



# Toolkits für die Visualisierung

Recherche im Bereich .NET

**Tim Lammarsch**

**Danube University Krems**

Department of Information and Knowledge Engineering (ike)

Authors: **Tim Lammarsch**

[tim.lammarsch@donau-uni.ac.at](mailto:tim.lammarsch@donau-uni.ac.at)  
<http://www.donau-uni.ac.at/ike>

Contact: **Danube University Krems**  
Department of Information and Knowledge Engineering (ike)

Dr.-Karl-Dorrek-Str. 30  
3500 Krems  
Austria, Europe

**T** +43 (2732) 893 - 245x

**F** +43 (2732) 893 - 4450

[ike@donau-uni.ac.at](mailto:ike@donau-uni.ac.at)

<http://www.donau-uni.ac.at/ike>

# Inhaltsverzeichnis

<b>1 Grundlagen</b>	<b>1</b>
1.1 Lizenzen . . . . .	1
1.2 Webcontrols . . . . .	1
<b>2 Toolkits</b>	<b>2</b>
2.1 Dart Chart . . . . .	2
2.2 .netCHARTING . . . . .	2
2.3 Dundas Data Visualization . . . . .	2
2.4 NPlot . . . . .	3
2.5 GO Diagram . . . . .	3
2.6 Piccolo . . . . .	3
2.7 Prefuse . . . . .	3
2.8 Quinn-Curtis Chart 2D . . . . .	4
2.9 Swordfish Charts . . . . .	4
2.10 ZedGraph . . . . .	4
<b>3 Genauere Erklärungen zu Schwachpunkten</b>	<b>5</b>
3.1 Quellcode . . . . .	5
3.2 HitTest . . . . .	5
3.3 Farbwechsel . . . . .	5
3.4 Erstellen von Webcontrols . . . . .	5

## **Zusammenfassung**

Dieses Dokument soll einen Überblick darüber bieten, welche Toolkits es gibt, die das Erzeugen von serverseitig gerenderten Visualisierungen in oder zumindest für .NET ermöglichen und für welche Einsatzgebiete die einzelnen Toolkits am besten geeignet sind.

Toolkits, die vielversprechender waren, wurden deutlicher untersucht, als Toolkits, bei denen frühzeitig ein klares Problem festgestellt wurde. Dennoch wurde kein Toolkit ganz außer Acht gelassen.

# Kapitel 1

## Grundlagen

### 1.1 Lizenzen

**GPL Version 2** Der Code darf frei verwendet und modifiziert werden. Der veränderte Code muss freigegeben werden. Alles, was dagegen gelinkt wird, muss freigegeben werden. Wie es mit DLLs aussieht ist nicht geklärt, aber GPL ist definitiv kritischer als LGPL.

**LGPL Version 2** Der Code darf frei verwendet und modifiziert werden. Der veränderte Code muss freigegeben werden. Wenn die Control als DLL angebunden wird, muss der die Control verwendende Code nicht freigegeben werden.

**3-Klausel-BSD-Lizenz** Der Code darf frei verwendet und modifiziert werden. Die original-Lizenz muss mit ausgeliefert werden. Freigegeben werden muss nichts.

**Quinn-Curtis-Lizenz** Kommerzielle Lizenz. Pro Entwickler: 150 Dollar. Pro Installation: 50 Dollar. Source Code: 500 Dollar.

**.netCHARTING-Lizenz** Kommerzielle Lizenz. Keine Kosten pro Entwickler. Pro Website: 395 Dollar. Eine TIS-Installation wäre wohl eine „Website“. Kein Source Code.

### 1.2 Webcontrols

Webcontrols stellen die Schnittstelle dar, mit der die Toolkits von TIS aus verwendet werden. Manche Toolkits kommen mit Webcontrol, manche nicht. Kein Toolkit verfügt innerhalb der Webcontrol über Interaktivität. Das Erstellen einer einfachen Webcontrol ist relativ einfach, weshalb das Fehlen einer Webcontrol kein großes Problem ist. Dies gilt umso mehr, weil auch bei vorhandenen Webcontrols wegen der Interaktivität viel getan werden muss. Vorhandene Interaktivitäts-Features der Toolkits lassen sich kaum nutzen, da sie dafür nicht vorgesehen sind.

# Kapitel 2

## Toolkits

### 2.1 Dart Chart

**Lizenz** GPL Version 2

**Plattformen** .NET 1.1 oder 2.0

**Erweiterbarkeit** Quellcode verfügbar.

**Funktionsumfang** Ungefähr so groß wie der eines der Visual Studio Templates.

**Webanbindung** Nein.

### 2.2 .netCHARTING

**Lizenz** .netCHARTING-Lizenz.

**Plattformen** .NET 1.1 und .NET 2.0.

**Erweiterbarkeit** Quellcode nicht verfügbar. Dadurch keine eigenen Visualisierungen. Erweiterbarkeit für ein Produkt ohne Quellcode zwar sehr gut, aber nicht ausreichend.

**Funktionsumfang** Kurven, Balken, diverse nicht verwendbare Charttypen, zwei Achsen pro Dimension, automatische Skalen, rudimentäre Zeitachsen, lineare und logarithmische Skalen, HitTest, Farbwechsel.

**Webanbindung** Ja, inklusive Interaktivität.

### 2.3 Dundas Data Visualization

**Lizenz** Kommerziell, nicht geprüft.

**Plattformen** .NET 1.1 oder 2.0

**Erweiterbarkeit** Quellcode nicht verfügbar, auch sonst nicht besonders gut erweiterbar.

**Funktionsumfang** Viel vorgefertigtes, aber wesentlich Punkte nicht, die .netCHARTING bietet.. Daher nicht weiter untersucht.

**Webanbindung** Ja.

## 2.4 NPlot

**Lizenz** 3-Klausel-BSD-Lizenz

**Plattformen** .NET 1.1, Mono. Bereits erfolgreich konvertiert nach .NET 2.0.

**Erweiterbarkeit** Quellcode verfügbar. Der Code ist relativ simpel gestaltet und daher gut erweiterbar.

**Funktionsumfang** U.a. Kurven, Balken, Stufen, zwei Achsen pro Dimension, automatische Skalen, rudimentäre Zeitachsen, lineare, ordinale und logarithmische Skalen.

**Webanbindung** Eine Webcontrol ist vorhanden. Diese hat relativ wenig Funktionalität, aber die Erweiterung ist relativ einfach.

## 2.5 GO Diagram

**Lizenz** Kommerziell, nicht geprüft.

**Plattformen** .NET 1.1 oder 2.0, Java, MFC

**Erweiterbarkeit** Quellcode nicht verfügbar.

**Funktionsumfang** Geht ein Stück weit an unseren Zielen vorbei, eher Richtung Visio. Daher nicht weiter untersucht.

**Webanbindung** Ja.

## 2.6 Piccolo

**Lizenz** 3-Klausel-BSD-Lizenz

**Plattformen** Java, .NET 1.1 (konvertierbar nach .NET 2.0), inklusive .NET Compact Framework

**Erweiterbarkeit** Quellcode verfügbar. Genauere Tests wurden unterlassen, da der Funktionsumfang evtl. zu gering ist.

**Webanbindung** Eine Webcontrol ist nicht vorhanden. Obwohl es relativ einfach ist, eine zu bauen, muss dabei fast die gesamte Funktionalität des Frameworks neu implementiert werden.

**Funktionsumfang** Mächtige Grafikroutinen, aber kein Code für Schaubilder. Die Grafikroutinen machen praktisch nur in einer Windows-Forms-Control Sinn. Eine Einbindung dieser Control im Internet Explorer scheint hier aber erstaunlicherweise problemloser möglich zu sein als bei allen anderen Versuchen, dies zu bewerkstelligen.

## 2.7 Prefuse

**Lizenz** 3-Klausel-BSD-Lizenz

**Plattformen** Java

**Erweiterbarkeit** Quellcode verfügbar. Dieser wurde bislang nicht genauer untersucht.

**Webanbindung** Eine Webcontrol ist nicht vorhanden. Es ist möglich, eine zu bauen, ein Teil der Funktionalität muss dabei aber neu implementiert werden.

**Funktionsumfang** Extrem mächtige Grafikroutinen. Kein Code für Schaubilder, dieser lässt sich aber mit Prefuse einfacher erzeugen als mit rudimentären Grafikroutinen. Dafür andere Dinge, wie automatischer FishEye-View und ein Hittest sowohl über BoundingBoxen als auch mit direktem Abgleich. Alles in allem das mächtigste Toolkit.

## 2.8 Quinn-Curtis Chart 2D

**Lizenz** Quinn-Curtis-Lizenz

**Plattformen** .NET 1.1, .NET 2.0. (Erfolgreich getestet mit .NET 2.0.)

**Erweiterbarkeit** Quellcode verfügbar, aber kostenpflichtig. Daher noch nicht evaluiert. Es fällt auf, dass selbst bei Verwendung des fertigen Codes noch relativ viel „Handarbeit“ anfällt, das sollte aber die Erweiterung nicht unbedingt erschweren.

**Funktionsumfang** Unter anderem Kurven, Balken, zwei Achsen pro Dimension, automatische Skalen, Zeitachsen, lineare, ordinale und logarithmische Skalen. Insgesamt überdurchschnittlich viele brauchbare Features.

**Webanbindung** Eine Webcontrol ist nicht vorhanden. Wie eingangs erwähnt ist es verhältnismäßig einfach, eine zu bauen.

## 2.9 Swordfish Charts

**Lizenz** 3-Klausel-BSD-Lizenz

**Plattformen** .NET 3.0 – nicht abwärtskompatibel da WPF.

**Erweiterbarkeit** Quellcode verfügbar.

**Funktionsumfang** Nicht überprüft.

**Webanbindung** Nicht überprüft.

## 2.10 ZedGraph

**Lizenz** LGPL Version 2

**Plattformen** .NET 1.1, .NET 2.0

**Erweiterbarkeit** Quellcode verfügbar. Der Code ist hochintegriert in einer Art und Weise, die Modifikationen erschwert. Es gelang z.B. nicht, mit geringem Arbeitszeitaufwand eine Zoom-Funktionalität für die Webcontrol zu erstellen.

**Webanbindung** Eine Webcontrol ist vorhanden. Diese hat relativ wenig Funktionalität und die Erweiterung ist schwierig, da der Zugriff auf dafür notwendige Funktionen vom integrierten Konzept erschwert wird.

**Funktionsumfang** U.a. Kurven, Balken, zwei Achsen pro Dimension, automatische Skalen, rudimentäre Zeitachsen, lineare, ordinale und logarithmische Skalen.

## Kapitel 3

# Genauere Erklärungen zu Schwachpunkten

### 3.1 Quellcode

Quellcode ist kein Selbstzweck, aber kein Toolkit ist so perfekt, dass man nicht etwas verbessern müsste. Gleichzeitig ist es aus wissenschaftlicher Sicht unabdingbar, eigene Charttypen zu verwenden, die über das in der bisherigen Praxis verwendete hinausgehen.

Diese Tatsache ist sehr schade, vor allem in Hinblick darauf, dass beispielsweise `.netCHARTING` sehr mächtig ist und viele Features bietet, die anderweitig nur mit Aufwand zu realisieren sind. Dennoch ist der fehlende Quellcode hier eine entscheidende Einschränkung, da es bei allen Fähigkeiten nicht perfekt ist.

### 3.2 HitTest

Von den Toolkits mit Quellcode hat nur Prefuse hat einen verwendbaren HitTest. `Piccolo` und `ZedGraph` haben einen BoundingBox-basierten, `NPlot` hat einen nur für Achsen. Es ist daher damit zu rechnen, dass bei den `.NET`-basierten Toolkits eigener Code gefragt ist, evtl. über die Erweiterung der `Graphics`-Klasse in Verbindung mit einem HitTest-Interface für Plots.

### 3.3 Farbwechsel

Spezielle Zeichenmethoden wie Farbwechsel innerhalb eines Plots, z.B. zum Hervorheben von falschen oder fehlenden Werten, wird von keinem Toolkit mit Quellcode unterstützt. Dies betrifft nur die Toolkits mit Schaubilder-Funktion, da das bei `Piccolo` und `Prefuse` sowieso neu kodiert werden müsste, ist dieser Schwachpunkt dort kein zusätzlicher Schwachpunkt.

### 3.4 Erstellen von Webcontrols

Eine rudimentäre Webcontrol lässt sich sehr einfach erzeugen. Interaktivität ist aber um so komplizierter zu bewerkstelligen, je mehr die Control können soll. Dies muss immer über JavaScript gehen.

Auch Java-Applikationen lassen sich als Renderer für Webcontrols relativ einfach einbinden. Hier ist der Datentransfer aber aufwendiger. Außerdem sind sie bislang deutlich langsamer, da jedesmal ein neuer Prozess gestartet werden muss. Dieses Manko lässt sich aber voraussichtlich mit mehr Entwicklungsaufwand ausmerzen.