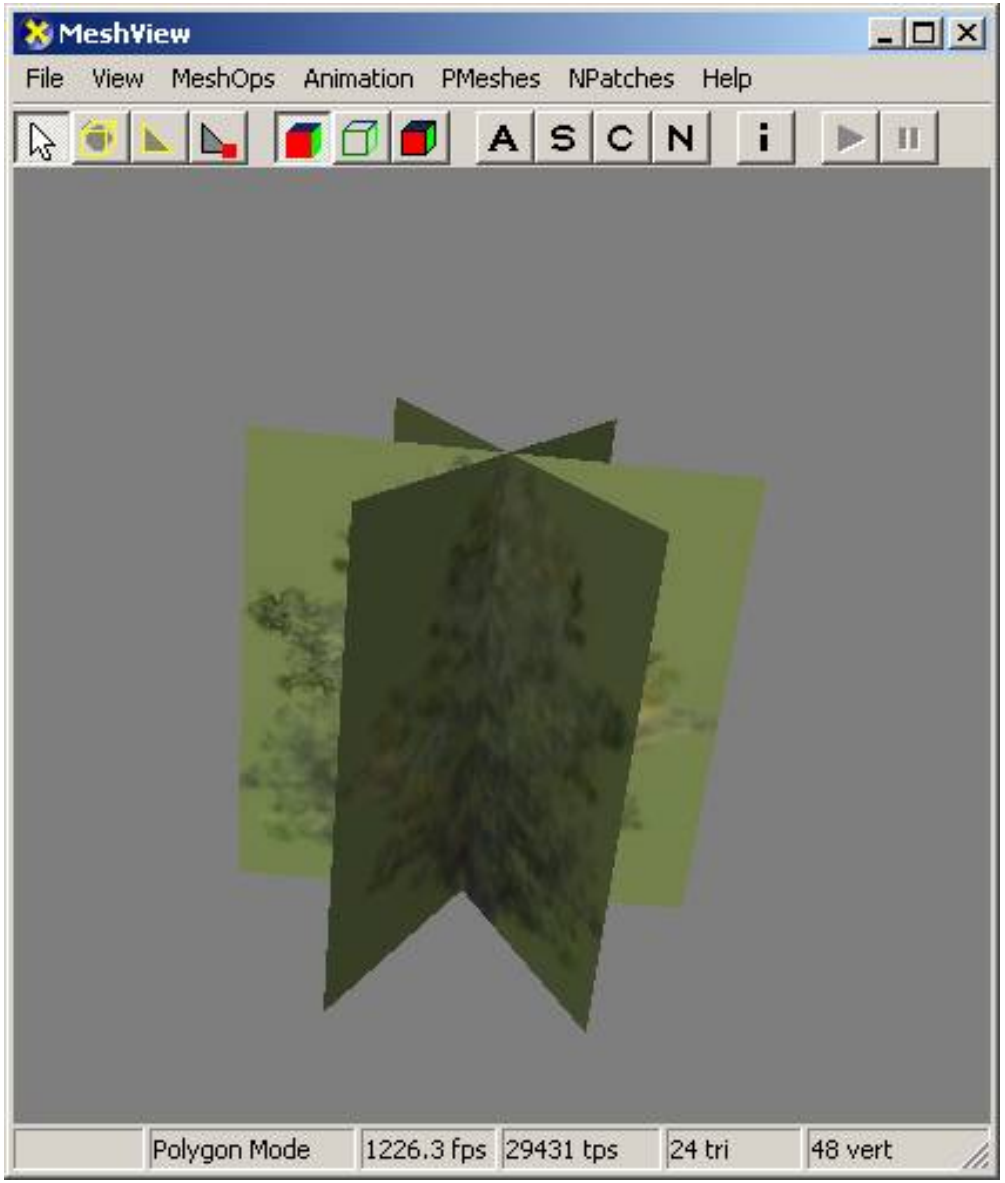
- Professioneller Editor mit Schema-Erstellung usw.: Altova XML-Spy (Basisversion kostenlos)



Mesh-Viewer aus DirectX-SDK

- Referenz-Darstellung für .x-Dateien
- Anwendung z.B. Test auf korrektes Dateiformat
- .x-Datei-Hierarchie
- Animation darstellbar
- Normalenvektoren
- Keine Transparenz
- Umwandlung ASCII/binär .x-Datei
- Erzeugung einfacher Körper

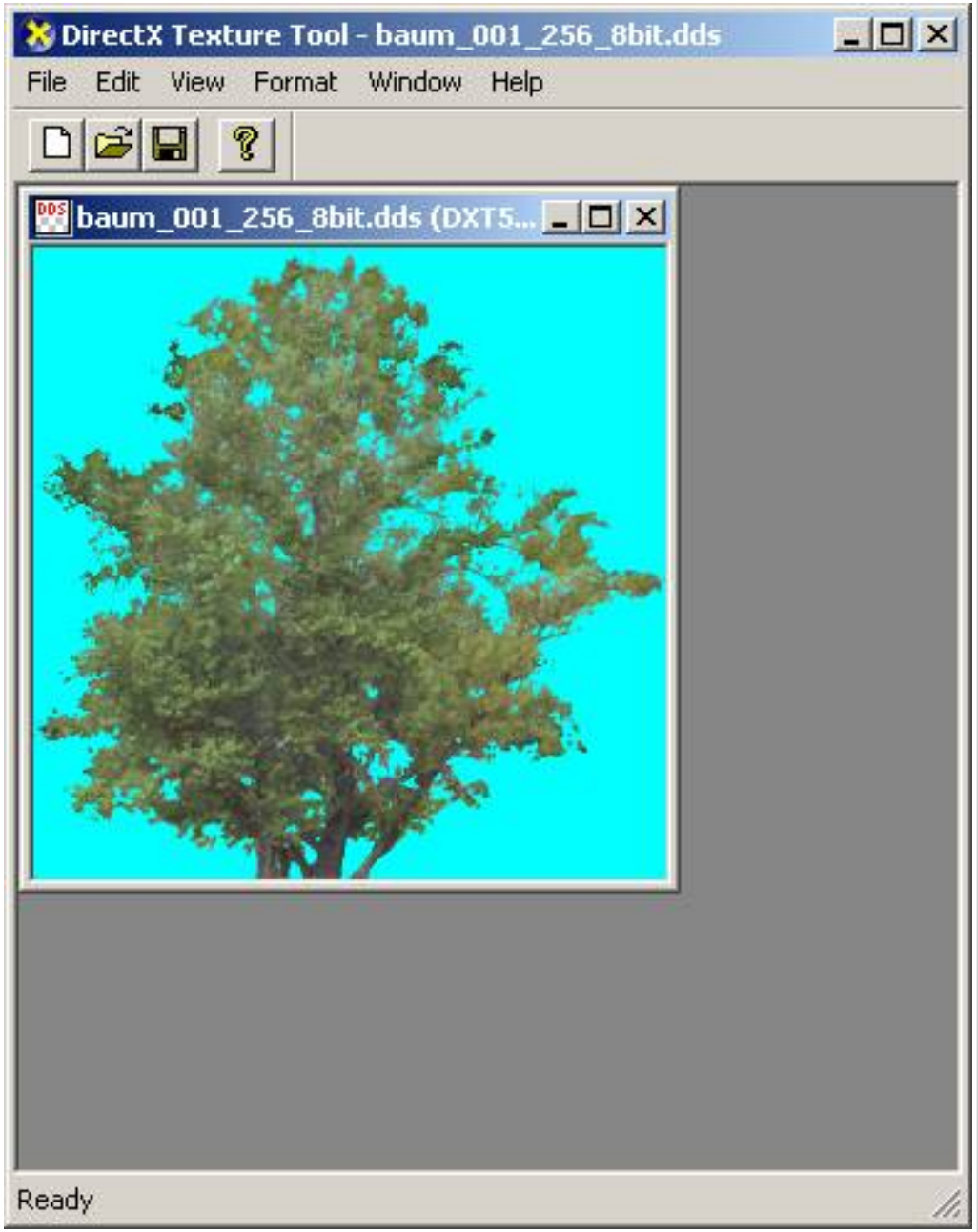
Dateiname: mview.exe



Texture-Tool aus DirectX-SDK

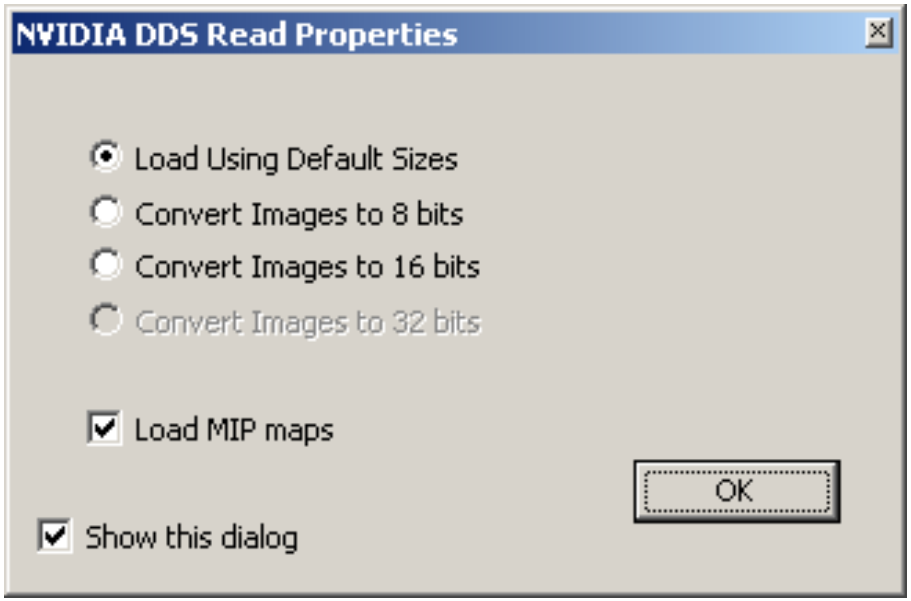
- Import gängiger Grafikformate
- Export als dds
- In dds-Unterformate umwandeln
- Alphakanal erzeugen bzw. dazuladen
- Mipmaps einzeln anzeigen
- Mipmaps erzeugen oder einzeln laden

Dateiname: DxTex.exe

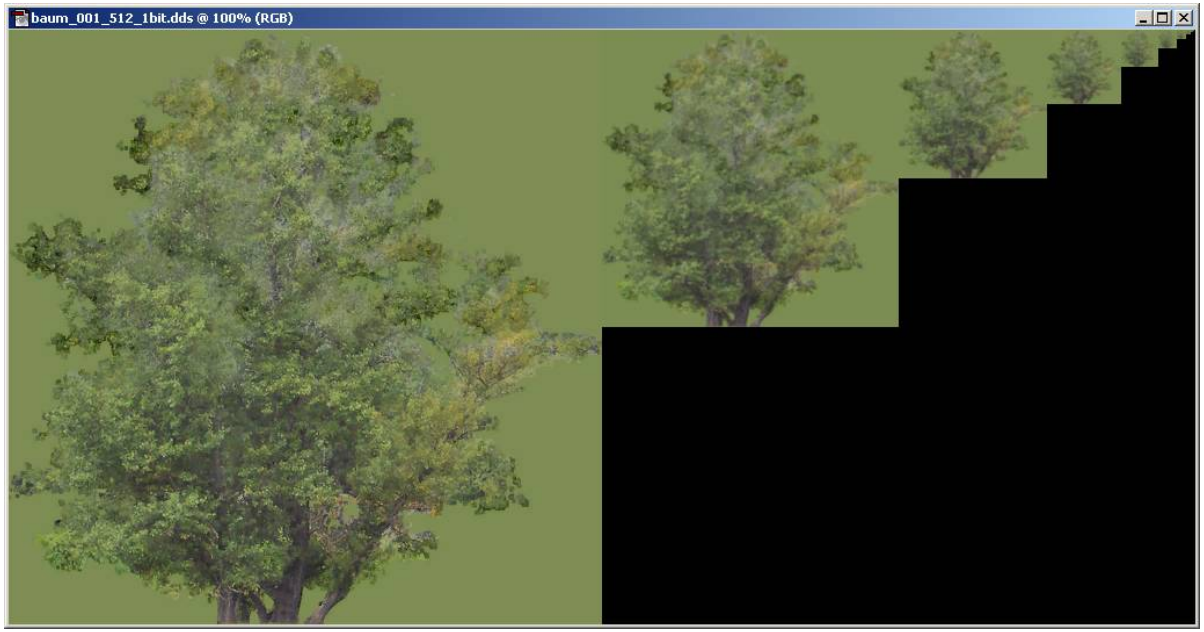


Adobe-Photoshop-dds-Plugin

- Beim Öffnen einer dds-Datei können die in der dds-Datei enthaltenen Mipmaps geladen werden



- Anzeige und Bearbeitung aller Mipmaps in einem Bild



Adobe-Photoshop dds-Export

- „Speichern unter“: dds
- DXT1- oder DXT3-Unterformat
- 2D Texture
- Use Existing Mipmaps speichert die Mipmaps wie vorhanden
- Generate Mipmaps erzeugt die Mipmaps aus dem Bildinhalt (in diesem Beispiel unsinnig, da Mipmaps schon vorhanden)
- Rest kann in Standardeinstellung bleiben