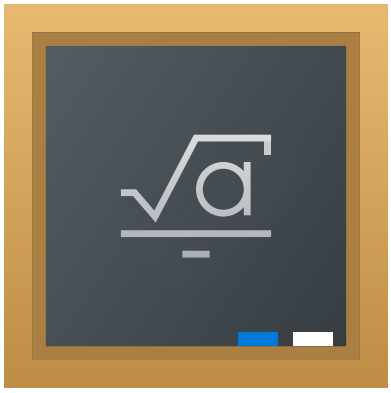
The Cantor Handbook

Alexander Rieder



Contents

1 Introduction **5**

2 Using Cantor **6**

2.1 Cantor features 6

2.2 The Cantor backends 7

2.3 The Cantor Workspace 8

2.3.1 The Worksheet 9

2.3.1.1 Settings 9

3 Command Reference **10**

3.1 The main Cantor window 10

3.1.1 The File Menu 10

3.1.2 The View Menu 11

3.1.3 The Worksheet Menu 11

3.1.4 The Settings Menu 11

3.1.5 The Help Menu 12

4 Developer's Guide to Cantor **13**

5 Questions and Answers **14**

6 Credits and License **15**

A Installation **16**

A.1 How to obtain Cantor 16

A.2 Requirements 16

A.3 Compilation and Installation 16

A.4 Configuration 16

Abstract

Cantor is an application, aimed at providing a nice worksheet view to other Free Software Math packages.

Chapter 1

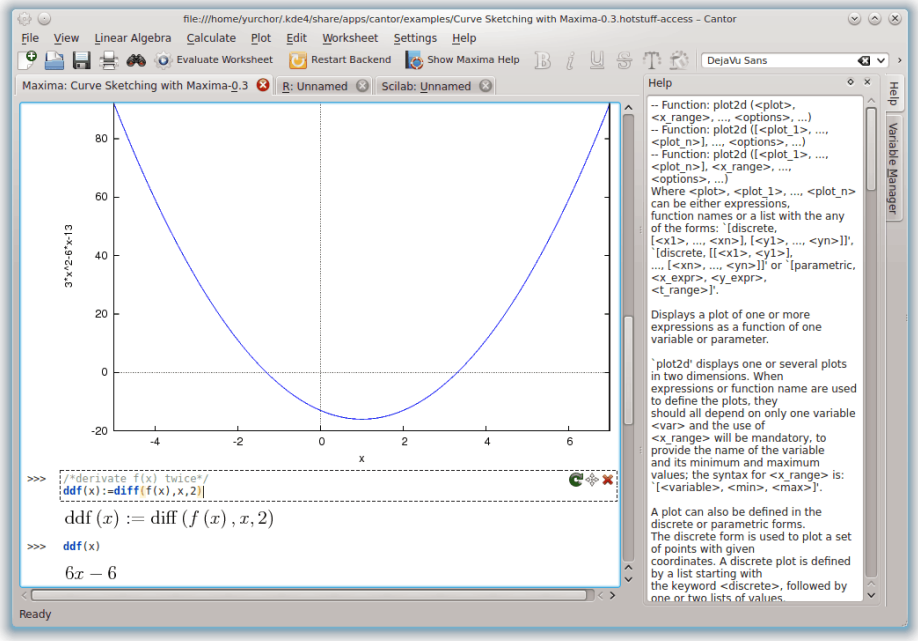
Introduction

Cantor is an application to allow you to you use your favorite mathematical applications from within an elegant KDE-integrated Worksheet Interface. It provides dialogs to assist with common tasks and allows you to share your worksheets with others.

Chapter 2

Using Cantor

Screenshot:



2.1 Cantor features

- Intuitive worksheet view for evaluating expressions.
- Results of graph plotting shown inside the worksheet.
- Easy plugin-based structure to allow different backends to be added.
- GetHotNewStuff integration to allow example worksheets to be downloaded (or uploaded.)
- Mathematical formula typesetting using LaTeX.
- Backend-aware syntax highlighting.
- Plugin-based assistant dialogs to help with common tasks, such as integrating a function, or entering a matrix.

2.2 The Cantor backends

Cantor offers you several choices for the backends you wish to use with it. The choice to make depends on what you want to achieve.

Currently the following backends are available:

Sage:

Sage is a free open-source mathematics software system licensed under the GPL. It combines the power of many existing open-source packages, within a common Python-based interface. See http://sagemath.org for more information.

Maxima:

Maxima is a system for the manipulation of symbolic and numeric expressions, including differentiation, integration, Taylor series, Laplace transforms, ordinary differential equations, systems of linear equations, polynomials, sets, lists, vectors, matrices, and tensors. Maxima yields high-precision numeric results by using exact fractions, arbitrary precision integers, and variable precision floating point numbers. Maxima can plot functions and data in two and three dimensions. See http://maxima.sourceforge.net for more information.

R:

R is a language and environment for statistical computing and graphics, similar to the S language and environment. It provides a wide variety of statistical (linear and nonlinear modelling, classical statistical tests, time-series analysis, classification, clustering, ...) and graphical techniques, and is highly extensible. The S language is often the vehicle of choice for research in statistical methodology, and R provides an open-source route to this. See http://www.r-project.org for more information.

KAlgebra:

KAlgebra is a MathML-based graph calculator, that ships with KDE Education project. See http://edu.kde.org/kalgebra/ for more information.

Qalculate!:

Qalculate! is not your regular software replication of the cheapest available calculator. Qalculate! aims to make full use of the superior interface, power and flexibility of modern computers. The center of attention in Qalculate! is the expression entry. Instead of entering each number in a mathematical expression separately, you can directly write the whole expression and later modify it. The interpretation of expressions is flexible and fault tolerant, and if you nevertheless do something wrong, Qalculate! will tell you so. Not fully solvable expressions are however not errors. Qalculate! will simplify as far as it can and answer with an expression. In addition to numbers and arithmetic operators, an expression may contain any combination of variables, units, and functions. See http://qalculate.sourceforge.net/ for more information.

Python2:

Python is a remarkably powerful dynamic programming language that is used in a wide variety of application domains. There are several Python packages to scientific programming.

Python is distributed under Python Software Foundation license (GPL compatible). See the official website for more information.

> NOTE
> This backend adds an additional item to the Cantor's main menu, **Package**. The only item of this menu is **Package** → **Import Package**. This item can be used to import Python packages to the worksheet.

> **WARNING**
> This backend supports Python 2 only.

Scilab:

Scilab is an free software, cross-platform numerical computational package and a high-level, numerically oriented programming language.

Scilab is distributed under CeCILL license (GPL compatible). See http://www.scilab.org/ for more information.

> **WARNING**
> You need Scilab version 5.5 or higher to be installed in your system to make this backend usable.

Octave:

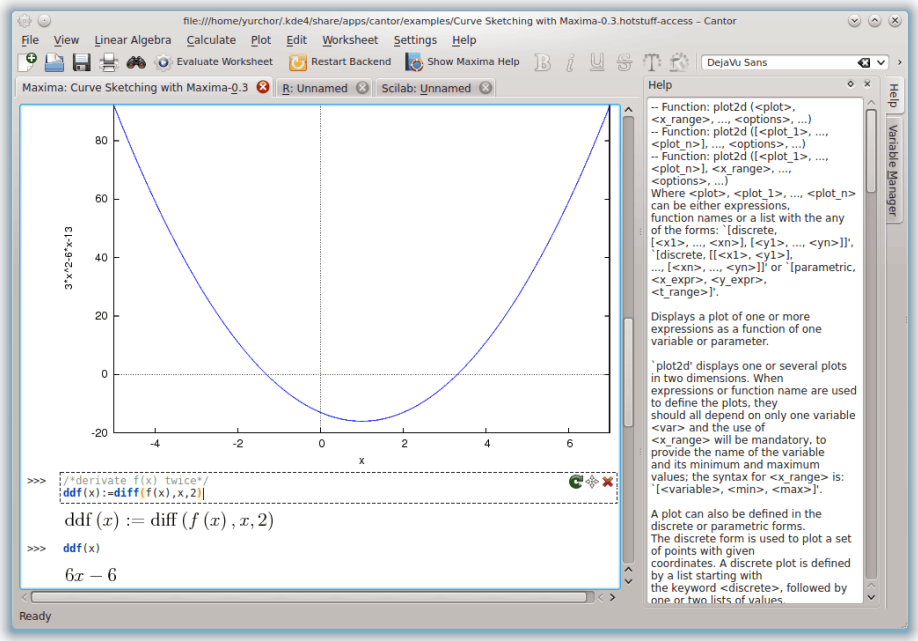
GNU Octave is a high-level language, primarily intended for numerical computations. It provides a convenient command line interface for solving linear and nonlinear problems numerically, and for performing other numerical experiments using a language that is mostly compatible with MATLAB. See http://www.gnu.org/software/octave/ for more information.

Lua:

Lua is a fast and lightweight scripting language, with a simple procedural syntax. There are several libraries in Lua aimed at math and science.
See http://www.lua.org/ for more information.

This backend supports luajit 2.

2.3 The Cantor Workspace



The Worksheet interface of Cantor consists of three parts:

1. The tab bar, where you can switch through the open worksheets.
2. A help panel, where help for a specific command is displayed if you enter ˝? command˝.
3. A Variable Manager panel where you can see the list of all available variables on the current worksheet. The Help and Variable Manager panels are shown as tabs on a sidebar of a Cantor window. Variable manager can be used for Maxima (if enabled on **Settings** → **Configure Cantor... Maxima** page), KAlgebra, Octave, Python2, Qalculate, and Scilab backends only.
4. The worksheet itself.

2.3.1 The Worksheet

The worksheet is the basic area for working with Cantor. Using it, you can enter expressions, allow the backend to perform calculations, and look at the results. The commands the worksheets accept depend on the chosen backend, so you should learn the appropriate backend syntax (the Show Help button takes you there directly). If you know the command, but you need additional information, you can type ˝? `command`˝, and contextual help will appear on the right of the worksheet. If you want to see some example worksheets, you can go to **Download Example Worksheets** in the **File** menu, and download worksheets other users have published.

To control each command entry of the worksheet you can use control buttons shown at the right part of the worksheet when hovering the entry field with a mouse cursor. There are three basic actions for each entry: **Evaluate Entry** to update the result of calculations for the entry, **Drag Entry** to move the entry at the other place of the current worksheet and **Remove Entry** to remove the entry from the current worksheet.

Font settings and alignment of worksheet entries can be changed using a toolbar above the worksheet.

2.3.1.1 Settings

The **Settings** menu allows you to change some settings of the appearance of your current worksheet.

Typesetting using L^AT_EX

This setting changes the way results are presented to you. If enabled, it passes the result through the L^AT_EX system, to produce visually appealing formulas. For example, '3*x^2*sqrt(2)*x+2/3' becomes $3\,x^2 + \sqrt{2}x + \frac{2}{3}$

Syntax Highlighting

Syntax Highlighting increases the readability of code, by highlighting keywords, or matching brackets in different colors.

Completion

When this is enabled, Cantor will show you possible completions for the command you are currently typing when you press the Tab-key. If there is only one choice remaining, it will insert the rest of the command for you.

Line Numbers

This setting controls whether the different expressions should have a number in front. This allows you to recall older results - in Maxima, for instance, this is done with the '%O1' command.

Animate Worksheet

This setting controls whether the different animations will be used to highlight changes on a worksheet.

Chapter 3

Command Reference

3.1 The main Cantor window

Cantor has the common KDE menu items, for more information read the Menu chapter of the KDE Fundamentals.

Additional or special menu items in Cantor:

3.1.1 The File Menu

File → New (Ctrl+N)

Creates a new worksheet with the default backend.

File → New → *Backend*

Creates a new worksheet with the chosen backend.

File → Download Example Worksheets

Opens the dialog to download examples uploaded by other Cantor users with GetHotNewStuff.

File → Open Example

Opens the dialog to choose an example to load. The examples should be downloaded with **File → Download Example Worksheets** beforehand.

File → Save Plain Text

Allows saving the current worksheet as a plain text document.

File → Export to LaTeX

Allows saving the current worksheet as a $\LaTeX$ document.

File → Publish Worksheet

Opens the dialog to publish your current worksheet and make it available to other Cantor users through GetHotNewStuff.

File → Run Script

Allows to load and run the batch scripts for the current backend.

3.1.2 The View Menu

View → Zoom In (Ctrl++)
Zoom in the current worksheet by 10%.

View → Zoom Out (Ctrl+-)
Zoom out the current worksheet by 10%.

View → Show Script Editor
Opens the batch script editor window for the current backend.

View → Panels → Help
Toggles the visibility of the **Help** tab in the sidebar.

View → Panels → Variable Manager
Toggles the visibility of the **Variable Manager** tab in the sidebar.

3.1.3 The Worksheet Menu

Worksheet → Evaluate Worksheet
Triggers the evaluation of the whole worksheet.

Worksheet → Evaluate Entry (Shift+Return)
Triggers the current entry evaluation.

Worksheet → Insert Command Entry (Ctrl+Return)
Inserts a new command entry below the current cursor position.

Worksheet → Insert Text Entry
Inserts a new text entry below the current cursor position.

Worksheet → Insert Image
Inserts the image template below the current cursor position. You will need to click with right mouse button at the template to configure the image details.

Worksheet → Insert Page Break
Inserts the page break below the current cursor position.

Worksheet → Remove current Entry (Shift+Del)
Removes current entry from the worksheet.

3.1.4 The Settings Menu

This menu contains standard KDE **Settings** items, as well as the items described in Settings section.

3.1.5 The Help Menu

Help → Cantor Handbook (F1)

Invokes the KDE Help system starting at the Cantor help pages. (this document).

Help → What's This? (Shift+F1)

Changes the mouse cursor to a combination arrow and question mark. Clicking on items within Cantor will open a help window (if one exists for the particular item) explaining the item's function.

Help → Report Bug...

Opens the Bug report dialog where you can report a bug or request a 'wishlist' feature.

Help → Switch Application Language...

Opens a dialog where you can edit the **Primary language** and **Fallback language** for this application.

Help → About Cantor

This will display version and author information.

Help → About KDE

This displays the KDE version and other basic information.

Chapter 4

Developer's Guide to Cantor

Cantor is easily extendable by plugins. You can write

Backends:
A new backend enables you to use the Cantor interface with a different program.

Assistants:
Assistants provide dialogs to allow common tasks to be automated (such entering a matrix or running a script.)

At present, there are no special developer tutorials or API documentation, so if you want to write a plugin you will have to read through the sources. All interface definitions are in the `lib` directory. For some example implementations, look at the plugins shipped with Cantor.

Chapter 5

Questions and Answers

This document may have been updated since your installation. You can find the latest version at http://docs.kde.org/ .

1. *The Menu Item **Typeset using LaTeX** is not available*

   Not every backend supports every option. Current version supports LaTeX output for Maxima, Qalculate! and Sage backends.

2. *I have installed Sage/Maxima, but the appropriate backend does not appear.*

   One possible explanation is that Cantor did not correctly detect the location. Go to the **Settings** → **Configure Cantor...** dialog and change the **Path** to the appropriate application.

3. *What a peculiar name - what does it have to do with mathematics?*

   Cantor is named after the German mathematician Georg Cantor, the creator of set theory.

Chapter 6

Credits and License

Cantor

Program copyright 2009-2011 Alexander Rieder alexanderrieder@gmail.com

Documentation Copyright (c) 2009-2011 Alexander Rieder alexanderrieder@gmail.com

This documentation is licensed under the terms of the GNU Free Documentation License.
This program is licensed under the terms of the GNU General Public License.

Appendix A

Installation

A.1 How to obtain Cantor

Cantor is part of the KDE project http://www.kde.org/ .

Cantor can be found in the kdeedu package on ftp://ftp.kde.org/pub/kde/ , the main FTP site of the KDE project.

A.2 Requirements

In order to successfully use Cantor, you need KDE 4. If you want support for type-setting to be enabled, you need to install LaTeX and LibSpectre. To use the different Backends, the appropriate math package is required.

A.3 Compilation and Installation

For detailed information on how to compile and install KDE applications see Building and Running KDE Software From Source

Since KDE uses **cmake** you should have no trouble compiling it. Should you run into problems please report them to the KDE mailing lists.

A.4 Configuration

Depending on your system, you may need to adjust the paths to the different applications used by Cantor.