Todolist for PGFPlots

`1.9-3-gdda32d2`

Christian Feuersänger

October 20, 2013

Contents

1 Release TODO

The following steps have to be done for every release:

- run tests against pgf CVS

- run tests against latest pgf stable

- finish manual

  - fix warnings
  - browse through it

- update ChangeLog with "RELEASED VERSION XXX" and update README

- assign git tag for new version

- ran `scripts/pgfplots/pgfplotsrevisionfile.sh` to assign the package versions based on tag

- recompile manual (to pick up the correct version)

- run primitive example files

- run context tests (basically compile and browse-through. There are no assertions.)
  - run against pgf CVS
  - run against latest pgf stable
- adjust the release makefile to get correct file names
- assemble TDS.zip and CTAN.zip (using the release makefile)
  - `make -f pgfplots/scripts/pgfplots/Makefile.pgfplots_release_sourceforge` this creates the release files
  - make sure the archives do not contain wrong files.
- upload to CTAN.

```
Submitted to host
dante.ctan.org
Your name and email
    Christian Feuersnger <cfeuersaenger@users.sourceforge.net>
Filename
    pgfplots_1.6.1.ctan.flatdir.zip
Version number
    1.6.1
Location on CTAN
    /graphics/pgf/contrib/pgfplots/
Summary description
    pgfplots - Create normal/logarithmic plots in two and three dimensions for LaTeX.
License type
    gpl
Announcement text
...
Notes to maintainers
    As requested, I created a flat directory structure containing all files (i.e. it
```

- release to sourceforge. Make sure to update the README at top-level.
- upload to sourceforge web space (using the release makefile)
  - `make -f pgfplots/scripts/pgfplots/Makefile.pgfplots_release_sourceforge uploaddis` this copies the manuals
  - `make -f pgfplots/scripts/pgfplots/Makefile.pgfplots_release_sourceforge upload` this updates the unstable
  - if necessary, update `scp://cfeuersaenger,pgfplots@web.sourceforge.net/htdocs/index.php`
- send announcement to `pgfplots-features@lists.sourceforge.net`

2   Tests

last test verifications:

	pgf CVS	pgf 2.10	pgf 2.00	pgf 2.00+compat=default
regressiontests	for 1.9	for 1.9	for 1.8 (7% fail)	2009-12-30
unittests	for 1.9	for 1.9		
manual	for 1.9	for 1.9	for 1.5	
pgfplotstable.pdf	for 1.9	for 1.9	for 1.5	
example latex	for 1.9	for 1.9	2009-12-30	
example context	for 1.9	for 1.9	2009-12-30	
example plain tex	for 1.9	for 1.9	2009-12-30	
tests context	for 1.9	for 1.9		

3   Documentation todo

pgfplotstodo.tex:149   **Documentation Todo**   [open, Priority 5, Epic '']

document installation requirements when using lualatex (LUAINPUTS should contain pgfplots install dir)

pgfplotstodo.tex:238   **Documentation Todo**   [open, Priority 5, Epic '']

bei dem Bsp-Tex zu pgfplotstable scheint eine Zeile im Tex-File zu fehlen: `\usepackage{pgfplotstable}`

Auerdem wre es zum Einstieg fr das aus der Datei lesen schn, wenn es zu den Daten auch ein kurzes Beispiel-File fr einen Plot gbe.

pgfplotstodo.tex:495   **Documentation Todo**   [open, Priority 5, Epic '']

try a bar plot with individually shaded bars

FIXME : collect details

pgfplotstodo.tex:511   **Documentation Todo**   [open, Priority 5, Epic '']

contour: a change label dist

pgfplotstodo.tex:544   **Documentation Todo**   [open, Priority 5, Epic '']

document 'execute at begin axis' and its new variants

pgfplotstodo.tex:549   **Documentation Todo**   [open, Priority 5, Epic '']

document how to plot against the coordindex

Documentation Todo   [open, Priority 5, Epic '']

document how to identify the source of "dimension too large" errors: tracingstuff.

Documentation Todo   [open, Priority 5, Epic '']

It seems as if the AMS command `$\text{\ref{ref:to:a:plot}}$` instantiates the `\ref` at least four times. Document somehow that it is better to use '`\hbox`' instead

Documentation Todo   [open, Priority 5, Epic '']

clickable lib: I have the impression that acroread fires warnings only for the manual - not always when the clickable lib is used. Why!?

Documentation Todo   [closed, Priority 5, Epic '']

layers:

1. ✓motivation and use-cases
2. ✓simple example
3. ✓multi-axis discussion
4. ✓tikz integration
5. ✓explain how to merge custom layers and pgfplots layers (and say that pgfplots overwrites layers of tikz)
6. ✓specialties: defining own layer sets
7. ✓limitations: show list of supported anchors and explain implications of cell picture

the clipping of marker paths should always be active - but at least for layered graphics. It also needs a better UI

TODO:

- implement 'clip mode=individual' for axis paths

  FIXME : is there are good reason why `clip mode=global` is a bad choice for the default!?

  Perhaps the layered graphics feature can be shipped in a first version – with `clip mode=global`. It is simpler anyway.

- ✓document 'mark layer'

Documentation Todo   [closed, Priority 5, Epic '']

document benefits of using lualatex (memory limits)

Documentation Todo   [closed, Priority 5, Epic '']

pgfplotstable: document that

```
\pgfplotstabletypeset[
  typeset cell/.append code={%
\ifnum\pgfplotstablerow<0
\pgfkeyssetvalue{/pgfplots/table/@cell content}{}%
\fi
  },
  outfile={table},
  header=false,
  columns/0/.style={string type,column type=r},
  columns/1/.style={string type,column type=l},
  columns/2/.style={string type,column type=l}
  ]
```

can be used to eliminate the displayed header line.

Documentation Todo   [closed, Priority 5, Epic '']

|smithchartmirrored— is undocumented! see `https://sourceforge.net/tracker/?func=detail&atid=1060657&aid=3486928&group_id=224188`

Documentation Todo   [closed, Priority 5, Epic '']

document that `axis lines=none` is essentially an alias for `hide axis` .

Documentation Todo   [closed, Priority 5, Epic '']

Document how to use decorations inside of plots

```
\begin{tikzpicture}[]

  \begin{axis}[axis lines=middle,
      xmin=-2,
      xmax=2,
      ymin=-2,
      ymax=2,
      xtick={-1,1},
      ytick={-1,1},
      yticklabel=\ ,% this disables the standard tick label *text* (but not the line)
```

```
        extra description/.code={
            % this generates custom y labels to implement individual
            % styles for every tick:
            \node[below left] at (axis cs:0,-1) {$-1$};
            \node[above left] at (axis cs:0,1) {$1$};
        },
        axis line style={->},
    ]%,x=1cm,y=1cm]
    \addplot[samples=100,domain=0:2*pi,
        % tedious, but necessary: pgfplots accidentally resets the
        % "decorate" option at the beginning of the path (probably a
        % bug).
        % This is a work-around:
        every path/.style={
            postaction={decorate},
            every path/.style={},
        },
        decoration={markings,
                mark=at position 0.25 with {\arrow{>}},
                mark=at position 0.5 with {\arrow{>}},
                mark=at position 0.75 with {\arrow{>}}}
        ]
        ({sin(deg(2*x))}, {sin(deg(x))});
    \end{axis}
  \end{tikzpicture}
```

pgfplotstodo.tex:226   **Documentation Todo**   [closed, Priority 5, Epic '']

document some FAQ for number formatting options.

This should contain how to get non-exponential number printing for log axes

pgfplotstodo.tex:230   **Documentation Todo**   [closed, Priority 5, Epic '']

|\pgfplotspointplotattime|.

pgfplotstodo.tex:242   **Documentation Todo**   [closed, Priority 5, Epic '']

document the possibiliy of skewed 3d axes by means of manually provided unit vectors

pgfplotstodo.tex:259   **Documentation Todo**   [closed, Priority 5, Epic '']

the \addplot table from is still supported – document a footnote about the "from" keyword.

```
  \begin{tikzpicture}
```

```
    \begin{axis}
% All these things are valid:
            \pgfplotstableread{data-set-two.txt}\datatable
        \addplot table[y = c] {\datatable} ;
        \addplot table[y = d] \datatable ;
        \addplot table[y = a] from \datatable ;
        \addplot table[y = b] from {\datatable} ;
    \end{axis}
  \end{tikzpicture}
```

Documentation Todo   [closed, Priority 5, Epic '']

contour: documentation is missing in large parts.

mentioning of point meta is missing .

Documentation Todo   [closed, Priority 5, Epic '']

document the new 'data cs' feature

Documentation Todo   [closed, Priority 5, Epic '']

Document how to make mesh plots with (white) filled cells (see matlabs mesh function).

Should be the same as surf with faceted color=white.

Documentation Todo   [closed, Priority 5, Epic '']

Document `scale mode` and other plot graphics related fine tunings

Documentation Todo   [closed, Priority 5, Epic '']

improve docs for `\pgfplotsforeachungrouped`:

```
        \pgfplotsforeachungrouped \i/\j in {
            1 / a,
            2 / b,
            3 / c
        }{
    \edef\temp{\noexpand\node at (axis cs: \i,0.5) {\j};}
    % \show\temp % zum verstaendnis, was als resultat dann in \temp steht
    \temp
        }
```

27

```
                    \pgfplotsforeachungrouped \i/\j in {
                          1 / a,
                          2 / b,
                          3 / c
                      }{
                   I = \i, J = \j;
                      }
```

Documentation Todo   [closed, Priority 5, Epic '']

mention `xtick=data` in docs for `symbolic x coords`

Documentation Todo   [closed, Priority 5, Epic '']

provide more examples and more detailed docs for `xbar` and `ybar` plot handlers

docs: Wie gehabt, die Groesse, Aufloesung und die Zuordnung der Axen etwas detailierter zu beschreiben waere so mein Tip

Example files: [see bugtracker/minimal_0.pdf]

```
\documentclass[a4paper]{report}
\usepackage{pgfplots}

\pgfplotsset{compat=1.3}
\begin{document}
\begin{tikzpicture}
  \begin{axis}[
    xbar,
    width=12cm,
    height=3.5cm,
    enlarge y limits=0.5,
    xlabel={\#participants},
    xmin=0,
    symbolic y coords={no,yes},
    ytick=data,
    nodes near coords,
    nodes near coords align={horizontal},
    ]
    \addplot coordinates {(3,no) (7,yes)};
  \end{axis}
\end{tikzpicture}

\begin{tikzpicture}
  \begin{axis}[
    xbar,
    width=12cm,
    height=3.5cm,
    enlarge y limits=0.5,
    xlabel={\#participants},
    symbolic y coords={no,yes},
    ytick=data,
    nodes near coords,
    nodes near coords align={horizontal},
    ]
    \addplot coordinates {(1,no) (9,yes)};
```

```
    \end{axis}
\end{tikzpicture}

\begin{tikzpicture}
  \begin{axis}[
    xbar,
    width=12cm,
    height=3.5cm,
    enlarge y limits=0.5,
    xlabel={\#participants},
    xmin=0,
    symbolic y coords={set A,set B},
    ytick=data,
    nodes near coords,
    nodes near coords align={horizontal},
    ]
    \addplot coordinates {(6,set A) (4,set B)};
  \end{axis}
\end{tikzpicture}

\begin{tikzpicture}
  \begin{axis}[
    ybar,
    enlargelimits=0.15,
    xlabel={\# of bananas},
    ylabel={\#participants},
    ytick={0,1,2,3},
    ymin=0,
    symbolic x coords={1,2,3,4,5,more},
    nodes near coords,
    ]
    \addplot coordinates {(1,1) (2,1) (3,3) (4,2) (5,1) (more,2)};
  \end{axis}
\end{tikzpicture}

\begin{tikzpicture}
  \begin{axis}[
    ybar stacked,
    enlargelimits=0.15,
    legend style={at={(0.5,-0.20)},
      anchor=north,legend columns=-1},
    ylabel={\#participants},
    symbolic x coords={tool1, tool2, tool3, tool4, tool5, tool6, tool7},
    xtick=data,
    x tick label style={rotate=45,anchor=east},
    ]
    \addplot+[ybar] plot coordinates {(tool1,0) (tool2,2) (tool3,2) (tool4,3) (tool5,0) (tool6,2) (tool7,0)}; %
    \addplot+[ybar] plot coordinates {(tool1,0) (tool2,0) (tool3,0) (tool4,3) (tool5,1) (tool6,1) (tool7,0)}; %
    \addplot+[ybar] plot coordinates {(tool1,6) (tool2,6) (tool3,8) (tool4,2) (tool5,6) (tool6,5) (tool7,6)}; %
    \addplot+[ybar] plot coordinates {(tool1,4) (tool2,2) (tool3,0) (tool4,2) (tool5,3) (tool6,2) (tool7,4)}; %
    \legend{never, rarely, sometimes, often}
  \end{axis}
\end{tikzpicture}

\begin{tikzpicture}
  \begin{axis}[
    ybar,
    enlargelimits=0.15,
    legend style={at={(0.5,-0.15)},
      anchor=north,legend columns=-1},
    ylabel={\#participants},
    symbolic x coords={tool8,tool9,tool10},
```

```
    xtick=data,
    nodes near coords,
    nodes near coords align={vertical},
    ]
    \addplot coordinates {(tool8,7) (tool9,9) (tool10,4)};
    \addplot coordinates {(tool8,4) (tool9,4) (tool10,4)};
    \addplot coordinates {(tool8,1) (tool9,1) (tool10,1)};
    \legend{used,understood,not understood}
  \end{axis}
\end{tikzpicture}

\begin{tikzpicture}
  \begin{axis}[
    ybar,
    enlargelimits=0.15,
    legend style={at={(0.5,-0.2)},
      anchor=north,legend columns=-1},
    ylabel={\#participants},
    symbolic x coords={excellent,good,neutral,not good,poor},
    xtick=data,
    nodes near coords,
    nodes near coords align={vertical},
    x tick label style={rotate=45,anchor=east},
    ]
    \addplot coordinates {(excellent,0) (good,8) (neutral,2) (not good,0) (poor,0)};
  \end{axis}
\end{tikzpicture}

\begin{tikzpicture}
  \begin{axis}[
    ybar,
    enlargelimits=0.15,
    legend style={at={(0.5,-0.2)},
      anchor=north,legend columns=-1},
    ylabel={\#participants},
    symbolic x coords={excellent,good,neutral,not good,poor},
    xtick=data,
    nodes near coords,
    nodes near coords align={vertical},
    x tick label style={rotate=45,anchor=east},
    ]
    \addplot coordinates { (excellent,0) (good,7) (neutral,3) (not good,0) (poor,0)};
  \end{axis}
\end{tikzpicture}
\end{document}
```

pgfplotstodo.tex:468   **Documentation Todo**   [closed, Priority 5, Epic '']

release notes: mention improvements of 'shader=interp'

pgfplotstodo.tex:472   **Documentation Todo**   [closed, Priority 5, Epic '']

There is a typo on section 4.5.12: "As for for dimensional patch plots "

pgfplotstodo.tex:477   **Documentation Todo**   [closed, Priority 5, Epic '']

quiver: the tests have a further pretty example where quiver is on top of a surf, attached to z =2 or so.

pgfplotstodo.tex:481	**Documentation Todo**	[closed, Priority 5, Epic ']

document 'shader=faceted interp'

pgfplotstodo.tex:485	**Documentation Todo**	[closed, Priority 5, Epic ']

document 'mesh/type'

pgfplotstodo.tex:489	**Documentation Todo**	[closed, Priority 5, Epic ']

document the 'plot graphics/points' feature.

pgfplotstodo.tex:499	**Documentation Todo**	[closed, Priority 5, Epic ']

document 'contour prepared', 'contour external' and 'contour gnuplot'.

pgfplotstodo.tex:503	**Documentation Todo**	[closed, Priority 5, Epic ']

contour external: Do not forget the `\"`, `\'` etc special handling .

pgfplotstodo.tex:507	**Documentation Todo**	[closed, Priority 5, Epic ']

contour: document 'labels over line' style

pgfplotstodo.tex:515	**Documentation Todo**	[closed, Priority 5, Epic ']

contour: document the special handling of "point meta".

pgfplotstodo.tex:523	**Documentation Todo**	[closed, Priority 5, Epic ']

clickable: document 'popup size' and its variants document 'clickable coords size' document 'richtext' and the formatting things document `\n` and friends

pgfplotstodo.tex:528	**Documentation Todo**	[closed, Priority 5, Epic ']

document ternary lib + do not forget 'cartesian cs' and its applications

pgfplotstodo.tex:532	**Documentation Todo**	[closed, Priority 5, Epic ']

document frac whole format

`pgfplotstodo.tex:536` **Documentation Todo**   [closed, Priority 5, Epic '']

document /pgfplots/empty line

`pgfplotstodo.tex:540` **Documentation Todo**   [closed, Priority 5, Epic '']

document 'clickable coords' and 'clickable coords code' features

`pgfplotstodo.tex:553` **Documentation Todo**   [closed, Priority 5, Epic '']

document the new 'getcolumnbyname=create col/....' feature

`pgfplotstodo.tex:557` **Documentation Todo**   [closed, Priority 5, Epic '']

document linear regression

`pgfplotstodo.tex:567` **Documentation Todo**   [closed, Priority 5, Epic '']

document how to fix dimension too large problems: restrict to domain for
example

`pgfplotstodo.tex:571` **Documentation Todo**   [closed, Priority 5, Epic '']

colorbar styles are not consistent between docs and code

`pgfplotstodo.tex:581` **Documentation Todo**   [closed, Priority 5, Epic '']

pgfplotstable: show how to use '`\begin{longtable}`'

4   Bugs/Features in PGF/TikZ

`pgfplotstodo.tex:598` **pgfbug**   [open, Priority 5, Epic '']

external library does not compute the shell escape flag for sub-processes
correctly if run using lualatex.

Furthermore, it would be smart to auto-detect the lualatex driver
automatically

`pgfplotstodo.tex:624` **pgfbug**   [open, Priority 5, Epic '']

When reading the manual v2.0 I found a typo 5.1 "Styling the nodes". Just
after the first block of code, there is a sentence saying "... can achieve them.
Once way is to use ..." which should be "One way is to use ..."

pgfbug   [open, Priority 5, Epic '']

Beamer + pgf: the default template introduces a white line on top.
Interestingly, it happens only for PGF CVS + beamer, but it appears to be
dependent on third-party tools as well (see mail conversation with Stefan
Tibus)

pgfbug   [open, Priority 5, Epic '']

When using externalize function together with a transform canvas, the result
is somehow croped. See this example, compare output with deativated and
activated externalize.

```
\documentclass{article}
\usepackage{tikz}
\usetikzlibrary{external}
\tikzexternalize % activate!
\begin{document}
\begin{tikzpicture}[transform canvas={scale=0.7}]
\node {root}
child {node {left}}
child {node {right}
child {node {child}}
child {node {child}}
};
\end{tikzpicture}
A simple image is \tikz \fill (1,0) circle(5pt);.
\end{document}
```

pgfbug   [open, Priority 5, Epic '']

pgf users Vol 50 issue 6:

```
Hi,

Thanks for TikZ.  I'm trying to use the externalization library with
the class file gOMS2e.cls, which is provided for the journal
Optimization Methods and Software.  The class file and related
files/documentation can be found here:
http://www.tandf.co.uk/journals/authors/gomslatex.zip

My problem is that the externalized figures are shifted up and to the
left significantly, cutting them off.  This problem does not occur
when not using externalization.
This seems to be related to the problem discussed here:
http://sourceforge.net/tracker/index.php?func=detail&aid=3037831&group_id=142562&atid=7527
and may also be related to this one:
```

In the other cases, the solution was to use \tikzifexternalizing for
whatever conflicts with the externalization, but it seems that I can't
do this when my class file is the offending bit. Is this true? I
would really like to be able to use the correct \documentclass to
generate the figures so that the size/fonts/etc. are consistent
throughout the resulting document.

A minimal test example is included at the end of this message. It
appears that the image is shifted ~1.25cm to the left and ~0.8cm up.
The problem goes away when using \documentclass{article}.
I'm using the CVS version of pgf, and I get the same result when I
produce postscript figures by using latex and setting
 \tikzset{external/system call={
   latex \tikzexternalcheckshellescape -halt-on-error
-interaction=batchmode -jobname "\image" "\texsource";
   dvips -o "\image".ps "\image".dvi}}

```
%--------------------------------------------------------------------------------
\documentclass[printer]{gOMS2e}
\usepackage{tikz}
\usetikzlibrary{external}
\tikzexternalize
\begin{document}
\begin{center}
  \begin{tikzpicture}
    \draw[step=.5cm] (-3,-3) grid (3,3);
    \draw[blue,line width=2mm] (-0.5,-3) -- (-0.5,1.2) -- (3,1.2);
  \end{tikzpicture}
\end{center}
\end{document}
%--------------------------------------------------------------------------------
```

Any help would be appreciated; I'm afraid it's over my head at this point.
Thanks!

pgfplotstodo.tex:745   **pgfbug**   [open, Priority 5, Epic '']

```
\documentclass{article}
\usepackage{german}
\usepackage[utf8]{inputenc} % erlaubt direkte Nutzung von Umlauten

\usepackage{pgfplots} % fuer plots

\usepackage{pgfplotstable} % fuer numeriktabellen
\usepackage{array,colortbl,booktabs}
```

19

```
\usetikzlibrary{external}
\tikzexternalize[force remake]

% DOESN'T WORK. Needs to disable externailization
\usepackage{vmargin}
\setpapersize{A4}
\setmarginsrb{2.5cm}{1cm}{2cm}{2cm}{8mm}{15mm}{5mm}{15mm}

\begin{document}
\begin{tikzpicture}
%\tracingmacros=2 \tracingcommands=2
\begin{axis}
\addplot {x};
\end{axis}
\end{tikzpicture}
\end{document}
```

pgfbug   [open, Priority 5, Epic '']

external bug:

```
\documentclass[
    pagesize=auto,                  % 1
        ]{scrbook}
\usepackage{tikz}
    \usetikzlibrary{external}
        \tikzexternalize
\begin{document}
        \KOMAoption{twoside}{semi}  % 2
            test
    \tikz \draw (0,0) circle (3pt);
\end{document}
```

pgfbug   [open, Priority 5, Epic '']

consider a matrix style which applies only to the outer matrix node style (see feature request
`https://sourceforge.net/tracker/?func=detail&atid=1060657&aid=3019259&group_id=224188`
)

pgfbug   [open, Priority 5, Epic '']

make assignments to \pgf@x and \pgf@y always \global

pgfplotstodo.tex:780   **pgfbug**   [open, Priority 5, Epic '']

implement `\pgfmathfloattocount`

pgfplotstodo.tex:834   **pgfbug**   [open, Priority 5, Epic '']

the fpu can't be used inside of paths. That should be fixed. ⤳ the problem is that paths may use `\pgfmath...` routines directly. ⤳ this should work! At least with the public math macros `\pgfmathadd`. The `\pgfmathadd@` might be implemented differently.

pgfplotstodo.tex:883   **pgfbug**   [open, Priority 5, Epic '']

pack the default 'system call' for dvips etc into drivers!

pgfplotstodo.tex:887   **pgfbug**   [open, Priority 5, Epic '']

active '—' characters result in compilation bugs (`\usepackage{program}`)

pgfplotstodo.tex:891   **pgfbug**   [open, Priority 5, Epic '']

'text height=1em' realisieren mit [node font units]1em

pgfplotstodo.tex:615   **pgfbug**   [closed, Priority 5, Epic '']

number printer: apply `set thousands separator={\cdot}` also to fractional parts: [see bugtracker/minimal_1.pdf]

```
\documentclass{article}
   \usepackage{pgf}
      \pgfset{/pgf/number format/.cd,
         set thousands separator={{{\cdot}}},
         precision=5,
      }
\begin{document}
   \pgfmathprintnumber{12345.54321} \par
   $12 \cdot 2345.543 \cdot 21$ expected \par
\end{document}
```

pgfplotstodo.tex:716   **pgfbug**   [closed, Priority 5, Epic '']

|\pgfmathdivide@{-0.8}{1.00002}\pgfmathresult|yields

-0.8

instead of -0.8

`pgfplotstodo.tex:748`   **pgfbug**   [closed, Priority 5, Epic '']

Implement support for space trimming and empty entries in `\usetikzlibrary` and its variants

`pgfplotstodo.tex:785`   **pgfbug**   [closed, Priority 5, Epic '']

external lib: think whether it is possible to provide the real jobname without explicit user input. Idea: transport it as TeX code argument to pdflatex

`pgfplotstodo.tex:789`   **pgfbug**   [closed, Priority 5, Epic '']

provide '×' or more general formatting rules to number printer

`pgfplotstodo.tex:793`   **pgfbug**   [closed, Priority 5, Epic '']

code 2 args doesn't work correctly with spaces between the arguments!?

`pgfplotstodo.tex:798`   **pgfbug**   [closed, Priority 5, Epic '']

external lib: implement `\tikzpicturedependsonfile#1`

`pgfplotstodo.tex:803`   **pgfbug**   [closed, Priority 5, Epic '']

in pgfplots: invoke `\tikzpicturedependsonfile`. perhaps the plot-from-table-struct should also use it.

`pgfplotstodo.tex:807`   **pgfbug**   [closed, Priority 5, Epic '']

external lib: 'list and make' does not work together with `\include` (aux files!) or other file writing things – at least not if one tries to do that in parallel.

`pgfplotstodo.tex:811`   **pgfbug**   [closed, Priority 5, Epic '']

consider the "plot function" patch from Andy Schlaikjer

`pgfplotstodo.tex:815`   **pgfbug**   [closed, Priority 5, Epic '']

it seems fadings don't work correctly with externalization!?

`pgfplotstodo.tex:819`   **pgfbug**   [closed, Priority 5, Epic '']

include addition of Christophe Jorssen for MD5 checksums in external lib

pgfplotstodo.tex:823   **pgfbug**   [closed, Priority 5, Epic '']

write new sub-package 'pgfmanual.sty' which contains a good user interface to the manual styles, environments and all that.

pgfplotstodo.tex:827   **pgfbug**   [closed, Priority 5, Epic '']

external lib: catcode changes inside of pictures do not work properly.

pgfplotstodo.tex:839   **pgfbug**   [closed, Priority 5, Epic '']

in the manual, the first two arguments of pgfqkeysactivatesinglefamilyandfilteroptions were inverted.

pgfplotstodo.tex:843   **pgfbug**   [closed, Priority 5, Epic '']

some predefined filters do not process unknown options correctly

pgfplotstodo.tex:850   **pgfbug**   [closed, Priority 5, Epic '']

external lib in pgf: think whether 'empty image extension' is a bug or a feature. ⤳ feature of \pgfimage! Otherwise it wouldn't be possible to provide an extension! ⤳ bug for external lib which never uses extensions!

pgfplotstodo.tex:874   **pgfbug**   [closed, Priority 5, Epic '']

fix landscape bug (pdflscape) in external lib (PGF) [see bugtracker/minimal_2.pdf]

```
\pdfcompresslevel=0
\documentclass{article}
\usepackage{pgfplots}
%\usepackage{lscape}
\usepackage{pdflscape}

\usetikzlibrary{external}
\tikzexternalize

\begin{document}
\begin{landscape}
La.
\begin{tikzpicture}
  \node at (0,0) {horizontal Text.};
\end{tikzpicture}
\end{landscape}

\end{document}
```

pgfbug   [closed, Priority 5, Epic '']

the pgf math parser has wrong precedence for '-' prefix op: `exp(-x^2)` is wrong.

pgfbug   [closed, Priority 5, Epic '']

compatiblity code todo: - the example for plot graphics (with view=090) doesn't work. ⤳ that's the 'exp(0-x^2)' bug which is still in pgf 2.00!

5   Bugs in PGFPlots

Bug   [open, Priority 5, Epic '']

Tick labels suffer from inexact arithmetics in Tikz foreach:

```
KAPUTT:

  \foreach \x in {1,1.1,...,2} {\x\par}

OK:

  \foreach \x in {1,1.1,...,2.001} {\x\par}
```

Bug   [open, Priority 5, Epic '']

ytick=data combined with minor y tick num does not work as expected: minor tick lines will be skipped below the lowest limit.

Bug   [open, Priority 5, Epic '']

pgfplotstable does not apply `postproc cell content` in `every row` styles

see `http://tex.stackexchange.com/questions/111492/postprocess-row-with-pgfplotstable`

Bug   [open, Priority 5, Epic '']

no markers: it appears to be impossible to switch off markers, only for scatter plots.

See `http://tex.stackexchange.com/questions/105850/pgfplots-points-with-no-marks-but-errorbars`

Bug [open, Priority 5, Epic '']

units and square brackets

Dann habe ich noch eine Anmerkung bzgl. der Verwendung von Einheiten in Achsbeschriftungen

------ Zitat
ANMERKUNG;
Die eckigen Klammern drfen nicht um Einheitenzeichen gesetzt werden. Angaben wie [kg] sind
------ Ende Zitat

'DIN 461: Graphische Darstellung in Koordinatensystemen' gibt Beispiele, wie es gemacht we

Hintergrund: formaler Zusammehang ist X={X}[X]. Das heit: X=5N --> {X}=5 und [X]=N. Unsinn

Wie gesagt: leider eine sehr schlechte Angewohnheit und den meisten unbekannt oder schlich

Bug [open, Priority 5, Epic '']

|...|and rounding:

Generell erscheint mir die ...-Syntax nicht robust. Bei kleinen, durchaus
nicht ungewhnlichen, Intervallen kommt es zu Rundungsfehlern:

Hauptgitter:

ytick={0.99,1.00,1.01,1.02,1.03,1.04,1.05,1.06,1.07,1.08}

Hilfsgitter:

minor ytick={0.99,0.991,...,1.08}

Fhrt dazu, dass die Hilfslinien mit zunehmender Gre immer weiter von den
Hauptgitterlinien verschoben sind, obwohl diese aufeinanderliegen sollten. Mit
Hilfsgitter komplett ausgeschrieben:

minor ytick={0.99,0.991,0.992,0.993,..............,1.08}

passt es. Die ...-Syntax ist daher absolut mit Vorsicht zu genieen.

Bug [open, Priority 5, Epic '']

box plots with draw direction=y and xtick=data fail to provide the correct
xtick locations

Bug [open, Priority 5, Epic '']

boxplots: predefined legend entries need to be improved

pgfplotstodo.tex:1003   **Bug**   [open, Priority 5, Epic '']

presets for mark size and tiny / footnotesize are wrong

pgfplotstodo.tex:1021   **Bug**   [open, Priority 5, Epic '']

clipping of tick lines does not respect the line width of the axis lines.

http://tex.stackexchange.com/questions/91517/
how-to-make-the-tick-thickness-as-the-axis-line/91645#91645

pgfplotstodo.tex:1045   **Bug**   [open, Priority 5, Epic '']

Inf geht nicht im math parser:

```
\documentclass[a4paper]{article}

\usepackage{pgfplots}


\begin{document}

\pgfmathfloatparsenumber{Inf}
\pgfmathresult

{
\pgfkeys{/pgf/fpu}
\pgfmathparse{Inf}
\pgfmathresult
}

\end{document}
```


pgfplotstodo.tex:1064   **Bug**   [open, Priority 5, Epic '']

|disablelogfilter,ymax=1e-6, ymode=log— fails. Apparently, the
coordinate is not parsed at all.

pgfplotstodo.tex:1068   **Bug**   [open, Priority 5, Epic '']

gnuplot interface: unbounded coords are not recognized as such (type=u).

pgfplotstodo.tex:1072   **Bug**   [open, Priority 5, Epic '']

gnuplot interface + raw gnuplot does not handle log scale properly (?)

Bug [open, Priority 5, Epic '']

|\closedcycle|does not work together with jumps / interrupted plots

Bug [open, Priority 5, Epic '']

auto tick label assignment can sometimes produce strange results:

[see bugtracker/minimal_3.pdf]

```
\documentclass{article}
\usepackage{pgfplots}
\pgfplotsset{compat=1.6.1}

\begin{document}

\begin{tikzpicture}
\begin{axis}[%
scale only axis,
xmin=0, xmax=0.02,
ymin=-1, ymax=1]
\end{axis}
\end{tikzpicture}%
\end{document}
```

See also the examples for boxplots in the manual

Bug [open, Priority 5, Epic '']

improper alignment of x tick labels which have different baselines or different heights.

Idea: introduce \strut

```
https://sourceforge.net/tracker/?func=detail&aid=3516368&group_
id=224188&atid=1060656
```

Bug [open, Priority 5, Epic '']

the3d clip path is sometimes bad: perhaps it should be the bounding box instead!?
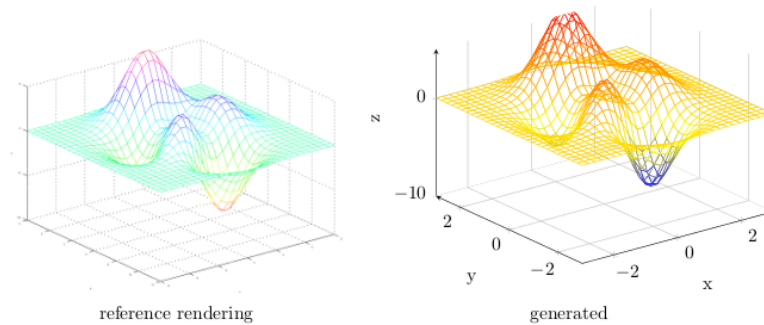
Figure 44: Mesh plot. `meshPlot`

pgfplotstodo.tex:1174   **Bug**   [open, Priority 5, Epic '']

One cannot manually load a table inside of a pgfplots axis if it contains empty lines: the scanline callback is active

pgfplotstodo.tex:1178   **Bug**   [open, Priority 5, Epic '']

|ybarlegend— does not contain `ybar` (sourceforge 3482770)

pgfplotstodo.tex:1182   **Bug**   [open, Priority 5, Epic '']

|`refstyle`|does only include partial information of the reference style. (sourceforge 3482770)

pgfplotstodo.tex:1188   **Bug**   [open, Priority 5, Epic '']

Using square brackets as first char inside of `\legend` leads to a failure: `\legend{[\ion{Ne}{2}],...}`.

Cause: the `\legend` command does not properly insert `[]` in front of every entry (as it ought to).

pgfplotstodo.tex:1194   **Bug**   [open, Priority 5, Epic '']

Adding `error bars/.cd` to `\addplot` options causes the `\ref` image to fail.

The problem is the key filtering apparently: it discards the `/pgfplots/.cd` but leaves the `error bars/.cd`.

85

Bug [open, Priority 5, Epic '']

quiver plots: the clip path does not respect arrow paths

Bug [open, Priority 5, Epic '']

Alignment bug: axis x line=middle combined with a yshift shifts the xlabel incorrectly: [see bugtracker/minimal_4.pdf]

```
\documentclass{report}
\usepackage{pgfplots}
\pgfplotsset{compat=1.3}

\begin{document}
  \begin{tikzpicture}
      \draw (0,0) circle (5pt);
    \begin{scope}[yshift=-3cm]
    \begin{axis}[width=10cm,height=3cm,xlabel={$x$},
      axis x line = middle]
     \addplot coordinates {
       (0,1) (1,-1) (2,1)
     };
    \end{axis}
    \end{scope}
  \end{tikzpicture}
\end{document}
```

Using `xlabel style = {yshift=3cm}` in the plot will correctly position the x label (to its default position).

Bug [open, Priority 5, Epic '']

cannot provide clip path usage in pgfplots commands because of the nested scopes.

to reproduce, try to give `\addplot+[/tikz/clip]` to some plot.

Bug [open, Priority 5, Epic '']

|\pgfplotsforeachungrouped|cannot be combined with three or more arguments like \foreach

Bug [open, Priority 5, Epic '']

the below example of a latex file gives the following error upon the 2nd run of latex. The first run works fine. This happens both when running dvilualatex and just latex, both from TexLive 2011.

The error:

. . .

```
(/usr/local/texlive/2011/texmf-dist/tex/generic/tex4ht/color.4ht)
(/usr/local/texlive/2011/texmf-dist/tex/generic/tex4ht/html4.4ht)
(/usr/local/texlive/2011/texmf-dist/tex/generic/tex4ht/html4-math.4ht))
(./epub.aux)
! Missing \endcsname inserted.
<to be read again>
                    \protect
l.30 \ref{govconsumptionlegend}

?


\makeatletter

\def\HCode{\futurelet\HCode\HChar}\def\HChar{\ifx"\HCode\def\HCode"##1"{\Link##1}\expandaf

\makeatother

\HCode "xhtml,png,charset=utf-8".a.b.c.


\documentclass[11pt,a4paper]{book}

\def\pgfsysdriver{pgfsys-tex4ht.def}
\usepackage{pgfplots}

\pgfplotsset{width=\textwidth,compat=1.3,every axis/.append style={font=\footnotesize},cyc

\begin{document}
\begin{tikzpicture}

\begin{axis}[ylabel=\%,x tick label style={ /pgf/number format/1000 sep=},ymin=0,xmin=1950
\addplot[smooth,solid] coordinates {
(1950,12.98732304) (1951,11.18937899) (1952,10.63447043) (1953,11.25741618) (1954,11.35201
};
\addlegendentry{Country 1}
\addplot[smooth,dotted] coordinates {
(1950,8.90574995) (1951,9.181850378) (1952,9.4040808) (1953,9.790597533) (1954,9.766571438
};
\addlegendentry{Country 2}
\end{axis}\end{tikzpicture}

\ref{govconsumptionlegend}

\end{document}
```

pgfplotstodo.tex:1688   **Bug**   [open, Priority 5, Epic '']

http://groups.google.at/group/comp.text.tex/msg/adcb1d071c2cba40

If I use a yshift in a scope to draw two graphs superimposed, the x label in the second plot (the one in the yshift scope) is not positioned correctly. I need to manually add another yshift, with the same value in the opposite direction, to get the label at the correct place. This happens if the `axis x line = middle` option is used. Without that option, the x label is positioned correctly. Example follows: [see bugtracker/minimal_5.pdf]

```
\documentclass{article}

\usepackage{pgfplots}

\pgfplotsset{compat=1.3}

\begin{document}

  \begin{tikzpicture}
    \begin{axis}[width=10cm,height=3cm,xlabel={$x$}]
      \addplot coordinates {
        (0,1) (1,-1) (2,1)
      };
    \end{axis}
    \begin{scope}[yshift=-3cm]
    \begin{axis}[width=10cm,height=3cm,xlabel={$x$},
        axis x line = middle]
      \addplot coordinates {
        (0,1) (1,-1) (2,1)
      };
    \end{axis}
    \end{scope}
  \end{tikzpicture}
\end{document}
```

Using `xlabel style = {yshift=3cm}` in the second plot will correctly position the x label (to its default position).

Gab

---

pgfplotstodo.tex:1864   **Bug**   [open, Priority 5, Epic '']

after using a preset key (milli) with x SI prefix, Next, I want to switch to the normal mode, so I write simply: x SI prefix=none, unfortunately the 'none' value is undefined and the compilation can not proceed

---

pgfplotstodo.tex:1926   **Bug**   [open, Priority 5, Epic '']

external lib + dvi/ps + windows: it seems the ';' doesn't work; use '&' to separate commands

---

pgfplotstodo.tex:1930   **Bug**   [open, Priority 5, Epic '']

check y tick scale label for 2nd y axis

`pgfplotstodo.tex:1947` **Bug** [open, Priority 5, Epic '']

groupplots + extra braces or foreach are incompatible.

`pgfplotstodo.tex:1951` **Bug** [open, Priority 5, Epic '']

numplotspertype and forget plot and ybar interval yields errors.

`pgfplotstodo.tex:1957` **Bug** [open, Priority 5, Epic '']

expression plotting and empty 'y' results in errors. Perhaps it would be better to handle that explicitly somehow? (occurs for hist when one input line is empty)

`pgfplotstodo.tex:1977` **Bug** [open, Priority 5, Epic '']

/pgfplots/samples at and /tikz/samples at work on the same axe. Tantau says that this key support foreach statement and thus the dotes notation. However, when I want to use two or more different dots notation within pgfplots, latex crashes ! Here is an example which clarify this issue :
`\addplot+[mark=none,variable={\t}, samples at = {\foreach \x in {0,10,...,180,200,...340}`

`pgfplotstodo.tex:1987` **Bug** [open, Priority 5, Epic '']

groupplots: mixing 2d/3d in one groupplot doesn't reset 'zmin,zmax' ?

`pgfplotstodo.tex:2008` **Bug** [open, Priority 5, Epic '']

3D axes: it is difficult to get an 1:1 correspondence to tikz.

`pgfplotstodo.tex:2024` **Bug** [open, Priority 5, Epic '']

3D axis: provide support for manual axis configuration, - depth (n vector), - foreground/background, - tick label axes, - ...

`pgfplotstodo.tex:2053` **Bug** [open, Priority 5, Epic '']

plot graphics: `\ref` legend doesn't work properly

`pgfplotstodo.tex:2074` **Bug** [open, Priority 5, Epic '']

ternary axes: the 'marker clipping' doesn't work (naturally)

`pgfplotstodo.tex:2084`   **Bug**   [open, Priority 5, Epic '']

polar axes:

- ✓is wrong since 'near ticklabel' anchor uses pointunitx which is not correctly initialised for polar axes.

- axis equal

- ✓data scaling needs to be disabled for X axis.

- ✓auto tick labels work only for the case of disableddatascaling

`pgfplotstodo.tex:2089`   **Bug**   [open, Priority 5, Epic '']

contour: the table/meta=2 default is wrong.

`pgfplotstodo.tex:2102`   **Bug**   [open, Priority 5, Epic '']

dimension too large sanity checking: TeX uses the maximum value instead. Perhaps that can be checked?

`pgfplotstodo.tex:2138`   **Bug**   [open, Priority 5, Epic '']

the quiver/scale arrows thing might need an "auto" option. If I don't add it now, it'll probably never work in the future.

`pgfplotstodo.tex:2143`   **Bug**   [open, Priority 5, Epic '']

'1.23456e4;' in a log plot resulted in hard-to-read error messages. Improve sanity checking here.

`pgfplotstodo.tex:2159`   **Bug**   [open, Priority 5, Epic '']

avoid dimension too large errors which occur due to a data range restrictions. Example: data range = 0:6000 view range = 0:1 ⇝ results in error. But that's easy to detect! Just compute the point coordinate in float (after the scaling is complete). Then, install a filter somewhere. perhaps an "a posteriori" filter in the pointxyz command?

`pgfplotstodo.tex:2173`   **Bug**   [open, Priority 5, Epic '']

|yticklabels={<list>},extra y ticks=...— is incompatible since the extra ticks share the same tick typesetting routine (which, in turn, queries the ¡list¿).

pgfplotstodo.tex:2187   **Bug**   [open, Priority 5, Epic '']

The 'text depth' in legend entries is incompatible with 'text width'. The problem: text width is implemented using `\begin{minipage}[t]` so its contents is all in the depth. Setting text depths overrides the height!

pgfplotstodo.tex:2197   **Bug**   [open, Priority 5, Epic '']

one can't provide 'disable log filter' to addplot (but it might be interesting)

pgfplotstodo.tex:2201   **Bug**   [open, Priority 5, Epic '']

FPU: atan doesn't check for unbounded inputs.

pgfplotstodo.tex:2206   **Bug**   [open, Priority 5, Epic '']

unbounded inputs: improve warning messages: they should not contain low level FPU args.

pgfplotstodo.tex:2212   **Bug**   [open, Priority 5, Epic '']

the user interface to set 'tickwidth=0' for a SINGLE axis is not very good: it seems one needs 'xtick style=/pgfplots/tickwidth=0' to do so... ⤳ can be solved if tickwidth has a family, I guess. Something like 'draw' which will not be pulled by pgfplots. But then remains a problem of key paths.

pgfplotstodo.tex:2220   **Bug**   [open, Priority 5, Epic '']

I have seen that 'plot table' with very large files can produce pool size problems – even if the coordinates are all filtered away. In other words: the code can't simply read a file and throw its contents away. The problem appears to be some math parsing using the table/x expr and friends. 'pool size = names of control sequences and file name' ⤳ the math parser could be improved with ifcsname

pgfplotstodo.tex:2224   **Bug**   [open, Priority 5, Epic '']

axis lines and 3D: some tick lines are not drawn, see manual examples

pgfplotstodo.tex:2235   **Bug**   [open, Priority 5, Epic '']

providing zmin/xmax to an axis activates 3D mode, ok – but lower dimensional input routines appear to fail.

`pgfplotstodo.tex:2239` **Bug** [open, Priority 5, Epic '']

one can't provide 'scale' as argument to a (3d) axis

`pgfplotstodo.tex:2248` **Bug** [open, Priority 5, Epic '']

it may still happen that log-axes get only *one* tick label (in my case
`10^{-0.2}`). That should never happen. The range is about ymin=4.7e-1,
ymax=9.5e-1

`pgfplotstodo.tex:2252` **Bug** [open, Priority 5, Epic '']

log samples in plot expression for 3D plots

`pgfplotstodo.tex:2256` **Bug** [open, Priority 5, Epic '']

different log bases and gnuplot

`pgfplotstodo.tex:2270` **Bug** [open, Priority 5, Epic '']

I tried placing a named coordinate inside one axis and using it in another. It
failed.

CF: The axis is drawn inside of its own picture which will only be shifted if
everything has been drawn. That will be the origin of this problem I guess

Miraculously I can use the coordinate outside axis env. So I have reached the
following solution:

`pgfplotstodo.tex:2276` **Bug** [open, Priority 5, Epic '']

plot coordinates doesn't check too well if 1. addplot3 is used but only two
coords are given 2. addplot is given but three coordinates are provided (also
for plot expression)

`pgfplotstodo.tex:2289` **Bug** [open, Priority 5, Epic '']

the compat things are not yet complete: I wanted to check when it is really
necessary (for example if 'x dir' is used)

`pgfplotstodo.tex:2294` **Bug** [open, Priority 5, Epic '']

the nodes near coords feature produces unexpected results when used together
with markers ⤳ this is due to the default configuration of scatter plots.

Bug [open, Priority 5, Epic '']

the ybar style won't be set inside of `\label{}`

Bug [open, Priority 5, Epic '']

axis equal for semilog plots is not correct (?)

Bug [open, Priority 5, Epic '']

backwards compatibility problem: axis descriptions can't contain /pgfplots/
styles any longer! This is a key path issue :-(

Bug [open, Priority 5, Epic '']

BUG: in empty axes, 'xtick=\empty' is ignored.

Bug [open, Priority 5, Epic '']

The automatic tick labeling sometimes produces inconsistent or confusing
labels: 1. engineering and fixed number style are mixed up. 2. If range of an
axis is so small that the labels differ only on the third decimal, still only two
decimals are used.

Bug [open, Priority 5, Epic '']

3D: error bars and stacked plots need to be updated.

Bug [open, Priority 5, Epic '']

interp shader is displayed transparently in evince

Bug [open, Priority 5, Epic '']

```
  3D:  the use of \addplot3 and \addplot is not sanitized properly
Possibilities:
-  used \addplot  when \addplot3 should have been used
-  used \addplot3 where \addplot should have been used.
 What can happen here!? Shouldn't this work in every case?

- The "xtick" value is not applied unless there is a coordinate in the x range:
$\leadsto$ that's the handling of empty figures...
not working:
  \begin{axis}[xtick=0]
  \end{axis}
```

93

```
not working:
  \begin{axis}[xmin=-5,xmax=5,xtick=0]
  \end{axis}
not working:
  \begin{axis}[xmin=-5,xmax=5,xtick=0]
\addplot coordinates { (-10, 0) };
  \end{axis}
working:
  \begin{axis}[xmin=-5,xmax=5,xtick=0]
\addplot coordinates { (0, 0) };
  \end{axis}
```

pgfplotstodo.tex:2393   **Bug**   [open, Priority 5, Epic '']

think about basic level commands for the axis lines – this should also allow !

pgfplotstodo.tex:2426   **Bug**   [open, Priority 5, Epic '']

In 3D case axis [xyz] line != box, there is just ONE hyperplane. My implementation works only if either ALL are box or ALL are 'middle'.

pgfplotstodo.tex:2436   **Bug**   [open, Priority 5, Epic '']

3D case : tick/grid lines are on top of the axis lines. This leads to poor quality.

pgfplotstodo.tex:2459   **Bug**   [open, Priority 5, Epic '']

javascript stuff does not work if the complete figure is rotated (sidewaysfigure).

pgfplotstodo.tex:2474   **Bug**   [open, Priority 5, Epic '']

javascript: incompatiblity with external library: 1. filenames: `\jobname` contains characters with incompatible catcodes and that funny insdljs package tries to assemble macros with these characters. ⤳ fixed; I simply use pgfplotsJS as temporary file name. 2. the images as such have corrupted forms ⤳ Can be fixed if `\usepackage{eforms}` is used BEFORE loading pgf. The reason: `\begin{Form}` and the shipout-hackery of the pgf externalization bite each other. `\begin{Form}` must come before the shipout hackery of pgf. 3. `\includegraphics` does not preserve PDF forms.

pgfplotstodo.tex:2481   **Bug**   [open, Priority 5, Epic '']

the interrupt bounding box feature should still update the data bounding box. Otherwise, transformations may fail.

Bug   [open, Priority 5, Epic '']

extra ticks can be disabled by the tick special cases for axis lines (when two axis lines cross each other)

Bug   [open, Priority 3, Epic '']

x tick scale label for style `tiny` has an unsuitable shift

Bug   [open, Priority 3, Epic '']

|\addplottable[blue]— ignores the color options!

Bug   [open, Priority 2, Epic '']

new layered graphics stuff: the style changes of the layer config are read too late; it is impossible to overwrite them within the same axis (for example using `set layers,tick style={on layer=...}`)

Bug   [open, Priority 2, Epic '']

#3213889 hyperref boxes are in wrong position for vertical labels

see `http://tex.stackexchange.com/questions/13364/` `how-to-make-pgfplots-vertical-labels-have-proper-hyperref-erence-box` for problem description and potential fixes

Bug   [open, Priority 2, Epic '']

The clipping of tick lines uses the middle of axis lines; it does not incorporate the line width of the axis lines. [see bugtracker/minimal_6.pdf]

```
\documentclass{article}
\usepackage{pgfplots}
\pgfplotsset{compat=1.3,
            every axis/.append style={semithick},
            every tick/.append style={semithick,color=black},
            tick align=outside
            }
\begin{document}
  \thispagestyle{empty}
  \begin{figure}[p]
    \centering
    \begin{tikzpicture}
      \begin{axis}[xmin=0,
                   xmax=30,
                   ymin=0,
                   ymax=1.2
                   ]
      \end{axis}
    \end{tikzpicture}
  \end{figure}
\end{document}
```

Bug   [open, Priority 2, Epic '']

can someone confirm the following behavior. The y label of a plot gets truncated in some circumstances if the external library is used. This happens for me if no title is specified for a plot. Consider the following example:

[see bugtracker/minimal_7.pdf]

```
\documentclass[11pt,a4paper]{article}

\usepackage{tikz}
\usepackage{pgfplots}

\pgfplotsset{compat=1.3}
\usepgfplotslibrary{external}
\tikzexternalize[force remake]

\begin{document}
\begin{tikzpicture}
\begin{axis}[y tick scale label style={inner sep=1pt}]
\addplot {x * 10^8};
\end{axis}
\end{tikzpicture}
\end{document}
```

Bug   [open, Priority 2, Epic '']

potential incompatibility: clickable and external. The clickable lib writes into pgfplots.djs which might cause multithreaded problems.

Bug   [open, Priority 2, Epic '']

OK : 'every node near coord/.append style=scale=0.7' NOT OK: 'every node near coord/.append style=scale=0.7,ybar' -⤳ sequence of shift and scale matters ...

Bug   [open, Priority 2, Epic '']

view=090 and enlargelimits=auto is not always satisfactory: it disables enlarged limits, but for contours, I'd like to have it. What is to do?

Bug   [open, Priority 2, Epic '']

contour external doesn't handle explicitly provided matrix data (mesh/rows and mesh/cols) yet.

Bug   [open, Priority 2, Epic '']

contour external doesn't handle the ordering flag correctly.

pgfplotstodo.tex:2148 **Bug** [open, Priority 2, Epic '']

the title style for 'footnotesize' is not as I want it to: it doesn't respect the depth below the baseline. Or does it need a `\strut`?

pgfplotstodo.tex:1146 **Bug** [open, Priority 1, Epic '']

Internal coordmath framework: it is not used everywhere

pgfplotstodo.tex:1170 **Bug** [open, Priority 1, Epic '']

`|log`number format code— is a global variable and cannot be set for individual axes.

This applies to `log ticks with fixed point` as well.

pgfplotstodo.tex:1472 **Bug** [open, Priority 1, Epic '']

CRASH: [see bugtracker/minimal_8.pdf]

```
\begin{tikzpicture}
        \begin{axis}[
                scale mode=scale uniformly,
                x={(1pt,0pt)},
                y={(-0.5pt,0.5pt)},
                z={(0pt,1pt)},
        ]

        % addplot3 works (with 3d coords):
        \addplot coordinates {
                (0,0) (1,0) (0,1)
        };

        \end{axis}
\end{tikzpicture}
```


pgfplotstodo.tex:1517 **Bug** [open, Priority 1, Epic '']

Using 0 in pgfplots coordinate systems does not necessarily mean "no offset". This is misleading. Bug sourceforge #3168030: [see bugtracker/minimal_9.pdf]

```
\documentclass[a4paper]{article}
\usepackage{german}
\usepackage[utf8]{inputenc}

\usepackage{pgfplots}

\usepackage{pgfplotstable}
\usepackage{booktabs}
\usepackage{array}
\usepackage{colortbl}

\begin{document}
```

97

```
\begin{tikzpicture}
        \begin{axis}[enlarge x limits=false, extra description/.code={\draw[very thick] (axis cs:2.5,0) -- ++(re
                \addplot coordinates{
                                                            (0,1)
                                                            (1,2)
                                                            (2,3)
                                                            (3,4)
                                                            (4,5)};
        \end{axis}
\end{tikzpicture}

\begin{tikzpicture}
        \begin{axis}[enlarge x limits=true, extra description/.code={\draw[very thick] (axis cs:2.5,0) -- ++(rel
                \addplot coordinates{
                                                            (0,1)
                                                            (1,2)
                                                            (2,3)
                                                            (3,4)
                                                            (4,5)};
        \end{axis}
\end{tikzpicture}
\end{document}


\end{document}
```

pgfplotstodo.tex:1645   **Bug**   [open, Priority 1, Epic '']

The legend has the `text depth=0.15em` initial configuration, which is
extremely bad for legend entries with huge depth (large fractionals or
formulas?)

pgfplotstodo.tex:1765   **Bug**   [open, Priority 1, Epic '']

The `mark list` produces a lot of

```
\XC@edef #1#2->\begingroup \ifnum \catcode '\!=13 \edef !{\string !}\fi \ifnum \catcode '\
[........]
{\if}

\@@tmp ->.!80!black
{true}
{the character !}
Missing character: There is no ! in font nullfont!
{the character 8}
Missing character: There is no 8 in font nullfont!
{the character 0}
Missing character: There is no 0 in font nullfont!
{the character !}
Missing character: There is no ! in font nullfont!
{the character b}
Missing character: There is no b in font nullfont!
```

```
{the character l}
Missing character: There is no l in font nullfont!
{the character a}
Missing character: There is no a in font nullfont!
{the character c}
Missing character: There is no c in font nullfont!
{the character k}
Missing character: There is no k in font nullfont!
{\def}
{\else}
```

bugs. Probably fixed with more recent version of xcolor?

Bug   [open, Priority 1, Epic '']

providing `\legend{}` without any `\addplot` commands causes a problem

Bug   [open, Priority 1, Epic '']

the axis line combination styles can't be adjusted for 3D because they are evaluated too early.

Bug   [open, Priority 1, Epic '']

there are a lot of .code 2 args styles which do not support spaces between their arguments. Fix this.

Bug   [open, Priority 1, Epic '']

the '/pgfplots/table/.search also' is overwritten during `\addplot table` with /.search also=/pgfplots. That's not so good.

Bug   [closed, Priority 11, Epic '']

polar lib: the clipping of markers doesn't work correctly for partial polar axes.

Bug   [closed, Priority 10, Epic '']

Markers in legends are not (always?) filled properly [see bugtracker/minimal_10.pdf]

```
\documentclass{article}
    \usepackage{pgfplots}
    \usepackage{pgfplotstable}
\begin{document}

    \begin{tikzpicture}
        \begin{axis}
```

```
        \addplot [mark=*,only marks] coordinates { (-1,1) (1,-1) };
        \legend{measured data}
      \end{axis}
    \end{tikzpicture}
\end{document}
```

caused by the fact that options of 'every axis legend' are in effect at this time
– which includes `fill=white`.

Bug   [closed, Priority 9, Epic '']

SCALING PROBLEMS in 3d:

- the plot box ratio and axis equal feature both need to imply
  `scale mode=scale uniformly`. But it is still wrong; even if one
  activates the correct scale mode.

  The problem: the axis equal stuff operates on the projected unit vectors
  and applies different scalings.

- combining plot box ratio and explicit limits seems to corrupt the display
  (?)

Bug   [closed, Priority 9, Epic '']

french babel and colorbars are not fully compatible. The problem is that
colorbars use '\addplot graphics ;' with a fixed catcode for the ';' – which
might lead to problems.

Bug   [closed, Priority 8, Epic '']

Decorations in plots appear to be problematic (this is a duplicate! caused by
the fact that decorate=false is used at the beginning of every plot, need to
adjust every path style): [see bugtracker/minimal_11.pdf]

```
\documentclass{scrartcl}

\usepackage{pgfplots}
\usetikzlibrary{decorations}

\begin{document}
\begin{tikzpicture}

\begin{axis}

\addplot+[postaction={draw, decorate, decoration=border}] coordinates {(0,0) (5,0.5)}; %funktioniert nicht

\end{axis}

\draw [postaction={draw, decorate, decoration=border}] (0,-3cm) -- ++(5cm,0.5cm); %funktioniert

\end{tikzpicture}
```

```
\begin{tikzpicture}

\begin{axis}

\addplot+[postaction={draw, decorate, decoration=border},
        % tedious, but necessary: pgfplots accidentally resets the
        % "decorate" option at the beginning of the path (probably a
        % bug).
        % This is a work-around:
        every path/.style={
            postaction={decorate},
            every path/.style={},
        },
        ] coordinates {(0,0) (5,0.5)}; %funktioniert nicht

\end{axis}

\end{tikzpicture}
\end{document}
```

Bug   [closed, Priority 7, Epic '']

It is not possible to use `\addplot ... node[pos=0.5] {a};` in pgfplots.

Reason: the timer information is tikz high level, but pgfplots uses the PGF basic layer.

DONE.

Open: the `\pgfplotspointplotattime` should provide more useful output: SCI notation and it should respect custom trafos

Bug   [closed, Priority 5, Epic '']

Bug: bounding box wrong (regression in git repo; not in stable) bisect:

```
3ad5df8050b8c79c67d8831246387cec67ed625d is the first bad commit
commit 3ad5df8050b8c79c67d8831246387cec67ed625d
Author: Christian Feuersaenger <ludewich@users.sourceforge.net>
Date:   Tue Jan 1 21:41:19 2013 +0100

    BB is now tight if hide axis=true - independent of clip path.


reproducable:
pdflatex unittest_enlargelimits_14.tex
```

Bug   [closed, Priority 5, Epic '']

regression: shader=interp can produce wrong clip regions for the shading, compile `http://tex.stackexchange.com/questions/91689/` `visualize-data-on-a-variable-radius-graph-network/93858#93858` to see it

While that issue is solved, there are still BB issues which need to be fixed. In particular, the BB can extend the clip region. Solution approaches:

- ✓compute precise BB (using necessary condition on derivative)

- ✓compute the pdf Xform's bounding box using the method here (i.e. the bounding box derived from control points). BUT: update the picture's BB using the interpolation points (not the control points). This is more natural anyway, and it will probably result in a correct BB. If not, the user can easily extend it.

Bug [closed, Priority 5, Epic '']

discontinuity marks on log axes fail with an error, compare

`http://tex.stackexchange.com/questions/84229/` `discontinuity-of-log-axis-in-pgfplots`

Bug [closed, Priority 5, Epic '']

the postscript driver might fail for advanced shadings. And: there are no tests ...

and: the edgeflag is wrong for triangle shadings.

fix it also for dvipdfmx

Bug [closed, Priority 5, Epic '']

pgfplots, nodes, and remember picture, and cell picture=true fails.

`http://tex.stackexchange.com/questions/72781/` `problem-tikz-pgfplots-and-external-coordinates-using-overlay/` `72804#72804`

Bug [closed, Priority 5, Epic '']

stacked plots + log basis y + log does not work. [see `bugtracker/minimal_12.pdf`]

```
\begin{tikzpicture}
    \begin{axis}[ymode=log
                , ybar stacked
```

102

```
            , log basis y=10
             ]
         \addplot coordinates {(0,1e5)};
         \addplot coordinates {(0,9e5)};
      \end{axis}
\end{tikzpicture}
```

The problem is documented as FIXME in pgfplotsstackedpltos.code.tex

suggested fix: refactor the log and exp methods: always provide the requested
basis explicitly, and provide some 'prepare log basis' method to improve
performance. Do not attach the log basis to the coord math.

Bug   [closed, Priority 5, Epic '']

The default label placement for axis lines=center in 3d appears to be wrong

```
\begin{tikzpicture}
\begin{axis}[
  axis lines=center,
  axis on top,
  xlabel={$x$}, ylabel={$y$}, zlabel={$z$},
  domain=0:1,
  y domain=0:2*pi,
  xmin=-1.5, xmax=1.5,
  ymin=-1.5, ymax=1.5, zmin=0.0,
  mesh/interior colormap=
   {blueblack}{color=(black) color=(blue)},
  colormap/blackwhite,
  samples=10,
  samples y=40,
  z buffer=sort,
 ]
  \addplot3[surf]
   ({x*cos(deg(y))},{x*sin(deg(y))},{x});
\end{axis}
\end{tikzpicture}
```

Potential fixes: (a) redefine `right of origin` and its friends for 3d; (b) do
not use the `right of origin` things, prefer `rel axis cs=1,0.5,0.5`.
Problem: `rel axis cs` must know where the fractions to find the origin (keep
in mind that a rel axis value of 0 means "lower end"). Perhaps some
"constant" value should expand to the fraction for zero?

Potential fix: `http://tex.stackexchange.com/questions/84442/`
`pgfplots-labels-and-width-issues-in-non-boxed-3d-plot-with-oblique-projection`

Bug   [closed, Priority 5, Epic '']

Using `hide axis` or `axis lines=none` causes the axis to vanish – but it will still consume space in the bounding box!

A work-around for the user who reported the bug was to use `clip=false`:

[see bugtracker/minimal_13.pdf]

```
\documentclass{article}
\usepackage[utf8]{inputenc}
\usepackage{pgfplots}
\pgfplotsset{compat=1.4}
\begin{document}

\begin{figure}
  \centering
  \fbox{%
  \begin{tikzpicture}
    \begin{axis}[axis equal,scale=2,axis lines=none,clip=false]
      \addplot3[surf,samples=9,domain=-1:1,y domain=0:2*pi,z buffer=sort,opacity=0.75]
        ({cos(deg(y)) * (1 + x/2 * cos(deg(y)/2))},
        {sin(deg(y)) * (1 + x/2 * cos(deg(y)/2))},
        {x/2 * sin(deg(y)/2)});
    \end{axis}
  \end{tikzpicture}}
  \caption{M"obiusband}
\end{figure}

\end{document}
```

Interestingly, this does NOT work for 1d plots... here is what I found out today:

- excluding the clip path helps for the example above.

- it has no effect for 1d plots (2d axis)

- excluding the background path instruction from the low level node causes the bounding box to be empty – for both 2d and 3d

See `unittest_hideaxis*`.

Seems to be better now (with the axis equal scaling fix)

Bug   [closed, Priority 5, Epic '']

Adding a decoration to a plot requires
`every path/.style={decorate,every path/.style={}}` because pgfplots
sets its options inside of a `\scope[<options>]`.

This should be fixed.

pgfplotstodo.tex:1308 **Bug** [closed, Priority 5, Epic '']

disable tick scale label if the ticks have been disabled.

`https://sourceforge.net/tracker/index.php?func=detail&aid=3457210&group_id=224188&atid=106`

pgfplotstodo.tex:1312 **Bug** [closed, Priority 5, Epic '']

nodes near coords is broken for layer branch

pgfplotstodo.tex:1316 **Bug** [closed, Priority 5, Epic '']

|axisequal,view=090— for a 3d axis leads to compilation errors (although it
seems to work)

pgfplotstodo.tex:1320 **Bug** [closed, Priority 5, Epic '']

xbar and nodes near coords does not automatically align the nodes, see
`http://tex.stackexchange.com/questions/31701/pgfplots-nodes-near-coords-on-xbar-chart-is-o`

pgfplotstodo.tex:1326 **Bug** [closed, Priority 5, Epic '']

view direction is imprecise. It seems as if the z direction is wrong.

See the recent commits on branch `mesh_bg_colormap`

pgfplotstodo.tex:1336 **Bug** [closed, Priority 5, Epic '']

3d: automatic label placement for 'axis lines=center' is buggy

pgfplotstodo.tex:1382 **Bug** [closed, Priority 5, Epic '']

If one specifies \scope within an axis, the plots (partially) use their variables,
but legends do not. [see bugtracker/minimal_14.pdf]

```
\documentclass{article}
    \usepackage{pgfplots}
        \pgfplotsset{
            compat=newest,
        }
\begin{document}
    \begin{tikzpicture}
        \begin{axis}[
% reverse legend, % uncomment and one entry is missing
                legend pos=north west,
        ]
            \begin{scope}[only marks]
                \addplot
                    coordinates { (0,0) (1,1) } node [right] {a};
                \addplot
% [green] % uncomment and legend does exactly the wrong thing
```

```
                        coordinates { (0,1) (1,2) } node [right] {b};
                \end{scope}
                \begin{scope}[mark=none]
                    \addplot
                        coordinates { (0,0.5) (1,1.5) } node [right] {c};
                    \addplot
% [orange] % uncomment and it works
                                % (I think this is luck, because it does the same
                                % thing as the [green] example above)
                        coordinates { (0,1.5) (1,2.5) } node [right] {d};
                \end{scope}
                \legend{
                    a,
                    b,
                    c,
                    d,
                }
            \end{axis}
        \end{tikzpicture}
\end{document}
```

Bug   [closed, Priority 5, Epic '']

the table package does not support non-ASCII column names. If there are
non-ASCII column names, it might fail to produce a readable error message.

Bug   [closed, Priority 5, Epic '']

the unit vector ratio impl does not work as intended: the manual example [see
bugtracker/minimal_15.pdf]

```
\documentclass{article}

\usepackage{pgfplots}
\begin{document}
\begin{tikzpicture}
\begin{axis}[axis equal]
% FokkerDrI_layer_0.patches.dat contains:
% # each row is one vertex; three consecutive
% # vertices make one triangle (patch)
% 105.577 -19.7332 2.85249
% 88.9233 -21.1254 13.0359
% 89.2104 -22.1547 1.46467
% # end of facet 0
% 105.577 -19.7332 2.85249
% 105.577 -17.2161 12.146
% 88.9233 -21.1254 13.0359
% # end of facet 1
\addplot3[patch]
        file
        {plotdata/FokkerDrI_layer_0.patches.dat};
\end{axis}
\end{tikzpicture}
\end{document}
```

fails and resorts to guesses!

Bug   [closed, Priority 5, Epic '']

It is not possible to provide `#` comments in inline tables.

```
\pgfplotstabletypeset[
]{
# GHz dB
1 0
2 -10
3 0
}
```

The problem occurs since the `#` has special handling and many internal checks fail. I started to implement special handling, but that might require vast changes.

One solution is to use

```
\toks0={#1}
\edef\macro{\the\toks0}
```

instead of

```
\def\macro{#1}
```

anywhere in the code – the `\def` introduces special checks for the `#` whereas the `\toks` does not.

Bug   [closed, Priority 5, Epic '']

Groupplots + named nodes doesn't yield the correct output. Perhaps scoping difficulties? Or problems adjusting the stored coords? [see bugtracker/minimal_16.pdf]

```
\documentclass[10pt]{article}

\usepackage{pgfplots}
\usepgfplotslibrary{groupplots}

\begin{document}

\begin{tikzpicture}%
%\begin{axis}[%
\begin{groupplot}[%
        group style={group size=1 by 1},%
]%
        \nextgroupplot;
        \node[name=a] at (axis cs:0.1,-1) {N};
        \addplot coordinates{(0,1) (1,2)};
\end{groupplot}
```

107

```
%\end{axis}

\draw (a) circle (5pt);
\end{tikzpicture}%
\end{document}
```

Bug [closed, Priority 5, Epic '']

Verify that the list termination (either with `\\` or with `,`) works correctly

Bug [closed, Priority 5, Epic '']

ternary lib: `\addplot` doesn't work correctly, only `\addplot3`

Bug [closed, Priority 5, Epic '']

I'm trying to create an extra y tick on a plot, but I want the tick and label to be on the right side of the plot. I want all the other y ticks and labels are all on the left side of the plot.

It's almost working properly, but it won't put the extra label on the right side of the plot where I want it. The tick is appearing on the right side, but the label is staying on the left side with all the other labels. I was using version 1.2.2 before and this was working fine, but I just upgraded to version 1.4 because I wanted to use a new feature that wasn't present in 1.2.2. Is it possible this was broken somewhere along the way?

[see bugtracker/minimal_17.pdf]

```
\documentclass{article}
\usepackage{pgfplots}
\begin{document}
\begin{tikzpicture}
\begin{axis}[
        small,
        width=12cm,
        height=1.8in,
        ymin=0,
                ymax=10,
        xmin=0,
                xmax=2,
        ybar,
        ymajorgrids=true,
        yminorgrids=false,
        minor y tick num=0,
        ytick pos=left,
        xtick pos=left,
        ytick align=center,
        yticklabel={$\pgfmathprintnumber{\tick}\%$},
        xtick align=outside,
        x tick style={},
        xticklabel style={rotate=45,anchor=east,font=\scriptsize\sffamily},
        extra y tick style={tick pos=right, ticklabel pos=right, grid
style={thick,color=black}},
```

```
              extra y ticks={6.25},
              extra y tick labels={Extra Label},
       ]

       %\addplot plot[error bars/.cd,y dir=plus,y explicit,x dir=none] table
       %[x=Index,y expr=100*\thisrow{AvgLocked},y error=Diff]{locked_tabbed.dat};

       \end{axis}
       \end{tikzpicture}
       \end{document}
```

pgfplotstodo.tex:1940   **Bug**   [closed, Priority 5, Epic '']

foreach variants in pgfplots accept only one parameter

```
%              \foreach \x/\y in {1/a, 2/b, 3/c}
%                   {\node at (axis cs:0,\x) {\y};}%   % doesn't work
%              \pgfplotsforeachungrouped \x/\y in {1/a, 2/b, 3/c}
%                   {\node at (axis cs:0,\x) {\y};}%   % doesn't work
```

pgfplotstodo.tex:1961   **Bug**   [closed, Priority 5, Epic '']

view normal vector does not correctly respect plot box ratio and x dir

pgfplotstodo.tex:1965   **Bug**   [closed, Priority 5, Epic '']

plot box ratio has a strange input format (compare with unit vector ratio).

pgfplotstodo.tex:1971   **Bug**   [closed, Priority 5, Epic '']

clickable and Windows Acrobat Reader 9 has been reported to fail

it this still active?

pgfplotstodo.tex:1991   **Bug**   [closed, Priority 5, Epic '']

'clip=false' does not disable marker clipping!

pgfplotstodo.tex:2004   **Bug**   [closed, Priority 5, Epic '']

multiple ordinates: grid lines are drawn on top of function plots; that's bad.
Check: I think you have to change the process line previously invoked, and
make the axes generation at the end : 1. generating adequate grid ⤳ 2.
plotting functions ⤳ 3. creating axes, tick nodes... You can take a minute
look at figure 1 @ "The addplot Command: Coordinate Input" section 4.2 p
19. and you can remark that colour filling overlaps x- and y-axis ! So I suggest

that you use "excute at end picture=¡axis generation code¿" tikz option or similar to avoid this issue.

$\rightarrow$ Should be fixed with layers

pgfplotstodo.tex:2016   **Bug**   [closed, Priority 5, Epic '']

3D axes: providing three unit vectors is not sufficient, one also needs to set 'view='. That should be done automatically.

- 3D axes: Providing three unit vectors manually yields incorrect axis initialisation.

pgfplotstodo.tex:2031   **Bug**   [closed, Priority 5, Epic '']

WONTFIX.

Patch plots: directly transform cdata. Should simplify interpolation during refine/triangulation etc. and shouldn't make a difference otherwise.

pgfplotstodo.tex:2038   **Bug**   [closed, Priority 5, Epic '']

manual errors of given pgfplots_unstable version: 94 2.5.12 `addplot+[patch] --> addplot3+[patch]` 162 "xmode, ymode, zmode" and "x dir, ..." come again on page 177

pgfplotstodo.tex:2043   **Bug**   [closed, Priority 5, Epic '']

don't loose `\ref`'s when externalizing I'll provide a minimal later

pgfplotstodo.tex:2049   **Bug**   [closed, Priority 5, Epic '']

incompatibility pdfpages (most recent version), MikTeX and tikz external lib (something with shipout routine)

pgfplotstodo.tex:2065   **Bug**   [closed, Priority 5, Epic '']

markers should not be drawn on top of everything else. Always restore the clipping region for each plot.

pgfplotstodo.tex:2070   **Bug**   [closed, Priority 5, Epic '']

mesh/patch plots: - jump thing + z buffer=sort probably doesn't work

`pgfplotstodo.tex:2113` **Bug** [closed, Priority 5, Epic '']

provide remark at end document "Package pgfplots: consider using the preamble command
`\pgfplotsset{compat=1.3} to improve label placement`"

`pgfplotstodo.tex:2124` **Bug** [closed, Priority 5, Epic '']

contour external should allow different variations how to deal with end-of-scanline markers. gnuplot requires empty lines; matlab doesn't deal with them as far as I know.

`pgfplotstodo.tex:2166` **Bug** [closed, Priority 5, Epic '']

the autodetection of the '\\' list format is buggy: it should return true if and only if the last element is '\\', not if '\\' occurs inside of the argument.

`pgfplotstodo.tex:2180` **Bug** [closed, Priority 5, Epic '']

'`\addplot[only marks]`' does not assign a plot mark; one needs 'mark=*' explicitly. that's confusing...

see also `https://sourceforge.net/tracker/?func=detail&atid=1060656&aid=3045389&group_id=224188`

`pgfplotstodo.tex:2231` **Bug** [closed, Priority 5, Epic '']

check for placement of tick scale label for compat=newest ⇝ I improved them for 2d and 3d ⇝ needs some further checks, I guess

`pgfplotstodo.tex:2243` **Bug** [closed, Priority 5, Epic '']

getthisrow still has to be fixed

`pgfplotstodo.tex:2260` **Bug** [closed, Priority 5, Epic '']

3D gnuplot: z buffer fails (see tests)

`pgfplotstodo.tex:2280` **Bug** [closed, Priority 5, Epic '']

gnuplot: set terminal table seems to be deprecated.

`pgfplotstodo.tex:2285` **Bug** [closed, Priority 5, Epic '']

gnuplot and 3D ⇝ I need a shared interface to prepare the required keys for expression plotting

pgfplotstodo.tex:2301   **Bug**   [closed, Priority 5, Epic '']

check whether /pgfplots/ keys are processed properly in legends. This is
certainly not the case for the `\label/\ref` legend! ⤳ which ones are the
problem?

pgfplotstodo.tex:2331   **Bug**   [closed, Priority 5, Epic '']

finish impl of ticklabel pos. I should use the same thing for tickpos as well.
And: the default arg processing which uses ticklabel pos = tickpos needs to be
fixed. the 2D axes are wrong.

pgfplotstodo.tex:2346   **Bug**   [closed, Priority 5, Epic '']

3D: axis equal implementation might not be correct (at least not for view
special cases)

pgfplotstodo.tex:2357   **Bug**   [closed, Priority 5, Epic '']

the `\thisrow` commands in the table package does not (always) respect aliases!

pgfplotstodo.tex:2400   **Bug**   [closed, Priority 5, Epic '']

the arguments to `plot file[#1] and plot table[#1]` are not consistent
with rest. They need to be treated as behavior options (maybe in a different
key path).

pgfplotstodo.tex:2421   **Bug**   [closed, Priority 5, Epic '']

verify that 'draw=none' works! Is something broken here? ⤳ write tests! + it
appears to be desired that (at least some) markers invoke
`\pgfusepathqfillstroke` ⤳ they always 'draw', regardless of tikz color
settings. ⤳ ok, I patched that in my marker code... (hackery :-( ) - no, it
works only partially: draw=none or fill=none works as expected. But 'blue'
disables filling!? - Possible fix: Overwrite
`\filltrue \fillfalse, \drawtrue, \drawfalse`: they should set a further
boolean '`\drawbooleanhasbeenset`' and '`\fillbooleanhasbeenset`'. ⤳
Replace the `\pgfusepathqfillstroke` if and only if the respective booleans
have been set *explicitly*. If they are unchanged, fall back to a "reasonable"
default.

pgfplotstodo.tex:2431   **Bug**   [closed, Priority 5, Epic '']

3D case : grid lines work correctly, but they are not satisfactory. I'd like grid
lines in the background only.

112

Bug [closed, Priority 5, Epic '']

the clickable library does *not* work inside of figure environments ⇝ yes.
That's fixed; was a bug in hyperref. - I could try to re-implement it without
insdljs. Ideas: - the document catalog's names dictionary needs to
'/JavaScript [(¡arbitrary script name¿) ¡dictionary with JS¿]' entry. The
¡dictionary with JS¿ contains document level javascript. - it is very simple to
generate these entries for my case. Unfortunately, this may be incompatible
with 'insdljs' or other tools which write DLJS. - I am not sure why the floating
figures of TeX produce an incompatibility here. It appears the 'hidden' flag in
the form fields are the problem - if that is the case, I'd need to reimplement
the form annotations (which could be more difficult).

Bug [closed, Priority 3, Epic '']

It is not (properly) possible to provide surf to \addplot.

```
\begin{tikzpicture}
\begin{axis}[]
\addplot[surf,domain=0:720,samples y=25] {cos(x)*sin(y)};%
\end{axis}
\end{tikzpicture}
! Package pgfplots Error: Sorry, you can't use 'y' in this context. PGFPlots expected to s
```

OK, I've been working on it:

- it is now possible to use \addplot[surf] and it works.

- it is *not* yet possible to *sample* matrices with \addplot[surf].

  I added the sample dim key. But it does not work yet... the plot
  expression implementation needs to be refactored.

Bug [closed, Priority 2, Epic '']

One cannot load the clickable lib before pgfplots: see also
https://sourceforge.net/tracker/?func=detail&atid=1060656&aid=
3033981&group_id=224188

Bug [closed, Priority 1, Epic '']

providing ymin=0 for a logarithmic axes has no effect; and there is no sanity
checking

6   Feature Proposals PGFPlots

Feature Proposal   [open, Priority 10, Epic 'Bar plots']

Bar plots suffer from the fact that stacked plots do not use the common serialization routines.

- Change stacked plots to use the standard serialization routines

- Provide access to visualization depends on

- Provide access to point meta (is it possible already?)

- Provide API access to the start point of the stack stuff ("zero level")

This turns out to be a major missing point

Feature Proposal   [open, Priority 9, Epic '']

filled area between 2 addplot's (already requested in mailing list) perhaps style 'fill plot' which is applied in vis phase. There, one can access the postprocessed information of the previous plot. DUPLICATE

Feature Proposal   [open, Priority 9, Epic '']

¿ Is it possible to shade the area between two curves, using pgfplots, such as ¿ in this example:
`http://www.mathworks.com/matlabcentral/fileexchange/13188` ¿ The only shading I could find is between one curve and the x axis... Shading ¿ between curves seems to be possible, but only with stacked curves. Is is ¿ possible to disable stacking somehow, but keep the closedcycle behavior?

DUPLICATE

Feature Proposal   [open, Priority 9, Epic '']

it might be interesting to fill the area between two paths. Perhaps there is such a feature in pgf; or perhaps I can generalize the `\closedcycle` implementation written for stacked plots.

DUPLICATE

IDEAS: the most flexible approach would be to allow multiple `\addplot` instructions in a specific sequence; probably combined with some "reverse sequence"–plot-handler. Perhaps this can be implemented in a similar way like TikZ's path construction things... even if it is much more involved due to the splitting in survey and visualization phase.

`pgfplotstodo.tex:2499` **Feature Proposal** [open, Priority 5, Epic '']

error bars have no access to point meta values or coord index; their draw instructions cannot be modified.

see `http://tex.stackexchange.com/questions/113289/`
`define-error-bar-color-in-scatter-class/113666#comment250975_`
`113666`

`pgfplotstodo.tex:2503` **Feature Proposal** [open, Priority 5, Epic '']

the `...` syntax in tick label position lists is inherently limited to $[-16384, 16384]$ (it inherits the limitations of `\foreach`). Think about alternatives

`pgfplotstodo.tex:2539` **Feature Proposal** [open, Priority 5, Epic 'Bar plots']

It is surprisingly difficult to have JUST axis ticks and tick labels and labels, but NO axis line. This is because I accidentally made `axis x line=none` equivalent to `hide x axis`. Too bad.

Idea: implement keys `axis x line hidden=true,false`. Perhaps with options `axis x line=bottom hidden` which is the same as
`axis x line hidden,axis x line=bottom`?

`pgfplotstodo.tex:2560` **Feature Proposal** [open, Priority 5, Epic '']

Implement document-level javascript for the clickable lib WITHOUT the eforms/insdljs package

should be quite straight-forward. Unless resource-acquisition problems occur (i.e. interoperability issues with other packages)

See

`http://tex.stackexchange.com/questions/3080/`
`what-is-the-best-way-to-insert-document-level-javascript-in-latex-documents?`
`rq=1`

`pgfplotstodo.tex:2566` **Feature Proposal** [open, Priority 5, Epic '']

Implement a custom legend environment such that one doesn't need to collect all options manually

`http://tex.stackexchange.com/questions/54794/using-a-pgfplots-style-legend-in-a-plain-old-`

pgfplotstodo.tex:2577   **Feature Proposal**   [open, Priority 5, Epic 'Bar plots']

Individual bars: allow to modify / adjust the bar plot handler(s) such that each bar can have its individual appearance

- create individual `\path` instructions for every bar

- discard the outer `\path` at the end

- allow simple styles of sorts `bar 1/.style={...}` or `bar value 1.23/.style={...}` perhaps using prefix search? similar to the request for nodes near coords

- should be dependend on point meta (like scatter plots)

pgfplotstodo.tex:2581   **Feature Proposal**   [open, Priority 5, Epic 'Bar plots']

nodes near coords: allow styles of sorts `node near coord 1/.style={...}` or `node near coord value 1.23/.style={...}` (similar to the feature request for bar plots)

pgfplotstodo.tex:2587   **Feature Proposal**   [open, Priority 5, Epic 'Bar plots']

bar plots: auto-select axis limits, unit size, bar width, and bar shift.

Perhaps it is sufficient to auto-select bar width.

pgfplotstodo.tex:2591   **Feature Proposal**   [open, Priority 5, Epic 'Bar plots']

Is it possible to have bar plots which do not start from the x or y axis?. For example a bar plot from (0,2) to (0,3).

pgfplotstodo.tex:2595   **Feature Proposal**   [open, Priority 5, Epic 'Bar plots']

bar plots: provide constant zero level?

pgfplotstodo.tex:2600   **Feature Proposal**   [open, Priority 5, Epic 'Bar plots']

polar axes: polar bar plots (see sourceforge feature request and `http://matplotlib.sourceforge.net/examples/pylab_examples/polar_bar.html`)

Feature Proposal [open, Priority 5, Epic 'Bar plots']

discontinuity in the middle of a plot (as an example see the phase diagram of water `http://pruffle.mit.edu/3.00/Lecture_29_web/img20.gif`)

`http://peltiertech.com/images/2011-11/Ybroken.png`

`http://tex.stackexchange.com/questions/46422/axis-break-in-pgfplots`

Feature Proposal [open, Priority 5, Epic 'Bar plots']

chunked bars: interrupt the bars at predefined coordinates (like white grid lines)

Feature Proposal [open, Priority 5, Epic 'Bar plots']

bar plots:

- bar interval plot handler which *assumes* uniform distances. This allows to eliminate the last, superfluos grid point (because it can be generated automatically as replication xlast + h for known h)

- in fact, I could also implement xlast + hlast and introduce a new name like 'bar interval*' or something like that

Feature Proposal [open, Priority 5, Epic 'Bar plots']

Mails from Stefan Ruhstorfer:

- Gruppierte Säulendiagramme sind nach meinem Wissenstand nur dann möglich wenn man in der Axis-Definiton die Bedindung ybar angibt. Ich finde diese Ausrichtung sehr unflexibel, da ich sehr oft über das Problem stolpere, dass ich in meinem gruppierten Säulendiagramm noch eine waagrechte Linie oder ähnlichs einzeichnen möchte um z.B. meine obere Toleranzgrenze einzuzeichnen. Bis jetzt mache ich das über den normalen draw Modus, was auch ausgezeichnet funktioniert. Jedoch habe ich dann das Problem, dass ich keinen schönen Legendeintrag mehr bekomme. Hier häte ich 2 Vorschläge. Zum einen die Legende "freier" zu gestalten. Also so, dass man beliebig (ggf. auch ohne Plot) ein Legendenelement hinzufüen kann und vllt. noch das zugehörige Symbol festlegen kann. (Bis jetzt habe ich das Problem, das ich mit tricksen zwar meine Obere Tolerangrenze in die Legende bekomme, dann jedoch mit einem Säulenzeichnen davor). Der andere Vorschlag ist, dass Säulendiagramm anders zu definiern. So das ich auch noch einen Plot hinzufügen kann, der mir eine waagrechte Linie ohne zu tricksen einzeichnen lässt.

- Eine Gruppierung von stacked bars ist nach meinem Wissen nicht möglich. Es ist zwar schwer sich ein Anwendungsgebiet dafür vorzustellen, aber wenn sie danach mal suchen (speziell im Excelbereich) werden sie sehen, dass viele Leute so eine Funktion benutzen. ⤳ siehe auch folgemails mit Beispielskizzen ⤳ beachte: Fall 2.) erfordert mehr arbeit als lediglich 'line legend', weil ybar ja den koordinatenindex verarbeitet!

pgfplotstodo.tex:2641   **Feature Proposal**   [open, Priority 5, Epic 'Bar plots']

let stacked bar plots respect the line width when stacking them on top of each other

pgfplotstodo.tex:2648   **Feature Proposal**   [open, Priority 5, Epic 'Bar plots']

Allow to filter out stacked plot elements (for example those in which the value "0" occurs)

see `http://tex.stackexchange.com/questions/102798/` `for-stacked-bar-chart-how-to-render-zero-height-bars-as-invisible`

pgfplotstodo.tex:2660   **Feature Proposal**   [open, Priority 5, Epic 'Bar plots']

On input: provide the desired distance between adjacent bars. The bar width should be automatically determined in a way which shows all bars.

An extreme case would degenerate to ycomb.

If the plot area does is insufficient, bar distances should be restricted (?)

Note: if (and only if) the bars have unit distance 1, a relative distance is already supported by providing bar width=0.9 or something like that...

Note: if we knew the distance between adjacent ticks, and we assume that each tick has its associated bar plot, we could define the bar width in terms of the tick distance... perhaps I should expose read-only access to the list of (computed) tick positions; with a convenience method to access the common distance? Or just a common distance?

pgfplotstodo.tex:2666   **Feature Proposal**   [open, Priority 5, Epic 'Bar plots']

bar plots: if you provide both, the desired distance between adjacent bars and the bar width, the axis limits should be increased or decreased automatically.

This is actually a more generic concept: it would be 'scale mode=limits only' (?)

pgfplotstodo.tex:2670   **Feature Proposal**   [open, Priority 5, Epic 'Bar plots']

Default bar plot styles should always include the (correct) origin in the visible axis range.

pgfplotstodo.tex:2676   **Feature Proposal**   [open, Priority 5, Epic 'Bar plots']

It should be simpler to customize the position of nodes near coords in a way to position them in the middle or below each bar.

This is particularly difficult for stacked bar plots

pgfplotstodo.tex:2694   **Feature Proposal**   [open, Priority 5, Epic '']

simplify tufte-style plots: modify the `node[at=<pos>]` feature such that `node[at value=42.4]` or `at max value` or `at min value`

pgfplotstodo.tex:2701   **Feature Proposal**   [open, Priority 5, Epic '']

Layered graphics: consider drawing tick lines which are on the "outer part" of the axis on the foreground layer.

See
`http://tex.stackexchange.com/questions/31708/draw-a-bivariate-normal-distribution-in-tikz/`
for a motivation (the tick lines are hidden by the surface)

pgfplotstodo.tex:2705   **Feature Proposal**   [open, Priority 5, Epic '']

filled contour plots (prototype is 10% ready)

pgfplotstodo.tex:2709   **Feature Proposal**   [open, Priority 5, Epic '']

allow support for units in `bar width` and `bar shift` (compare the implementation for circles/ellipses)

pgfplotstodo.tex:2721   **Feature Proposal**   [open, Priority 5, Epic '']

provide log labels without exponents, i.e. 10000 instead of 10^4

pgfplotstodo.tex:2726   **Feature Proposal**   [open, Priority 5, Epic '']

it would be nice to have automatic PNG export for huge graphics. Such an approach, combined with plot graphics, could result in considerably smaller pdfs and faster rendering. At the same time, it would not suffer the limitation which arises if one uses the external lib and converts the complete figure to png (including axis descritpions)

pgfplotstodo.tex:2736   **Feature Proposal**   [open, Priority 5, Epic ‘’]

There is no simple way to provide LOG colorbars:

1. ymode=log is not supported in ‘every colorbar’ due to key filtering problems

2. disablelogfilter appears to be useless and does not respect ‘log basis’

If those two this would be fixed, one could provide
`colorbar style={ymode=log,disablelogfilter}` and would get a proper logarithmic colorbar. Perhaps even combined with `log basis` ... ?

pgfplotstodo.tex:2740   **Feature Proposal**   [open, Priority 5, Epic ‘’]

Cases-statement in math parser

pgfplotstodo.tex:2748   **Feature Proposal**   [open, Priority 5, Epic ‘’]

provide a way to provide more customization to stacked plots as in

`http://tex.stackexchange.com/questions/13627/pgfplots-multiple-shifted-stacked-plots-in-on`

(stacked and clustered bar charts)

pgfplotstodo.tex:2752   **Feature Proposal**   [open, Priority 5, Epic ‘’]

the `empty line` feature should produce a log notice when it finds an empty line in compat mode.

pgfplotstodo.tex:2760   **Feature Proposal**   [open, Priority 5, Epic ‘’]

Support something like
`’\addplot table[x symbolic expr={\thisrow{year}-\thisrow{month}-\thisrow{day}}]’`.

pgfplotstodo.tex:2764   **Feature Proposal**   [open, Priority 5, Epic ‘’]

What about a ‘draft’ mode which does nothing but typeset an empty axis without descriptions?

pgfplotstodo.tex:2772   **Feature Proposal**   [open, Priority 5, Epic ‘’]

Provide features of an axis *outside* of the axis environment. For a start, this could use the `axis cs` (or an alias to it).

Details and examples:

https://sourceforge.net/tracker/?func=detail&atid=1060659&aid=
3086794&group_id=224188

pgfplotstodo.tex:2776 **Feature Proposal** [open, Priority 5, Epic '']

add 'force 2d axis' key (or similar)

pgfplotstodo.tex:2786 **Feature Proposal** [open, Priority 5, Epic '']

could you extend the /tikz/prefix key so it also works as a prefix for imported
files/tables? So far one has to type for example
 \addplot table {plots/data/test.txt};

If there would be a search path like \graphicspath for graphics it would be
really nice.

See also `https://sourceforge.net/tracker/?func=detail&atid=`
`1060659&aid=3020246&group_id=224188`

pgfplotstodo.tex:2794 **Feature Proposal** [open, Priority 5, Epic '']

Support standard filters for `hist` and its variants.

Improve filtering for `hist` and similar plot handlers.

I already added the `hist/data filter` and `pre filter` keys
(undocumented!). Use them.

pgfplotstodo.tex:2799 **Feature Proposal** [open, Priority 5, Epic '']

the 'xtick' syntax accepts only numbers, not even constant expressions are
possible (and 'pi' is even more complicated).

pgfplotstodo.tex:2804 **Feature Proposal** [open, Priority 5, Epic '']

Table Package: support context–based `row predicate`s (some kind of
WHERE clauses)

pgfplotstodo.tex:2808 **Feature Proposal** [open, Priority 5, Epic '']

support the `/data point/x` method for all key filters and in all contexts (i.e.
in the same context where \thisrow is accepted)

pgfplotstodo.tex:2812   **Feature Proposal**   [open, Priority 5, Epic '']

Support selection of individual 3D axis lines which shall be drawn (or "floor")

pgfplotstodo.tex:2816   **Feature Proposal**   [open, Priority 5, Epic '']

Support custom unit vectors for 3D axes

pgfplotstodo.tex:2821   **Feature Proposal**   [open, Priority 5, Epic '']

implement properly layered graphics — especially for grid lines should
probably also respect multiple ordinates

pgfplotstodo.tex:2825   **Feature Proposal**   [open, Priority 5, Epic '']

linear regression which passes through (0,0) (see mail of Stefan Pinnow)

pgfplotstodo.tex:2830   **Feature Proposal**   [open, Priority 5, Epic '']

plot graphics 3D: handle the case when the first two points share the same x
(or y) coordinate

pgfplotstodo.tex:2834   **Feature Proposal**   [open, Priority 5, Epic '']

hist does not allow modifications to the data range

pgfplotstodo.tex:2859   **Feature Proposal**   [open, Priority 5, Epic '']

see the interesting things at
`http://peltiertech.com/Excel/Charts/axes.html#Broken` broken (y) axis:
remove interval [a,b] idea: if y¡a : visualize as usual if a¡y¡b : use coordinate
y=a if b¡y : use coordinate y=y-(b-a) axis:

- compute two sets of axis descriptions. Perhaps one can try to compute
  the step size just once, and discard only [a,b] afterwards? This would
  require to use a canvas axis length corresponding to the unremoved axis
  range. BTW: I need access to the unremoved axis range; both for tick
  computation and for 'nodes near coords' or the clickable lib.

- draw a decoration at the break.

- perhaps also a decoration near affected coords.

- perhaps I should apply the thing during the visualization phase, not
  before. Then, I have all limits and the correct coordinates; only canvas
  coords are affected.

`pgfplotstodo.tex:2868` **Feature Proposal**   [open, Priority 5, Epic '']

feature to replicate axis descriptions on both sides

`pgfplotstodo.tex:2875` **Feature Proposal**   [open, Priority 5, Epic '']

couldn't you add something like `\providecommand*\pgfplotsset[1]{}` to the
"tikzexternal.sty" so one doesn't have to do it by hand when switching from
tikz/pgfplots?

`pgfplotstodo.tex:2879` **Feature Proposal**   [open, Priority 5, Epic '']

ternary diagram for extractions (more details will come)

`pgfplotstodo.tex:2891` **Feature Proposal**   [open, Priority 5, Epic '']

make work `\matrix in \matrix` so one can use groupplots or "Allignment in
Array Form" (section 4.18.4) with legends

`pgfplotstodo.tex:2909` **Feature Proposal**   [open, Priority 5, Epic '']

nested axes would be a nice feature. TODO: - update the list of global state
variables - "interrupt" these variables somehow. - make sure local redefinitions
of TikZ commands (like point commands) work; the `\let...@orig=`
assignments should be handled somehow. - What about keys? They will be
inherited from the outer axis... perhaps the best would be an


```
\endgroup
<nested axis>
\begingroup
<restore state>
```


which includes the keys of the outer axis!?

`pgfplotstodo.tex:2919` **Feature Proposal**   [open, Priority 5, Epic '']

groupplots: group-wide axis labels

`pgfplotstodo.tex:2937` **Feature Proposal**   [open, Priority 5, Epic '']

is there a way to get the current row/col index during addplot?

Feature Proposal   [open, Priority 5, Epic '']

plot
shell: - It would be nice if the standard shell interpreter could be replaced. Idea:
`\pgfkeys{/pgfplots/plot shell/interpreter/.code 2 args={sh #1 > #2}}`
then in the code
`\pgfkeysvalueof{/pgfplots/plot shell/interpreter/.@cmd}{#1.sh}{#1.out}\pgfeov`
- the pgfshell macro is quite general and could be added to pgf (as suggested
by you, Stefan). However, this would also need modifications in tikz.code.tex
to get some sort of high-level user interface. I find plot shell very useful, and it
could be added easily. My suggestion: Either write a high level user interface
for tikz or rename the command to pgfplotsshell and put it into
pgfplotscoordprocessing.code.tex. In the meantime, I added it to
pgfplotscoordprocessing.code.tex (bottom). - there is a potential difficulty
with the 'addplot table shell' command (which is a good solution!): the
semicolon in this routine will have a fixed catcode. But packages like babel
with french language will change it to active, so french people can't use
addplot table shell. The solution is technical and I am not proude of my own
anyway... we'll just have to think about one. - documentation for the 'table
shell' feature is missing yet. - I am not sure if the replication of /tikz/prefix
and /tikz/id is helpful or confusing....

Feature Proposal   [open, Priority 5, Epic '']

new `\plotnumofactualtype` thing: if you set /tikz/ plot handlers in
`\begin{axis}`, they won't be set before the visualization phase. consequently,
I can't count them! Idea: add a 'family' to each of them. Or wright a coord
filter which checks for `\tikz@plot@handler` . Or write pgfplots styles which
set them.

Feature Proposal   [open, Priority 5, Epic '']

feature request for line styles in tikz/pgf or pgfplots respectively: add
dash-dotted line which is quite commen in engineering field for example
something like

```
\tikzset{
dash-dot/.style={
dash pattern=on 4pt off 3pt on 1pt off 3pt,
},
}
```

Feature Proposal   [open, Priority 5, Epic '']

disable bounding box updated during addplot – it makes no sense and wastes
time (unless the axis is hidden)

76

Feature Proposal [open, Priority 5, Epic '']

polar:

- is my current datascaling approach correct? I mean, is the linear trafo feasible at all?

- the *affine* radius datascaletrafo could be enabled, if only parts of the circle are drawn at all, for example xmin=0,xmax=45, ymin=1e-4,ymax=1.003e-4 Idea: check arc size and disable the radius *affine* data scaling only if the arc has more than 90 (?) degrees Is that mathematically correct? And: is it useful at all?

- handle "empty axis". It should reset to a circle, not a box.

Feature Proposal [open, Priority 5, Epic '']

patch visualization: provide displacement input format

Feature Proposal [open, Priority 5, Epic '']

the following keys should process their argument with pgfmathparse:

-

 xyz tick,

- min/max

- tickmin/max

- meta min/max

- domain/ y domain,

- error bar arguments,

- without FPU: width/height/ view

- check optimizations of the math parser!

- check if I can activate the FPU during the survey phase!

Feature Proposal [open, Priority 5, Epic '']

add polar coordinates

77

Feature Proposal   [open, Priority 5, Epic '']

Idea for input stuff: implement high level user interface for coordinate input, similar to the pgf basic level framework. Then, add styles on top of it (try to be compatible with DV engine)

Feature Proposal   [open, Priority 5, Epic '']

Idea: implement an automatic /pgf/number format setting which determines a suitable representation for a *set* of numbers. For example, 1e-17 0.2 0.4 0.8 should be printed as 0 0.2 0.4 0.8 whereas 1e-17 2e-17 3e-17 should be printed using the scientific range (perhaps even using some sort of scaling as for ticks). This would be useful for contour plot labels as well. ⤳ a realization should check the data range (especially its exponent). Thus, I want a *relative* number printing style.

Feature Proposal   [open, Priority 5, Epic '']

new plot structure : use the '/data point' key interface coming with pgf CVS

Feature Proposal   [open, Priority 5, Epic '']

new structure for math operations:

- aim: interface for math operations which works independent of lowlevel repr

- ¿ FPU vs basic pgf vs LUA vs 'fp.sty' vs ....

- ¿ log axes can be done in pgf (faster)

- necessary: high level `\pgfmathparse` *and* mid level invocation of operations

- necessary: parsenumber, tofixed, tostring

- datascaling needs access to exponents and base 10 shifts

- necessary: check for nan and inf

- necessary: the max/min routines which are no longer supported by pgf (the `\pgfplotsmath...` routines)

interface:

- transparent exchange of math mode routines

- fast (enough)

- for each axis separately (optimized for log)

78

- variable number of arguments

- expansion of arguments should be possible

- the interface is necessary for *coordinate* arithmetics, not necessarily for the pgf interaction (can keep register math)

realization ideas:

- command suffix for each axis '@basic' versus 'float'

- central interface to invoke math ops:
  `\pgfplotscoordmath{x}{multiply}{{<arga>}{<argb>}}` Idea: use `\edef` on the arguments.

- provide `\pgfplotssetmathmode{x}{<suffix>}` should assert that the desired interface is complete

- `\pgfmathparse` may need to be adjusted if it uses a different output format than ¡suffix¿

TODO:

- rethink data scaling transformation. Should it be done as "coord math"?

- handling of depth searching needs to be implemented with "default" coordmath

- the log routines -⤳ also use it for table package. BUGGY! compare examples in manual. Minor log ticks don't work at all, default log tick labels are simply wrong.

- disablelogfilter case

- ✓error bars work with both, float and log

- plothandlers.code.tex

- prepare@ZERO@coords

pgfplotstodo.tex:3228   **Feature Proposal**   [open, Priority 5, Epic '']

rewrite the read number routines. They should allow 'disableddatafilter' thing during addplot.

`pgfplotstodo.tex:3256` **Feature Proposal** [open, Priority 5, Epic '']

quiver plots:

- allow to disable update of axis limits

- provide rescaling of arrows such that they don't overlap. manual rescaling is simple, auto is more difficult. auto: if I have a matrix, I could rescale such that its mesh width is larger than the largest vector. Same fo a vector of input data. But what if I don't know whether it's a vector or matrix? ⇝ second run. ⇝ after the first, it should be possible to autocomplete the mesh rows/cols. Try it. If that works, we have a matrix. ⇝ could be done from within the scanlinelength routines: auto-detect mesh/rows mesh/cols mesh/ordering mesh/width but that fails if there is no scanline marker.

- what with log plots? What with other axis features like symbolic trafos? ⇝ need difference type!

- that is: quiver plots in log coords are *multiplicative* and invoke the same routines. make special handling for '0'.

- allow feature where (u,v) are *coords*, not vectors. this could allow additive log quiver plots.

`pgfplotstodo.tex:3260` **Feature Proposal** [open, Priority 5, Epic '']

plot expression: make the sampling parameters available within survey phase

`pgfplotstodo.tex:3264` **Feature Proposal** [open, Priority 5, Epic '']

the table package uses a lot of logs – but it can't change the log basis.

`pgfplotstodo.tex:3269` **Feature Proposal** [open, Priority 5, Epic '']

3D + axis line variants: someone might prefer GRID LINES as for the boxed case combined with axis line=left...

`pgfplotstodo.tex:3313` **Feature Proposal** [open, Priority 5, Epic '']

Mail by Hubertus Bromberger:

- ✓Period in legend, without the need of using the math environment? `\legend{ML spcm$.$, CW spcm$.$, ML AC};`

- Maybe a more straight forward way for legend to implement something like shown in the graph. (see his mail .tex) ⇝ plot marks only at specific points. thus, the legend image should contain both lines and marks, but there are effectively two addplot commands.

- As a physicist, I often have the problem to fit curves. A job gnuplot can do very well. It should be possible using "raw gnuplot" but maybe you can either provide an example or even implement a more straight forward way for this purpose.

- The color scheme is not really my taste. In CONTEXT:

```
 cycle list={%
{Col1,mark=*},
{Col2,mark=square*},
{Col3,mark=diamond*},
{Col4,mark=star},
{Col5,mark=pentagon*},
{Col6,mark=square*},
{Col7,mark=diamond*},
{Col8,mark=triangle*} }}
   \definecolor[Col1][r=0.24106,g=0.05490,b=0.90588]  % blau
   \definecolor[Col2][r=1,g=0.05490,b=0.06667]        % rot
   \definecolor[Col3][r=0.65490,g=0.73333,b=0.01176]  % grn
   \definecolor[Col4][r=0.08627,g=0.92549,b=0.91373]  % tyrkis
   \definecolor[Col5][r=1,g=0.5,b=0]                  % orange
   \definecolor[Col6][r=0.54118,g=0.51765,b=0.51765]  % grau
   \definecolor[Col7][r=0.80784,g=0.49804,b=0.06275]  % okker
   \definecolor[Col8][r=0.74902,g=0.07451,b=0.91765]  % lila
```

- Sometimes it would be good to have a bit more of a programming language, but still that's not what tex is made for. The python-script looks promising, it's just, that I think it doesn't work with context.

pgfplotstodo.tex:3322   **Feature Proposal**   [open, Priority 5, Epic '']

add something like

```
\pgfplotstabletypeset[
  cell { 1 }{ 2 }={\multirow{*}{3}{text}}
]
```

pgfplotstodo.tex:3366   **Feature Proposal**   [open, Priority 5, Epic '']

ternary diagrams todo:

- the `\pgfplotsqpointoutsideofaxis` work only for position 1, nothing in-between (since it doesn't compute the other axis components correctly)

- data ranges are currently only correct if in [0,1] or if one provides the [xyz]min and [xyz]max keys (and the ternary limits relative=false). How should it work!?

pgfplotstodo.tex:3383   **Feature Proposal**   [open, Priority 5, Epic '']

contour:

- labels=true,false,auto ⤳ auto should deactivate labels if there are too many contour lines.

- labels should not be clipped...

- add label position shifting facilities. ⤳ identify by contour label *and* an optional index. There may be more than one line.

pgfplotstodo.tex:3391   **Feature Proposal**   [open, Priority 5, Epic '']

contourf: I guess filled contour plots could be possible if always two adjacent color levels are combined into a single path which is then filled with the simplified even/odd rule (not the winding fill rule). With the underlying smoothness assumption C^0, there can't be any level between two adjacent ones, and there can't be self-intersections.

pgfplotstodo.tex:3396   **Feature Proposal**   [open, Priority 5, Epic '']

it would be very interesting to allow more flexible handling of empty lines in input data, especially files.

pgfplotstodo.tex:3415   **Feature Proposal**   [open, Priority 5, Epic '']

contour draft TODO:

- color of text nodes

- make sure there is at least one label node

- implement contourf

  - often: use 'even odd rule' to fill adjacent contours.
  - but this works only if adjacent contours are contained in each other.
  - if that's not the case, perhaps I need to add an artifical path from the data limits.
  - idea: in case I know the corner values, I'd know which contour plateau requires the artifical path.
  - other idea: I could implement some sort of even-odd rule in TeX. This should also yield the information.

pgfplotstodo.tex:3421 **Feature Proposal** [open, Priority 5, Epic '']

implement simplified constructions to access DIFFERENCE coordinates. For example, \draw ellipse needs x radius and y radius.

pgfplotstodo.tex:3442 **Feature Proposal** [open, Priority 5, Epic '']

the 'table/y index' should be changed. It should be min(numcols,1) instead of 1.

pgfplotstodo.tex:3453 **Feature Proposal** [open, Priority 5, Epic '']

table package and axes should improve their communication. Namely:

-
- communicate table names.
- communicate xmode/ymode
- communicate log basis [xy]

pgfplotstodo.tex:3458 **Feature Proposal** [open, Priority 5, Epic '']

provide and document access to (sanitized?) mesh/rows and mesh/cols fields during the survey phase. This might allow 2d key filters

pgfplotstodo.tex:3465 **Feature Proposal** [open, Priority 5, Epic '']

Praktisch fände ich, wenn man folgende Dinge spezifizieren kann: 1. Welche Zeilen aus der Datei ausgelesen sollen (häufig gibt es nicht nur 1, sondern mehrere Header-Zeilen, oder auch am Ende noch sonstige Zeilen)

pgfplotstodo.tex:3492 **Feature Proposal** [open, Priority 5, Epic '']

improve support for multiple ordinates

1.  ```
    * \pgfplotsset{set layers}
    * scale only axis
    * xmin=..., xmax=...,
    * axis y line*=left
    * axis y line*=right
    * axis x line=none
    ```

    would be hidden in the doubleaxis definition,

2. the first addplot would be the left one and the second, the right one, (???)

83

3. the comma separated list in the legend command's argument applies
   successively to the two addplot.

4. the colours of the two plots are given by the color cycle list.

Feature Proposal   [open, Priority 5, Epic '']

it would be useful if the clipping could be disabled for certain parts of the axis.
Is that possible?

- yes. Idea: start clipping for every axis element separately! Shouldn't be
  much more expensive than a single marker path.

- should work in the same way as before, there is no difference!

- scopes should introduce no further problems

- I could eliminate the nasty marker list

Feature Proposal   [open, Priority 5, Epic '']

provide a `\pgfplotspathcube` command as generalization from the cube
marker. The cube command should work similar to pathrectangle or
rectanglecorners.

Feature Proposal   [open, Priority 5, Epic '']

re-implement sampling loops. I should discard the compatibility with foreach
internally in order to gain accuracy! Maybe it is necessary to invoke different
loops - one for tikz foreach (samples at) and one "standard" sampling routine.

Feature Proposal   [open, Priority 5, Epic '']

optimization ideas:

- replace `\pgfpointscale` with a 'q' version $\rightsquigarrow$ it invokes the expensive
  math parser.

- pgfmultipartnode evaluates every anchor twice

- implement a cache for expensive, repeated math operations like 'view'
  directions or common results of $1/||e_i||$.

- search for unnecessary math parser invocations; replace with 'q' versions
  if possible.

- implement a hierarchical generalization of the 'applist' container (a tree
  applist of arbitrary length)

84

- eliminate the deprecated 'non-legend-option' processing.

- remove the different (empty) paths of the axis node – it appears they are not necessary and waste only time and mem.

- try implementing an abstract 'serialize' and 'unserialize' method - it might be faster to re-process input streams instead of generating preprocessed coordinate lists.

- try to reduce invocations of pgfkeys

- optimize the filtered pgfkeys invocations - the filter is slower than necessary!

- the plot mark code invokes a lot of math parsing routines - which is a waste of time in my opinion. All expressions etc. have already been parsed.

- the point meta transform is set up twice for scatter plots.

- my elementary data structures always use `\string` to support macros as data structure names. I fear this might be ineffective. Perhaps its better to check if the argument is a macro (at creation time, thus only once) and call `\edef#1{\string#1}` to assign some sort of name to it. This will invoke `\string` only once. Is this faster?

- eliminate the 'veclength' invocations for single axes - they can be replaced with "inverse unit length * (max-min)"

- the key setting things can be optimized with pgfkeysdef

- create the /pgfplots/.unknown handler (.search also=/tikz) once and remember it.

- the (new) tick label code might be very expensive:

  - check for (unnecessary) calls to `\pgfpointnormalised` – the normal vectors are already normalised!

  - check the cost for bounding box size control of the tick labels – maybe this can be optimized away if it is not used. But this decision is not easy.

pgfplotstodo.tex:3565   **Feature Proposal**   [open, Priority 5, Epic '']

perhaps math style
`{grid=major, axis x line=middle, axis y line=center, tick align=outside}`

pgfplotstodo.tex:3569   **Feature Proposal**   [open, Priority 5, Epic '']

asymmetric error bars

`pgfplotstodo.tex:3574` **Feature Proposal** [open, Priority 5, Epic '']

provide access to axis limits and data bounding box. It would be useful to get access to axis coordinates, for example in 'circle (XXX)'

`pgfplotstodo.tex:3579` **Feature Proposal** [open, Priority 5, Epic '']

allow math expressions for axis limits etc. Idea: try float parsing routine; if it fails: use math parser first.

`pgfplotstodo.tex:3586` **Feature Proposal** [open, Priority 5, Epic '']

write a public math interface which provides access to axis internals like limits, the 'dimen-to-coordinate' method and so on. ⤳ it might be useful to use pgfmathparse for any numerical input argument as well.

`pgfplotstodo.tex:3622` **Feature Proposal** [open, Priority 5, Epic '']

Store the axis limits into the axis' node as saved macros. This would allow

- 'use [xy] limits of=¡axis name¿'

- access to axis limits from other macros.

- provide a command `\pgfplotslimits{current axis}{x}{min}` which expands to the 'xmin' limit. PROBLEM: to WHICH limit: the untransformed one? The transformed one? The logarithmized one?

  - ¿ I can't compute exp(xmin) in log plots!
  - Ideas:
  - provide both, if possible. It is NOT possible for log axes.
  - use log-limits (possibly combined with 'logxmin=' option ?)
  - The operation requires several operations because floats need to be converted. Idea: do that only for NAMED AXES.
  - all user-interface macros must be expandable!
  - I don't want to spent time for number format conversions unnecessarily here!
  - provide `\pgfplotslimits` and `\pgfplotstransformedlimits` combined with simpler key-value interfaces
  - I could also provide access to the unit lengths (they are available as macro anyway)
  - ALTERNATIVE: implement access to axis limits as a math function which simply defines `\pgfmathresult`.
  - that is probably the most efficient way to do it. I only need to register the new function(s) to PGF MATH.

- PGF 2.00: use
  `\csname pgfmath@parsefunction@\pgfmath@parsedfunctionname\endcsname`
- PGF ¿ 2.00: use `\pgfmathdeclarefunction` Is it possible to
  provide 'string' arguments which are not parsed? No.

pgfplotstodo.tex:3628   **Feature Proposal**   [open, Priority 5, Epic '']

I could provide public macros for the data transformations (and inverse
transformations). This would also allow relatively simple access to axis limits.

pgfplotstodo.tex:3632   **Feature Proposal**   [open, Priority 5, Epic '']

cycle list should be implemented using an array structure. That's faster.

pgfplotstodo.tex:3638   **Feature Proposal**   [open, Priority 5, Epic '']

what about a feature like 'draw[xmin=...,xmax=...] fitline between points (a)
(b)'?

pgfplotstodo.tex:3642   **Feature Proposal**   [open, Priority 5, Epic '']

interpolate missing coordinates for stacked plots.

pgfplotstodo.tex:3648   **Feature Proposal**   [open, Priority 5, Epic '']

the error bar implementation is relatively inefficient. Think about something
like '/pgfplots/error bars/prepare drawing' which sets common style keys for
every error bar

pgfplotstodo.tex:3678   **Feature Proposal**   [open, Priority 5, Epic '']

think about using a combination of the visualization engine of pgf CVS and
my prepared-list-structure. Maybe I can adjust the list format for the current
plot type? I need

- scatter/line plots 2D

- meta coords

- quiver may need extra vectors

- matrix plots may need twodimensional structure

- error bars could be handled more consistently

- ...

- ¿ implement a visualization class which provides methods

  – prepare()
  – visualize()
  – serialize()
  – visualizestream() and provide protected pgfplots methods
  – axis⤳preprocesscoordinate (filters, logs)
  – visualizer⤳prepare()
  – axis⤳processcoordinate()
  – visualizer⤳serialize()
  – axis⤳postprocesscoordinate() The markers as they are
    implememted now don't really fit into this framework. The clipping
    region is not really what I want here... Idea: enable/disable clipping
    separately for each drawing command!

Feature Proposal   [open, Priority 5, Epic '']

the coordindex shouldn't be changed by z buffer=sort

Feature Proposal   [open, Priority 5, Epic '']

table package: provide abstract layer for low level storage interface. Idea: the
interface should allow the container interface

- push_back()
- get(i)
- set(i)
- foreach()
- pop_front()
- newempty()
- clone()
- unscope()
- startPushBackSequence()
- stopPushBackSequence()

⤳ this could allow to use arrays for fast algorithms. At least it would make
things easier to read. Problem as always: the 'unscope()' operation. Currently,
I have two different structures: the applists which have fast construction
properties and the standard lists which implement the rest. Can I combine
both? Yes, by means of the incremental construction pattern:

```
\startPushBackSequence
\push_back
\push_back
\push_back
\stopPushBackSequence
```

⤳ inside of the construction, only `\push_back` is allowed and the structure is in "locked state" (low level: applist repr) ⤳ Idea: the creation is fast, afterwards, it has flexibility.

Feature Proposal   [open, Priority 5, Epic '']

It is certainly possible to write some sort of CELL-BASED 'mesh/surf' shader - a combination of 'flat corner' and cell based rectangles:

- every coordinate denotes a CELL instead of a corner,

- the "shader" maps the cdata into the colormap to determine the cell color

- details?

    - to get well-defined cells, I have to enforce either a non-parametric lattice grid or do a LOT of additional operations (?).

    - alternative: define N*M cells by N+1 * M+1 points.

    - perhaps a combination of both? ⤳ that's more or less the same as 'flat mean' up to the further row/column pair

- it would be generally useful to have an "interval" or "cell" mode: the idea is that every input coordinate defines an interval (1d) or a cell (2d). To define the last cell, one needs to add one "mesh width" somehow. I just don't know where:

    - the artificial cell should be processed with the normal streams - including limit updates, stacking etc.

    - the artificial cell needs to know when the end-of-stream occurs. For 1d plots, that may be possible. For 2D plots, this information requires a valid 'cols' key.

    - I suppose it would be best to patch @stream@coord.. at least for the 'cell' mode.

    - Idea:

        * the `\pgfplots@coord@stream@coord` implementation realizes the cell-mode: after every 'cols' coordinate, a further one is replicated. This needs the "last mesh width". Furthermore, it needs to accumulate a row vector, the "last row". This last row is need during stream@end to replicate the further row:

* the `\pgfplots@coord@stream@end` implementation has to realize the last step of cell mode: the replication of a further row. It also has to realize the implementation of 'interval' mode (replication of last coordinate). My idea is to simply use an applist for this row accumulation. The format should be compatible with `\pgfplots@coord@stream@foreach@NORMALIZED`. That doesn't produce problems, even when the end command is invoked within a foreach@NORMALIZED loop - because the loop has already ended.

pgfplotstodo.tex:3771   **Feature Proposal**   [open, Priority 5, Epic '']

support `\multicolumn` for legends

pgfplotstodo.tex:3776   **Feature Proposal**   [open, Priority 5, Epic '']

it appears line breaks in legend descriptions are a problem (?) ⤳ bug in pgf: `\\` is overwritten and won't be restored.

pgfplotstodo.tex:3784   **Feature Proposal**   [open, Priority 5, Epic '']

pgfplotstable file open protocol: provide public listener interface

pgfplotstodo.tex:3788   **Feature Proposal**   [open, Priority 5, Epic '']

`\addplot coordinates {\macro};`

pgfplotstodo.tex:3812   **Feature Proposal**   [open, Priority 5, Epic '']

precise width calculation idea:

- Problem: total width depends on width of axis descriptions
- width of axis descriptions depends on position of axis descriptions
- position of axis descriptions depends on width of axis
- width of axis depends on width of axis descriptions
- non-linearly coupled system.
- Idea: introduce a loop.
    - details:
        1. place axis descriptions + the axis rectangle into a box.

2. Measure box'es width, throw it away if it is too bad. Keep it and stop iteration otherwise.
3. recompute the complete scaling.
4. go back to step 1.) and iterate

– one or two iterations should be enough .

– it's not necessary to recompute the prepared and stored plots. Just keep them in main memory until the scaling is fixed.

Feature Proposal   [open, Priority 1, Epic '']

log plots: minor tick num would be useful here! If tick labels are placed at '1e-5, 1e0', minor tick num= 4 would lead to the minor tick lines at '1e-4,1e-3,1e-2,1e-1' which is useful. So:allow minor tick num for log axes. ⇝ need to adjust the check for "uniform log ticks"

Feature Proposal   [cancelled, Priority 5, Epic '']

idea: 'mesh/ordering=auto'. Just check for 'x varies' and 'y varies'! The two first points inside of a scanline are enough.

Feature Proposal   [closed, Priority 5, Epic '']

plot surface plots with explicit colors, not colormap (see http://tex.stackexchange.com/questions/97523/ pgfplots-color-a-3d-surf-using-arbitrary-rgb-colors)

First prototype is up-and-running. Todo:

- ✓shader=flat corner

- ✓shader=interp (tested for shading type 4)

- ✓shader=flat

- ✓patch plots lib and its refinement strategies

- ✓improve input syntax

  - ✓accept both xcolor and normalized RGB / CMYK expressions

  - ✓accept only normalized expressions – without colorsspace if the color space is fixed in advance

  - ✓accept math expressions for individual components which map to 0,1

- ✓communicate auto-detected input colorspace to the shader. Hm; ok, it could also lazily use the first encountered one and assume that all have the same. might work.

91

- ✓BUG : providing color=black does not work! result in just one component. Wrong colorspace

- BUG : combination with refine + faceted interp

- docs:
  - color input
  - new freedom for colormap definitions: more input + output colorspaces
  - `point meta/symbolic={x,y,1}`

- tests: changed colormap stuff requires unit tests

pgfplotstodo.tex:2550   **Feature Proposal**   [closed, Priority 5, Epic '']

contour plots in any other direction than z is unnecessarily difficult.

if you want a contour in direction x, you can easily modify the gnuplot script to

```
unset surface; set cntrparam levels 10; set contour; splot "contour3d_contourtmp0.dat" usi
```

the same should be possible with pgfplots - either by telling gnuplot what to do or by generating the tmp input file accordingly.

pgfplotstodo.tex:2717   **Feature Proposal**   [closed, Priority 5, Epic '']

improve support for circle / ellipse paths inside of an axis

compare `http://www.digipedia.pl/usenet/thread/16719/198`

```
http://sourceforge.net/mailarchive/forum.php?thread_name=
D595FD68-AFAB-4C1C-8B9D-A2F84D1A0598%40mac.com&forum_name=
pgfplots-features
```

pgfplotstodo.tex:2756   **Feature Proposal**   [closed, Priority 5, Epic '']

smith charts: provide the same as now, but mirrored (concentric from left end rather then right end)

pgfplotstodo.tex:2864   **Feature Proposal**   [closed, Priority 5, Epic '']

plot graphics for 3D axes.

pgfplotstodo.tex:2915 **Feature Proposal** [closed, Priority 5, Epic '']

support for "spy"glass into particular parts of an axis

appears to work correctly!?

pgfplotstodo.tex:2923 **Feature Proposal** [closed, Priority 5, Epic '']

It would be really great to have the possibility to attach a style to every nth row of a data table. For example, I would like to have a `\midrule` not after every line or after odd/even lines but after every fifth (or whatever) line.

pgfplotstodo.tex:3066 **Feature Proposal** [closed, Priority 5, Epic '']

Konnodalplots fuer Ternary Axes

given: pairs of points (A_i, B_i) with $A_i, B_i \in R^3$ for the connodals

aim: connect $A_i - -B_i$ for each i *and* create the binodal line
$A_1 - -A_2 - - \cdots A_n - -B_n - -B_{n-1} - - \cdots B_1$

Remarks of stefan:

Im Anhang ist ein Beispiel gezeigt, wie es gehen könnte.

Noch einmal zur Klärung der Begriffe, mit denen ich gleich argumentieren werde:

- Binodale: Kurve

- Konode(n): Gerade(n) [engl.: tie line]

- Kritischer Entmischungspunkt: Ist der Punkt, an dem die beiden Punkte der Konode zusammenfallen. (nicht eingezeichnet)

- Mischungslücke: Das Gebiet, was von der Binodalen eingeschlossen wird. [engl.: miscibility gap]

Im Anhang findest du zum Einen die Daten-Datei und zwei mögliche Darstellungsformen. Das "`gibbs_phase_diagram`" ist die Darstellung im Dreieckdiagram (was auch Gibbs'sches Phasendiagramm oder Gibbs'sches Phasendreieck genannt wird); "`cartesian_phase_diagram`" entsprechend im Kartesischen Phasendiagramm.

Wenn man die Daten generiert, bekommt man üblicherweise 2 Matrizen mit den jeweiligen Zusammensetzungen an den Enden der Konoden (A_y bzw. B_y, wobei y die jeweilige Komponente ist). Diese kann man dann einfach nebeneinander setzen und erhält z.B. das mitgelieferte Textfile. Jetzt könnte man schon einmal die Binodale zeichnen. Dazu generiert mein Kollege in

93

Matlab eine neue Matrix, indem er die UpDownGeflippte-Matrix B unter die Matrix A hängt und diese dann zeichnen lässt. Damit die Binodale "schön rund" ist, erzeugt man häufig mehr Punktepaare, als man nachher als Konoden anzeigen lassen möchte. In den mitgelieferten Plots ist so nur jede 5. Konode eingezeichnet.

Die Frage ist nun, wie man das Abfragen der Konoden gestalten kann. Dafür gäbe es jetzt die Möglichkeit einen Key zu erstellen, der sowas sagt wie "plot every Xth tie line".

Ich denke mal, du brauchst auch noch einen schönen Namen den Aufruf dieses Spezialfalls. Da diese zum Zeichnen von Mischungslücken dient, wäre der Englische Name dafür (s.o.) eine Möglichkeit.

was mir noch eingefallen ist:

- Zuweisung der Spalten Es sollte weiterhin möglich sein, Spalten zuzuweisen. Die Frage ist jetzt nur, wie man das macht. Am Einfachsten dürfte es sein, in den ersten 3 Spalten nach den Namen zu suchen. Sollte sie dort nicht gefunden werden, sollte eine Fehlermeldung erscheinen. Zum Zuweisen der "zweiten" dazugehörigen Spalte sollte zu der gefundenen Spaltennummer 3 hinzuaddiert werden. Metadaten können somit erst ab der 7. Spalte auftauchen.

- kartesische Darstellung hier hatte ich vergessen zu erwähnen, wie dies überhaupt funktioniert (vielleicht hast du es aber auch schon alleine herausbekommen).

Da sich die 3. Komponente immer als Differenz zu den gezeigten beiden ergibt, ist diese nicht zwingend zum Darstellen erforderlich. Ausgehend von der gleichen gegebenen table-Datei muss nun nur noch angegeben werden, welche beiden Komponenten dargestellt werden sollen. Dies sollte wie schon oben beschrieben wurde möglich sein.

Das Plotten sollte dann out-of-the-box möglich sein.

Feature Proposal   [closed, Priority 5, Epic '']

output cs:

- implement automatic limit computation ⤳ I prepared something like that; use it. I guess I'll need to convert the streamed data to the accepted format of the axis, at least in order to update limits.

- IDEA:

  - provide the "data cs" as option (not "output cs")
  - convert to the required axis cs automatically before limits are checked
  - keep the converted coordinate system

pgfplotstodo.tex:3115 **Feature Proposal** [closed, Priority 5, Epic '']

write better on-the-fly table generation support like
`\addplot table[y=create col/linear regression{x=Basis,y=L2/ref_h,xmode=log,ymode=log},]`

pgfplotstodo.tex:3119 **Feature Proposal** [closed, Priority 5, Epic '']

improve access to 'create on use' things in addplot table.

pgfplotstodo.tex:3124 **Feature Proposal** [closed, Priority 5, Epic '']

linear regression: at least when used inside of addplot table, the initial values
of x,y,xmode,ymode should be acquired from pgfplots!

pgfplotstodo.tex:3336 **Feature Proposal** [closed, Priority 5, Epic '']

I got several feature requests for non-cartesian axes. Perhaps there is a way to
generalize the complete procedure... as far as I remember, I use the pointxyz
routines anyway to place tick marks and so on. Perhaps it can be reconfigured
to do something "advanced". Idea: nonlinear transformation into the axis
combined with special drawing rotuines for the axis? ternary diagrams
`http://staff.aist.go.jp/a.noda/programs/ternary/ternary-en.html`.
smith charts `http://www.mathworks.com/access/helpdesk/help/toolbox/`
`rf/f2-999699.html http://www.siart.de/lehre/smithdgr.pdf`

pgfplotstodo.tex:3354 **Feature Proposal** [closed, Priority 5, Epic '']

smith charts

`http://www.siart.de/lehre/tutorien.xhtml#smishort`
`http://www.siart.de/lehre/smithdgr.pdf`

`www.amanogawa.com/archive/docs/G-tutorial.pdf`

`http://www.mathworks.com/access/helpdesk/help/toolbox/rf/`
`f2-999699.html`

ok, basic things work todo still:

- UI for default tick positions

- `dense smithchart ticks` is not perfect

- there are problems with limits beyond +-16000

pgfplotstodo.tex:3438   **Feature Proposal**   [closed, Priority 5, Epic '']

provide a `\numplotsperplothandler` or something like that. This would improve things for bar plots!

pgfplotstodo.tex:3767   **Feature Proposal**   [closed, Priority 5, Epic '']

external lib + makefile support: provide data files automatically as prereqs

pgfplotstodo.tex:3780   **Feature Proposal**   [closed, Priority 5, Epic '']

external lib + makefile support: provide data files automatically as prereqs