

---

Visual Mining, Inc.

# **NetCharts Pro 4.0 Migration Guide**

**A Guide to migrating NetCharts Beans programs to  
NetCharts Pro 4.0**

---

## Table of Contents

<b>1.0 SCOPE</b> .....	<b>3</b>
<b>2.0 OVERVIEW</b> .....	<b>4</b>
NEW FEATURES IN NETCHARTS PRO .....	4
MIGRATION STRATEGIES .....	5
<b>3.0 API CHANGES BETWEEN NETCHARTS BEANS AND NETCHARTS PRO</b> .....	<b>6</b>
PACKAGE NAME CHANGES .....	6
LICENSE API CHANGES .....	6
CHART OBJECT NAME CHANGES .....	7
CONSTRUCTOR CHANGES .....	7
THE SET() API .....	8
IMAGE GENERATION CHANGES.....	8

## 1.0 Scope

This document provides programmers with guidelines for migrating applications using NetCharts Beans to NetCharts Pro. NetCharts Pro and NetCharts Beans are Java programmer-friendly implementations of Visual Mining's suite of chart components. A companion document, *Programming with NetCharts Pro*, provides additional useful information on the effective use of NetCharts Pro. Additional information on Java is available from Sun at <http://www.javasoft.com/products/java>.

**Note to our customers:**

Thank you for evaluating or upgrading to NetCharts Pro Version 4.0. We sincerely believe that the charts and chart images produced by NetCharts Pro 4.0 are among the most robust online charts available.

Please direct any questions or comments on this product to [support@visualmining.com](mailto:support@visualmining.com).

—*The Visual Mining Team*



# 2.0 Overview

NetCharts Pro is Visual Mining's newest component library for professional programmers needing to add dynamic charts to Java applets, web applications or desktop applications. NetCharts Pro (NCPro) contains all the features found in NetCharts Beans (NCB). In addition NetCharts Pro contains new features based on feedback from our NetCharts Beans customers.

## ***New Features in NetCharts Pro***

The most significant change for programmers is NCPro's improved programming API. Each component of a chart has a new object representation of its state, and all the chart types in NCPro contain new get and set methods that accept these state objects. For example, chart axes can be created and manipulated through NFAxis objects. Individual attributes of an axis such as color and tic layout can be set through methods on NFAxis. This presents a much more logical and familiar interface for Java programmers and is an improvement over the string based API in NCB.

NetCharts Pro also introduces the concept of chart data models. Programmers can write classes that implement a NetCharts Pro data model interface to extract chart data from arrays, JDBC result sets, DOM objects or any other data structure or source. Data model interfaces are provided for one, two and three-dimensional data and certain chart-specific types like stock charts and time charts.

NetCharts Pro is based on Java Swing, or lightweight, components. NetCharts Beans was based on Java AWT, or heavyweight components. Migrating desktop applications from NCB to NCPro will involve moving from AWT to Swing based graphics as well. Applets using NCPro will require the use of a Java plug-in in the client browser in order to support Swing.

NetCharts Pro 4.0 contains charting features that are not present in NetCharts Beans 4.0. These include:

- **New Font Specifications.** NetCharts Pro and NetCharts Beans charts use the "bold" style of a font by default. Programmers may override this behavior in NetCharts Pro by adding the word "Plain" to the end of a font specification.
- **New CDL Parameter - Dial Clip.** NetCharts Pro introduces the DialClip CDL parameter. The DialClip parameter allows chart developers to specify a clip area to be defined within a dial chart. This parameter can be used to show any half or any quarter of a dial chart.
- **Improved Axis Autoscale Behavior.** Previously, all three attributes of the Scale CDL parameter needed to be specified or the chart would be auto-scaled. NetCharts Pro now allows you to specify any of the three attributes of the scale. If only one or two of the attributes are specified, NetCharts Pro will auto-scale only those attributes that are unspecified. This allows users, for example, to specify a minimum and a step size, and have NetCharts Pro automatically select the maximum value.
- **New Piechart parameters.** PieSize sets minimum and maximum sizes for the actual pie in a pie chart, allowing programmers to guarantee that multiple pie charts on a single page will all have the same size. Also, SliceLabelContent has been modified to allow more control over the content of pie slice labels.

- New output formats. NetCharts Pro contains support for the creation of TIFF and WBMP image types.

### ***Migration Strategies***

Visual Mining recommends that NetCharts Bean's users consider migrating their applications to NetCharts Pro. Care has been taken in the design of NetCharts Pro to make the transition from NetCharts Beans to NetCharts Pro as simple as possible, which presents programmers with options when planning a migration.

Programmers may follow the guidelines in this document for translating API calls from NCB to NCPro. This represents the shortest and least expensive path for moving standard NCB programs to NCPro programs, but it will not take advantage of any of the new features in NCPro.

Programmers may also upgrade their applications to take full advantage of the new programming and charting features in NCPro. While this may take more time, the end result should be smaller more manageable code.

# 3.0 API changes between NetCharts Beans and NetCharts Pro

The programming guides for NetCharts Beans 4.0 and NetCharts Pro 4.0 recommend the same basic model for chart-enabling applications. NCB applications that followed this model can be migrated to NCPPro without any architectural changes, although they would not take advantage of any new NetCharts Pro features. The following API level changes would be required in such an exercise.

## ***Package Name Changes***

The names of the packages imported into a chart-enabled class have changed.

In NCB, classes typically imported the following packages

```
import netcharts.util.*;
import netcharts.graphics.NFLicense2;
import chartworks.cb.util.*;
import chartworks.cb.beans.*;
```

In NCPPro, classes will typically need to import these packages

```
import netcharts.pro.common.*;
import netcharts.pro.util.*;
```

## ***License API Changes***

The package, class and method name used to programmatically set the product license keys have changed. The content of the license key strings has also changed.

In NCB, the calls to set the license keys were of this form:

```
netcharts.graphics.NFLicense2.setKey("NetChartsBeans4.0 KEY=...");
netcharts.graphics.NFLicense2.setKey("NetCharts4.0 KEY=...");
```

In NCPPro, the calls to set the license keys are of this form:

```
netcharts.pro.common.NFGraph.setLicenseKey("NetChartsPro4.0 KEY=...");
netcharts.pro.common.NFGraph.setLicenseKey("NetCharts4.0 KEY=...");
```

## Chart Object Name Changes

The fully qualified names of the chart objects have changed according to the following table.

NetCharts Beans	NetCharts Pro
chartworks.cb.beans.NFBarchart	netcharts.pro.charts.bar.NFBarchart
chartworks.cb.beans.NFBoxchart	netcharts.pro.charts.box.NFBoxchart
chartworks.cb.beans.NFBubbleChart	netcharts.pro.charts.bubble.NFBubblechart
chartworks.cb.beans.NFComboChart	netcharts.pro.charts.combo.NFCombochart
chartworks.cb.beans.NFDiagram	netcharts.pro.charts.diagram.NFDiagramchart
chartworks.cb.beans.NFDialChart	netcharts.pro.charts.dial.NFDial
chartworks.cb.beans.NFGraph	netcharts.pro.common.NFGraph
chartworks.cb.beans.NFLineChart	netcharts.pro.charts.line.NFLinechart
chartworks.cb.beans.NFPiechart	netcharts.pro.charts.pie.NFPiechart
chartworks.cb.beans.NFRadarChart	netcharts.pro.charts.radar.NFRadarchart
chartworks.cb.beans.NFStockChart	netcharts.pro.charts.stock.NFStockchart
chartworks.cb.beans.NFStripChart	netcharts.pro.charts.strip.NFStripchart
chartworks.cb.beans.NFTimeChart	netcharts.pro.charts.time.NFTimechart
chartworks.cb.beans.NFXYChart	netcharts.pro.charts.xy.NFXYchart

## Constructor Changes

The mechanisms for creating new chart objects have changed. The following table shows the recommended migration path from each NCB chart creation mechanism to an equivalent NCPro mechanism.

NetCharts Beans Chart Creation	NetCharts Pro Chart Creation
NFBarchart()	NFBarchart()
NFBarchart(Applet parentApplet)	NFGraph.getGraphFromTemplate( File template, String documentBase)
NFBarchart(Applet parentApplet, Frame frame, NFCdf cdf)	NFGraph.getGraphFromTemplate( File template, String documentBase)
NFBarchart(File cdlFile)	NFGraph.getGraphFromTemplate(File cdlFile)
NFBarchart(Frame frame, NFCdf cdf)	NFGraph.getGraphFromTemplate(File cdlFile)
NFBarchart(String cdl)	NFGraph.getGraphFromTemplate(String cdl)
NFBarchart(URL cdlFile)	NFGraph.getGraphFromTemplate(URL cdlFile)
NFGraph.getGraphBean( Applet parentApplet, Frame f, NFCdf cdf)	NFGraph.getGraphFromTemplate( File template, String documentBase)
NFGraph.getGraphBean(File cdlFile)	NFGraph.getGraphFromTemplate(File cdlFile)
NFGraph.getGraphBean(String cdl)	NFGraph.getGraphFromTemplate(String cdl)
NFGraph.getGraphBean(URL cdlFile)	NFGraph.getGraphFromTemplate(URL cdlFile)

NCPro does not provide a mechanism for using `NFCdf` objects during chart creation. Direct use of `NFCdf` objects in NCPro is deprecated. Instead, the strings or input sources used to create `NFCdf` objects can be sent directly to equivalent NCPro chart constructors.

NCPro does not support the use of AWT Applets or Frames in any chart creation mechanism. For information on the recommended use of NCPro in applets, see the applet examples in the NCPro distribution.

## The set() API

The recommended mechanism for setting individual attributes on NCB charts was the six set() methods in the beans superclass, `netcharts.chart.NFChart`. These six methods have been exposed on the new NCPRO chart objects but they are deprecated. Programmers may take advantage of these methods in their initial migration, but it is strongly recommended that they plan to take advantage of the NCPRO data models and the many new object based setter methods available on each chart type. See *Programming with NetCharts Pro* for a discussion on how to fully exploit the new NCPRO API.

## Image Generation Changes

The mechanism for creating images and image maps have moved from methods on the chart objects to a separate class, `netcharts.pro.common.NFImageGeneration`

NetCharts Beans Image Generation	NetCharts Pro Image Generation
<code>chart.generateImage(mimeType, imageOutputStream )</code>	<pre>byte[] imageData = NFImageGeneration.generateImage(     chart,     NFImageGeneration.MIME_TYPE_PNG); imageOutputStream.write(imageData, 0, imageData.length);</pre>
<code>chart.getServerGeneratedImage( String mimeType, boolean createImageMap)</code>	<pre>NFImageGeneration.getServerGeneratedImage(     NFGraph graph,     String mimeType,     String textDescription,     boolean createImageMap)</pre>

The convenience methods provided for servlet writers to create and populate formatted HTML IMG and MAP tags moved from `chartworks.cb.util.DefaultPageCreator` to `netcharts.pro.util.NFServletUtil`. The servlet examples that use these convenience methods in NCB have been ported to NCPRO and can provide additional guidance for making a successful transition.