

Software Libre para enseñar o aprender Matemática Porqué y Cómo

Pablo De Nápoli

pdenapo@dm.uba.ar

Introducción

- En esta charla, presentaremos algunos de los programas que son **software libre**, que pueden utilizarse para docencia e investigación en matemática.
- Discutiremos también cuáles son las ventajas de utilizar software libre para la docencia y la investigación en matemática.

¿Por qué ? : ¿Qué ventajas tiene utilizar software libre?

¿Qué es el Software Libre?

Según la definición propuesta por **Richard Stallman** un programa es software libre si el usuario tiene (de acuerdo a la **licencia** del programa) las siguientes libertades o derechos:

- 0) La libertad de **ejecutar el programa**, con cualquier propósito.
- 1) La libertad de **estudiar cómo funciona el programa**, y **adaptarlo** a sus necesidades.
- 2) La libertad de **distribuir copias**, con lo que puede ayudar a su vecino.
- 3) La libertad de mejorar el programa y **hacer públicas las mejoras** a los demás, de modo que toda la comunidad se beneficie.

Una cuestión de Derechos

- El software libre es una **cuestión de derechos**, no una cuestión de tecnología.
- Lo que decide si un programa es o no libre, es la **licencia** bajo la cual el programa se distribuye.

Una licencia de software es un **contrato** entre el titular del copyright (derecho de copia) sobre un software, y el usuario, que establece que cosas el usuario puede hacer con el programa (y cuales no).

- Software libre (free software) \neq software gratuito (freeware).

“El software libre es una cuestión de libertad, no de precio”.

Escapando al dilema moral...

Una ventaja de usar software libre en la docencia, es que podemos distribuir copias del programa **legalmente** a los alumnos.

Esto permite que los alumnos puedan utilizar el programa en sus casas.

La licencia del programa nos autoriza a hacerlo.

“Si usas un programa sin la libertad de distribuir copias, pronto te enfrentarás a un **dilema moral** cuando un amigo te diga: Ese programa que tienes me sería útil. ¿Podrías darme una copia?” (Richard Stallman)

Accediendo al conocimiento...

Pero quizás la ventaja más importante de utilizar software libre en la docencia o la investigación matemática, es que permite **acceder al conocimiento** que hay detrás del software.

Utilizando software libre, nuestros alumnos pueden, por ejemplo, ir y ver que algoritmo utiliza el programa para realizar determinado cálculo.

O incluso pueden tomar el **código fuente** en sus manos y mejorarlo, o adaptarlo para hacer algo diferente.

Enseñando valores...

Si partimos del convencimiento de que la educación no tiene por objeto exclusivamente transmitir una serie de conocimientos técnicos o prácticos, sino que busca fundamentalmente **transmitir valores** socialmente positivos, resulta claro que la utilización de software libre puede contribuir a este propósito:

“Un programa que sea software libre puede no ser técnicamente superior, pero siempre será éticamente superior” (Richard Stallman)

- Compartir es bueno.
- No al software “trucho”: debemos enseñar a respetar la ley.

Una cita de Linus Torvalds

“I think, fundamentally, open source does tend to be more stable software. It’s the right way to do things. I compare it to **science vs. witchcraft**. In science, the whole system builds on people looking at other people results and building on top of them. In witchcraft, somebody had a small secret and guarded it but never allowed others to really understand it and build on it.

Traditional software is like witchcraft. In history, witchcraft just died out. The same will happen in software. When problems get serious enough, you can’t have one person or one company guarding their secrets. You have to have everybody share in knowledge.”

S. L. y Método Científico

- Existe un estrecho paralelo entre la forma que el software libre se desarrolla y el **método científico**.
- En la ciencia es fundamental, la crítica de los resultados por parte de otros científicos (**revisión por parte de los pares**); y su utilización por parte de otros para obtener nuevos resultados.
- **Exactamente lo mismo sucede** en el mundo del software libre. Uno puede tomar el código de otro programador, examinarlo, mejorarlo, corregirlo, e incluso usarlo como base para desarrollar un nuevo programa.
- Las licencias de software libre permiten hacer esto **sin tener que pedir permiso** al autor original del programa.



**¿Cómo?: ¿Qué alternativas libres
existen?**

Alternativas Libres

- En esta segunda parte de la charla presentaremos una selección de los programas existentes, que tiene el propósito de convencerlos de que existen **alternativas libres** a otros programas privativos que quizás son más populares.
- Todos los programas que vamos a presentar, se pueden utilizar sobre el sistema operativo **GNU/Linux**, pero algunos de ellos funcionan sobre otros sistemas operativos.
- El mundo del software libre es **un mundo de alternativas**. En general no suele existir un sólo programa para una tarea, sino varias alternativas para elegir.

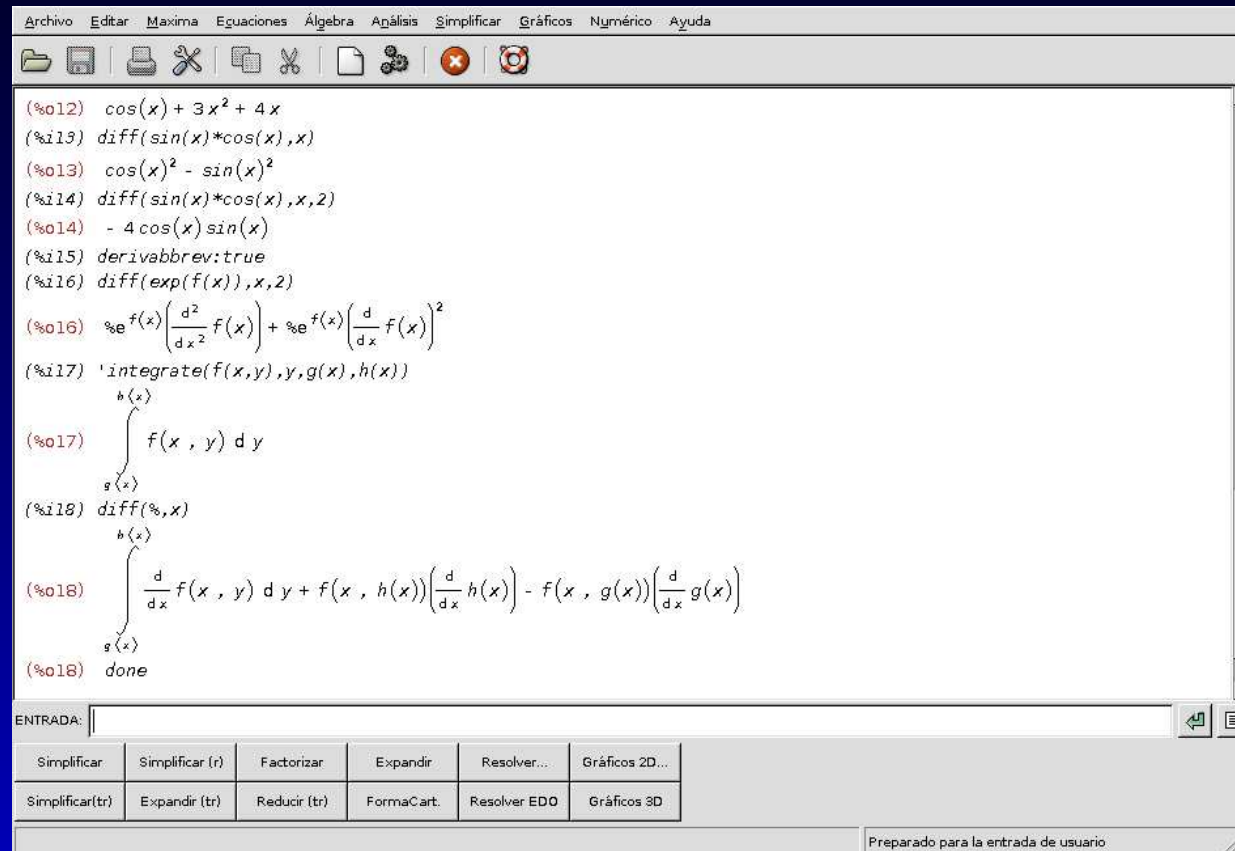


Sistemas de Álgebra Computacional

Álgebra Computacional

- Los **sistemas de álgebra computacional** son programas capaces de realizar **operaciones simbólicas** como diferenciación, integración, factorización de polinomios, etc.
- Soportan aritmética de precisión arbitraria.
Por ejemplo: Axiom, Maxima, Sage, Yacas, etc.
- Algunos son para aplicaciones específicas:
Pari/Gp (teoría de números), Gap (teoría de grupos), Singular y Macaulay2 (para geometría algebraica).

wxMaxima



The screenshot shows the wxMaxima interface with a menu bar (Archivo, Editar, Maxima, Ecuaciones, Álgebra, Análisis, Simplificar, Gráficos, Numérico, Ayuda) and a toolbar. The main window contains a list of operations and their results:

```
(%o12)  $\cos(x) + 3x^2 + 4x$   
(%i19)  $\text{diff}(\sin(x) \cdot \cos(x), x)$   
(%o13)  $\cos(x)^2 - \sin(x)^2$   
(%i14)  $\text{diff}(\sin(x) \cdot \cos(x), x, 2)$   
(%o14)  $-4 \cos(x) \sin(x)$   
(%i15)  $\text{derivabbrev:true}$   
(%i16)  $\text{diff}(\exp(f(x)), x, 2)$   
(%o16)  $e^{f(x)} \left( \frac{d^2}{dx^2} f(x) \right) + e^{f(x)} \left( \frac{d}{dx} f(x) \right)^2$   
(%i17)  $\text{'integrate}(f(x,y), y, g(x), h(x))$   
(%o17)  $\int_{g(x)}^{h(x)} f(x, y) dy$   
(%i18)  $\text{diff}(\%, x)$   
(%o18)  $\int_{g(x)}^{h(x)} \frac{d}{dx} f(x, y) dy + f(x, h(x)) \left( \frac{d}{dx} h(x) \right) - f(x, g(x)) \left( \frac{d}{dx} g(x) \right)$   
(%o18) done
```

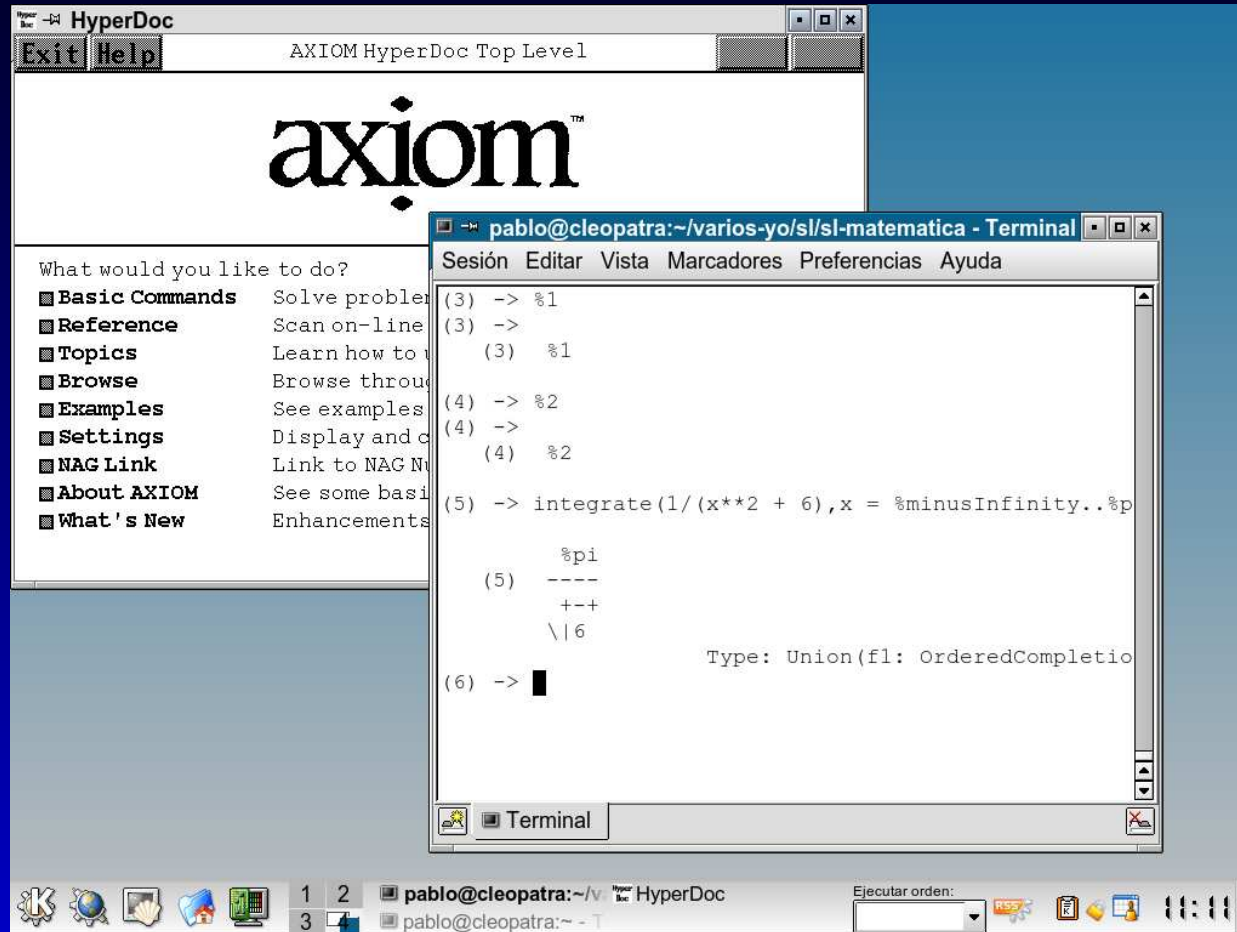
Below the main window is an input field labeled "ENTRADA:" and a toolbar with buttons: Simplificar, Simplificar (r), Factorizar, Expandir, Resolver..., Gráficos 2D..., Simplificar(tr), Expandir (tr), Reducir (tr), FormaCart., Resolver EDO, Gráficos 3D. At the bottom right, it says "Preparado para la entrada de usuario".

wxMaxima: ejemplos de diferenciación e integración

características de Maxima

- Es un sistema para la **manipulación de expresiones simbólicas y numéricas**, incluyendo diferenciación, integración, expansión en series de Taylor, transformadas de Laplace, ecuaciones diferenciales ordinarias, sistemas de ecuaciones lineales, y vectores, matrices y tensores.
- Maxima es un descendiente de Macsyma, el legendario sistema de álgebra computacional desarrollado a finales de 1960 en el MIT.
- Licencia: **GNU GPL** (versión 2).
- Lenguaje de programación: Lisp.

Axiom

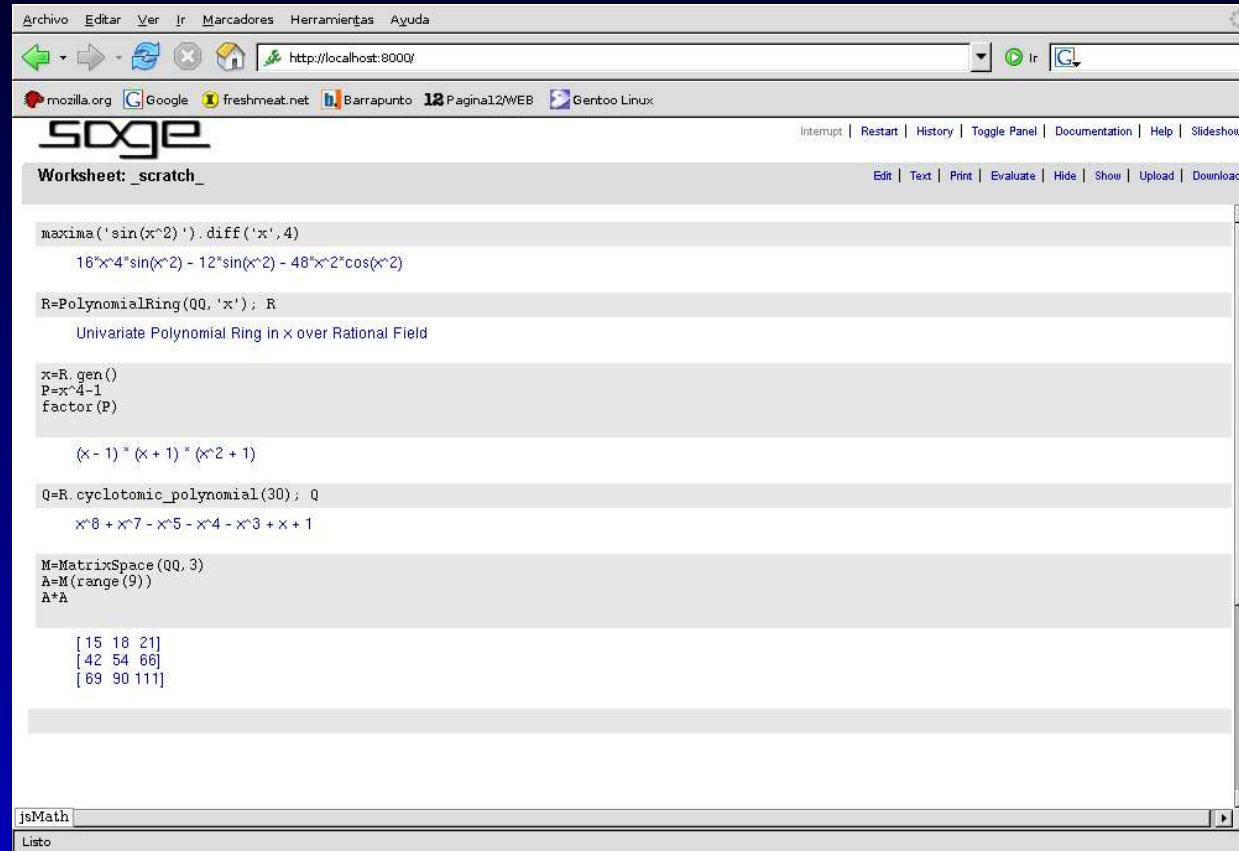


Axiom funcionando bajo KDE

Características de Axiom

- Es un sistema de álgebra computacional, con **capacidades simbólicas**.
- Está en desarrollo desde 1973. Originalmente Axiom fue desarrollado por investigadores de IBM bajo el nombre de **Scratchpad**.
- Lenguajes de programación: Lisp, Spad, (Aldor).
- Licencia: **BSD modificada**.
- Posee una **jerarquía de tipos de datos** muy elaborada, que representa muy bien la manera de pensar de los matemáticos (tiene “categorías” como anillo o grupo).
- Viene con abundante documentación.

SAGE



The screenshot shows a web browser window with the SageMath interface. The browser's address bar shows 'http://localhost:8000/'. The SageMath logo is visible at the top left, and the worksheet title is 'Worksheet: _scratch_'. The interface includes a menu bar with options like 'Archivo', 'Editar', 'Ver', 'Ir', 'Marcadores', 'Herramientas', and 'Ayuda'. Below the menu bar, there are navigation icons and a search bar. The main content area displays the following SageMath code and its output:

```
maxima('sin(x^2)') . diff('x', 4)
16*x^4*sin(x^2) - 12*sin(x^2) - 48*x^2*cos(x^2)

R=PolynomialRing(QQ, 'x'); R
Univariate Polynomial Ring in x over Rational Field

x=R.gen()
P=x^4-1
factor(P)
(x - 1) * (x + 1) * (x^2 + 1)

Q=R.cyclotomic_polynomial(30); Q
x^8 + x^7 - x^5 - x^4 - x^3 + x + 1

M=MatrixSpace(QQ, 3)
A=M(range(9))
A*A
[15 18 21]
[42 54 66]
[69 90 111]
```

At the bottom of the browser window, the status bar shows 'jsMath' and 'Listo'.

SageNotebook funcionando con Mozilla Firefox

Características de SAGE

- Es un sistema de álgebra computacional escrito en Python.
- Provee una interfase de alto nivel que integra a otros programas y librerías pre-existentes como:
 - Pari/GP, NTL, Nwrank (teoría de números)
 - GNU GSL, Scipython, Numpy (aplicaciones numéricas)
 - Linbox (álgebra lineal numérica), Cvxopt (optimización)
 - Gap (teoría de grupos)
 - Singular (álgebra conmutativa y geometría algebraica)
 - Maxima, Sympy (manipulaciones simbólicas)

Más sobre SAGE ...

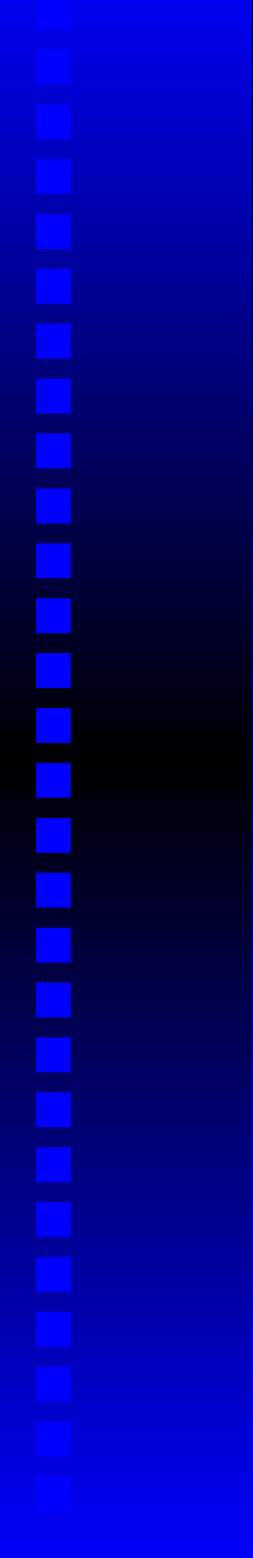
- **SAGE** significa “Software for Algebra and Geometry Experimentation”
- El desarrollo de SAGE es liderado por William Stein, de la Universidad de Washington.
- Licencia: **GNU GPL**, versión 2.
- El hecho de ser software libre hace que no sea necesario reinventar la rueda cada vez, sino que se pueda aprovechar paquetes pre-existentes.
- SAGE puede utilizarse desde un navegador web (**Mozilla Firefox**) (SageNotebook).

¿y porqué utilizar Python?

- **Python** es un lenguaje de programación de muy alto nivel, moderno y fácil de entender.
- Es ampliamente utilizado para distintas aplicaciones.
- Python **es fácil de aprender**. Lo recomiendo para enseñar a programar.

Entonces ...

- El código de SAGE es **fácil de entender y modificar**.
- No es necesario aprender un lenguaje específico para utilizar SAGE.
- **Python** es una herramienta eficaz para integrar otros paquetes y librerías pre-existentes.

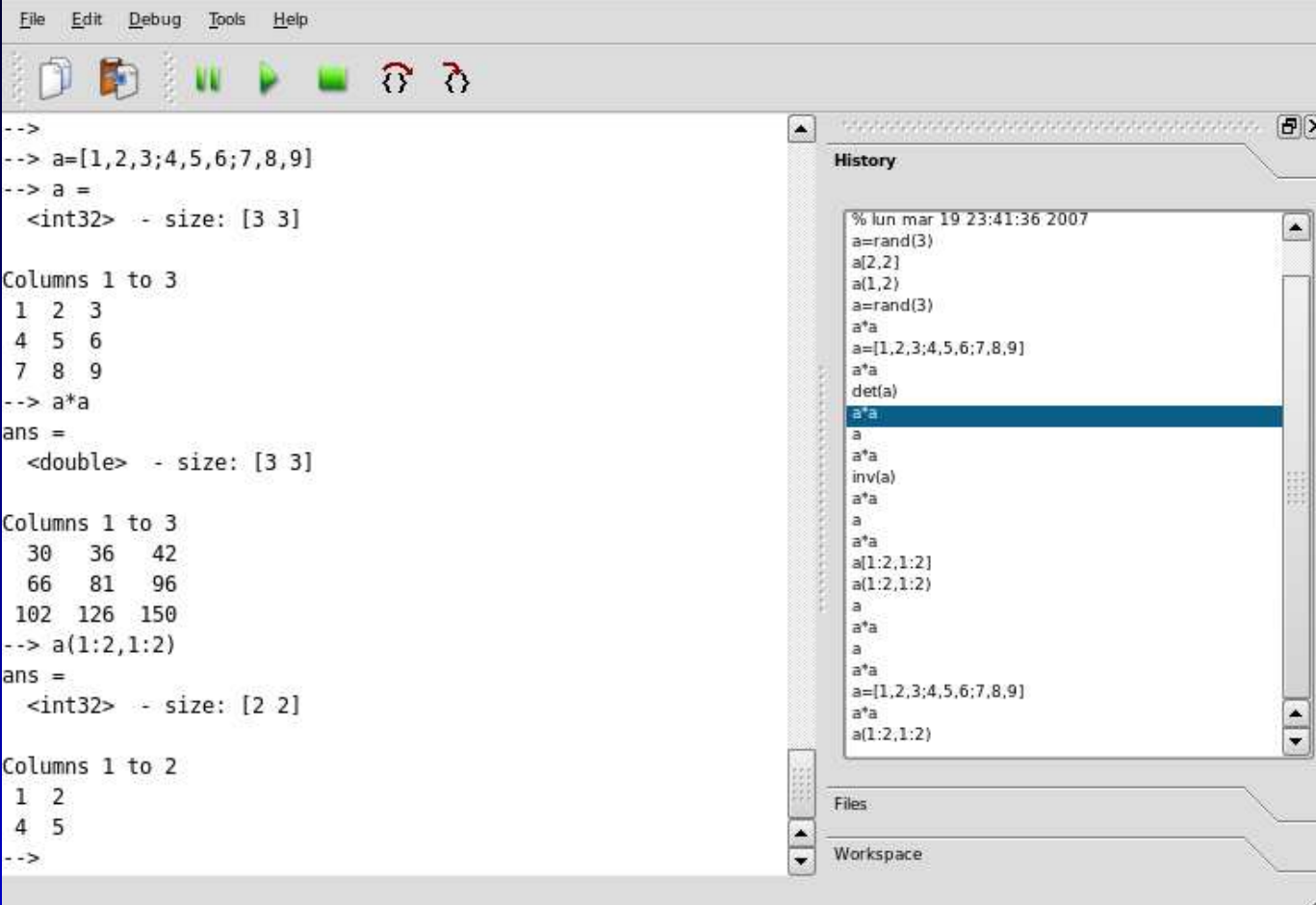


Programas Orientados al Cálculo Numérico

Cálculo Numérico

- Otra categoría de programas que podemos mencionar son los orientados al **cálculo numérico** y las aplicaciones a la ingeniería.
- Clones de Matlab:
 - GNU Octave
 - FreeMat
 - Scilab (No es software completamente libre porque su licencia no permite redistribuir versiones modificadas con fines comerciales).
 - Euler (No es realmente un clon de Matlab pero la idea es similar)

FreeMat



```
File Edit Debug Tools Help
-->
--> a=[1,2,3;4,5,6;7,8,9]
--> a =
<int32> - size: [3 3]

Columns 1 to 3
1 2 3
4 5 6
7 8 9
--> a*a
ans =
<double> - size: [3 3]

Columns 1 to 3
30 36 42
66 81 96
102 126 150
--> a(1:2,1:2)
ans =
<int32> - size: [2 2]

Columns 1 to 2
1 2
4 5
-->
```

History

```
% lun mar 19 23:41:36 2007
a=rand(3)
a[2,2]
a(1,2)
a=rand(3)
a*a
a=[1,2,3;4,5,6;7,8,9]
a*a
det(a)
a*a
a
a*a
inv(a)
a*a
a
a*a
a[1:2,1:2]
a(1:2,1:2)
a
a*a
a
a*a
a=[1,2,3;4,5,6;7,8,9]
a*a
a(1:2,1:2)
```

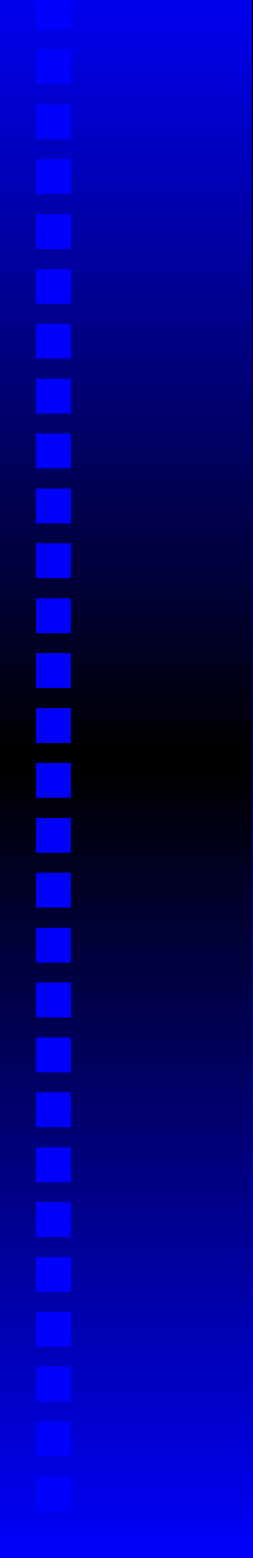
Files

Workspace

FreeMat mostrando algunas operaciones con Matrices

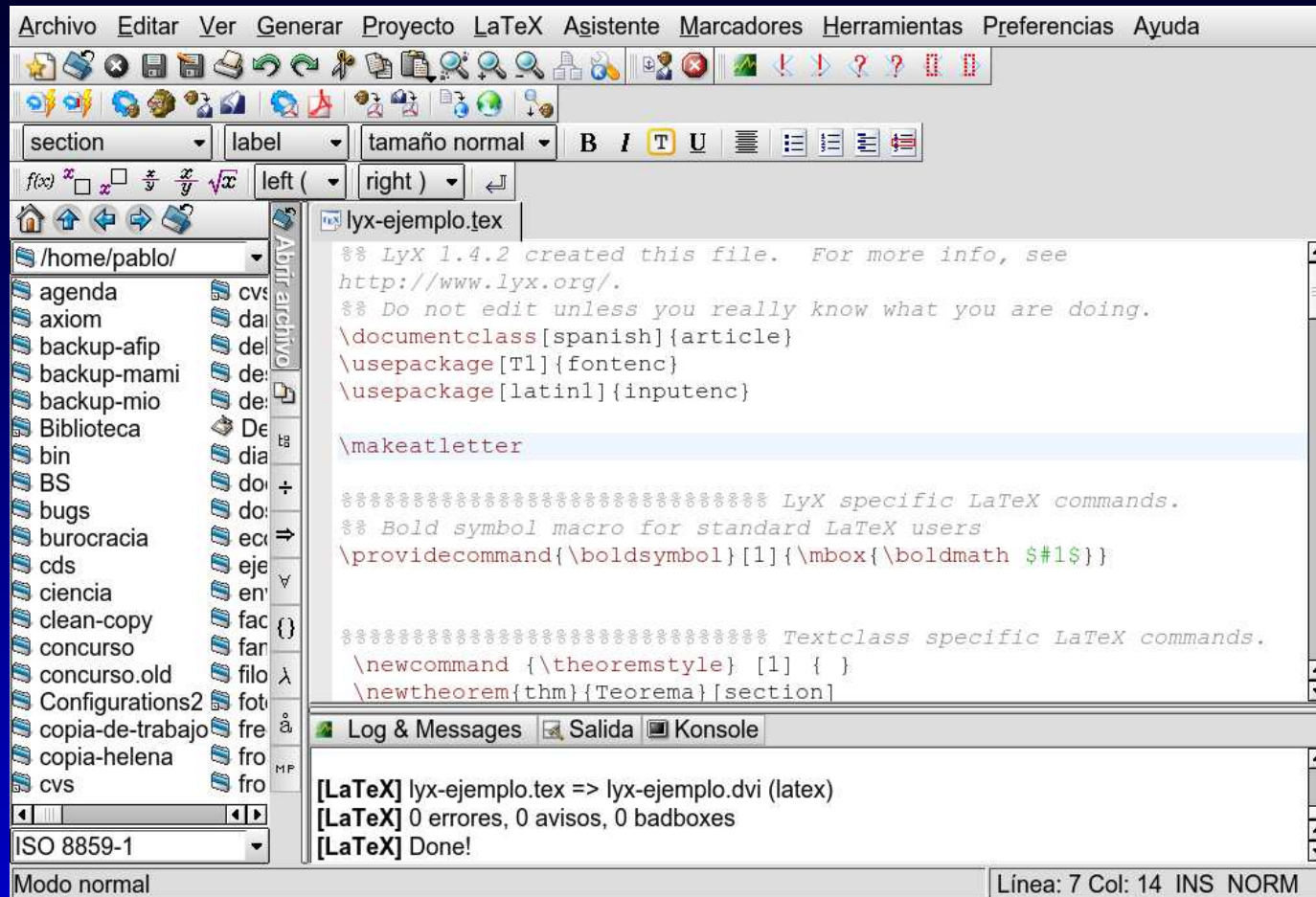
Otras alternativas

- El paquete **Scipy** (integrado en **SAGE**) provee funcionalidades para cálculo numérico (similares a las de Matlab), pero en **Python**.
- **GNU R** : Un programa para estadística. (similar al S)



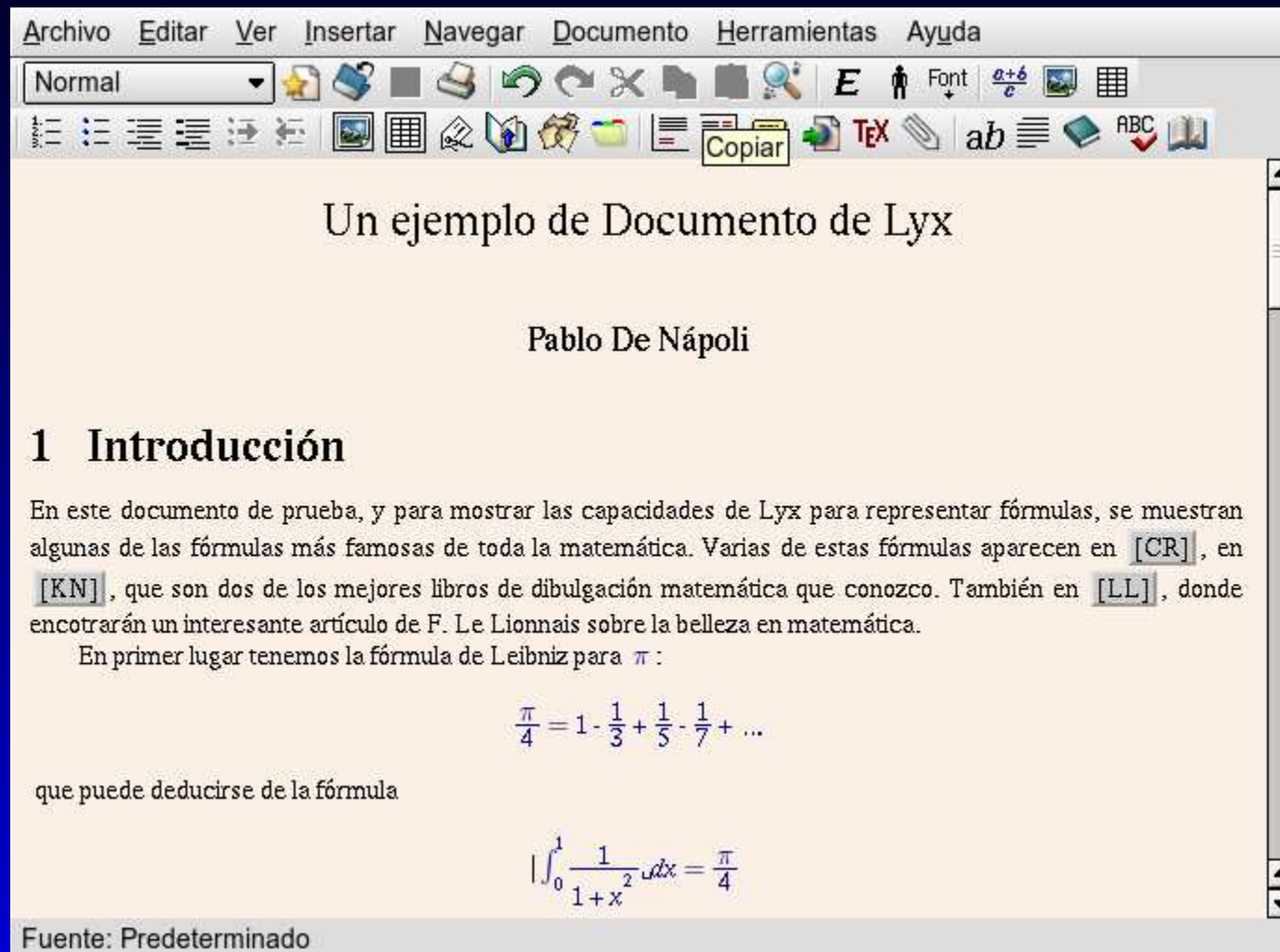
Programas para escribir matemática

Kile



Es un entorno de desarrollo integrado (IDE) para escribir en $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$.

Lyx



“Lyx: en tu pantalla se ve como cualquier procesador de texto, en el papel como ningún otro”



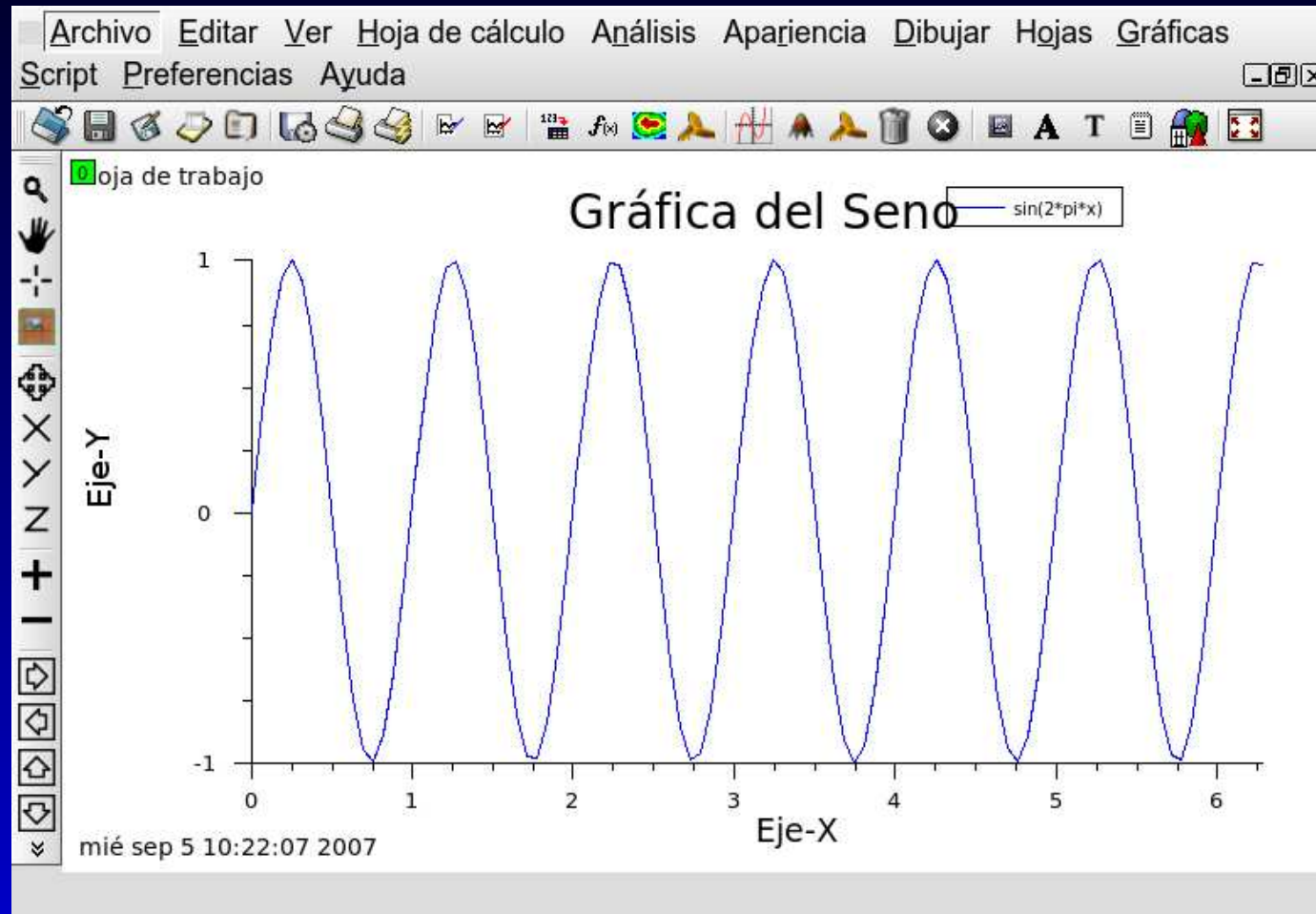
Graficadores

Graficadores

Existen programas específicamente destinados a graficar funciones y datos, como por ejemplo:

- Gnuplot (¡que no es parte del proyecto GNU!)
- Labplot
- Paraview
- Scigraphica, Qtiplot, Grace, ...

Labplot

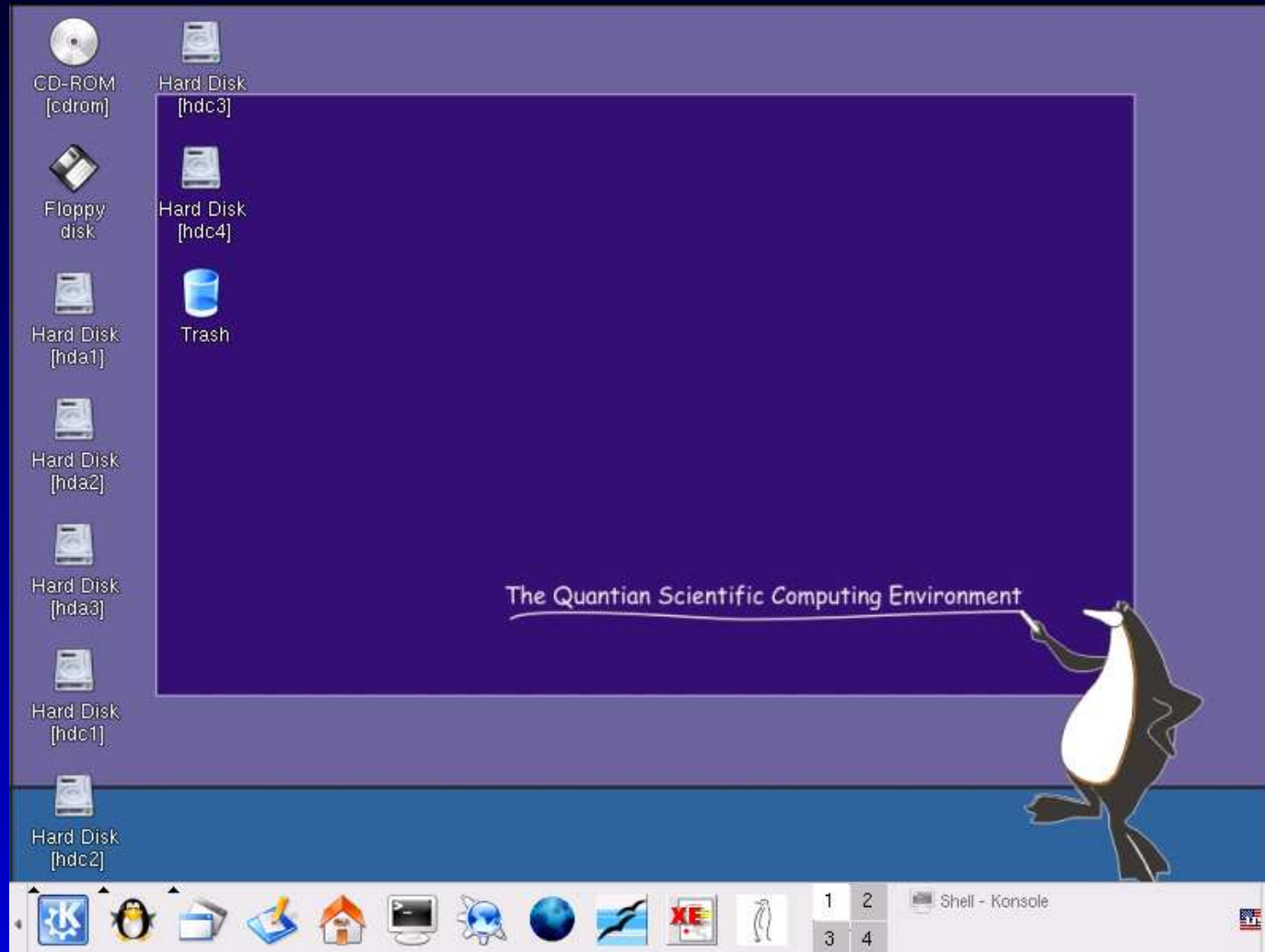


Labplot es un graficador de funciones y datos para KDE.



Distribuciones específicas de GNU/Linux

Quantian



Quantian es una distribución de GNU/Linux específica para computación científica.

Características de Quantian

- Es una distribución en Live-CD/ Live-DVD derivada de Knoppix.
- Incluye casi todos los programas mencionados en esta charla.
- Tiene soporte para procesamiento paralelo (OpenMosix).

Más referencias...

Pueden encontrar enlaces a los programas mencionados en esta charla (y otros no mencionados) en mi página personal:

[http://mate.dm.uba.ar
/~pdenapo/mathsoft.html](http://mate.dm.uba.ar/~pdenapo/mathsoft.html)

Sobre la filosofía del movimiento de software libre, la referencia obligada es el sitio del Proyecto GNU (Free Software Foundation)

<http://www.gnu.org>

Licencia de Este Documento

Copyright 2007-8 Pablo Luis De Nápoli

Se garantiza el permiso para copiar, distribuir y/o modificar este documento bajo los términos de la Licencia de Documentación Libre GNU, versión 1.2 (GNU Free Documentation License, Version 1.2) o cualquier otra versión posterior publicada por la Free Software Foundation; este documento se presenta sin Secciones Invariables (no Invariant Sections), sin Textos de Tapa (no Front-Cover Texts) y sin Textos de Contratapa (no Back-Cover Texts).