Todolist for PGFPlots

1.18.3

Christian Feuersänger

August 26, 2026

Contents

1 Release TODO **1**

2 Tests **3**

3 Documentation todo **6**

4 Bugs/Features in PGF/TikZ **6**

5 Bugs in PGFPlots **72**

6 Feature Proposals PGFPlots **75**

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
  - update the description for the `compat` key to document changes introduced by the most recent compatibility level in `pgfplots.preliminaries.tex`
    Note that the key `/pgfplots/compat/mostrecent` is automatically inserted into the manual wherever the suggested level is required.
  - disable `\todosp` in `\pgfplots.preamble.tex`
  - fix warnings
  - browse through it
  - the example of `colorbar horizontal` uses "rand" – and that produced unexpected output once without clue why. Observe it.
- ensure that the "compile" matrix (below this list) is complete for pgf CVS and the latest pgf Stable

- run context tests (basically compile and browse-through. There are no assertions.)
  - run against pgf CVS
  - run against latest pgf stable
- update ChangeLog with "RELEASED VERSION XXX" and update README
- assign git tag for new version
- run `scripts/pgfplots/pgfplotsrevisionfile.sh` to assign the package versions based on tag

  (done automatically when compiling the manual)
- recompile manual (to pick up the correct version)
- run primitive example files
- adjust the release makefile to get correct file names
- ensure that copyright notes are up-to-date, at least in README
- 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 Feuersänger <cfeuersaenger@users.sourceforge.net>
Filename
    pgfplots_1.6.1.ctan.flatdir.zip
Version number
    1.6.1
description:
PGFPlots draws high-quality function plots in normal or logarithmic scaling with
a user-friendly interface directly in TEX. The user supplies axis labels, legend entr
and the plot coordinates for one or more plots and pgfplots applies axis scaling,
computes any logarithms and axis ticks and draws the plots. It supports line
plots, scatter plots, piecewise constant plots, bar plots, area plots, mesh and surfa
plots, patch plots, contour plots, quiver plots, histogram plots, box plots, polar ax
ternary diagrams, smith charts and some more.

Pgfplots is based on PGF/TikZ (PGF); it runs with LaTeX/TeX/ConTeXt.
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 contains latex style files, context package files, plain tex
    include files, generic implementation files, and LUA code all in the same
    directory). I will also upload a TDS zip file to
    /install/graphics/pgf/contrib/pgfplots.
    Thanks for maintaining CTAN!
```

Attention: the announcement text must not be too long. In particular, README extracts are too long.

- release to sourceforge.
  - Make sure to update the README at top-level.
  - make sure to mark the new TDS as "default download for all platforms"
- upload to sourceforge web space (using the release makefile)
  - `make -f pgfplots/scripts/pgfplots/Makefile.pgfplots_release_sourceforge uploaddis` this copies the manuals
  - change `Makefile.pgfplots_release_sourceforge` back to unstable
  - `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`
- use `git push origin --tags` or `git push origin 1.13` to push the tag(s)

2 Tests

last test verifications:

	pgf git	pgf 3.0.1	pgf 3.0.0	pgf 2.10	pgf 2.00	pgf 2.00+con
regressiontests	for 1.17	for 1.16	for 1.12.1	for 1.12*	for 1.8 (7% fail)	2009-12-30
unittests	for 1.16	for 1.16	for 1.12	for 1.12		
manual	for 1.17	for 1.16	for 1.12	for 1.9	for 1.5	
pgfplotstable.pdf	for 1.17	for 1.16	for 1.12	for 1.9	for 1.5	
example latex	for 1.16			for 1.9	2009-12-30	
example context(mkII)	for 1.15			for 1.9	2009-12-30	
example context(mkIV)	for 1.16		for 1.12.1	for 1.9	2009-12-30	
example plain tex	for 1.16			for 1.9	2009-12-30	
tests context (mkII)	for 1.15		for 1.12	for 1.9		
tests context (mkIV)	for 1.16		for 1.12*			

Note that context MK IV needs the pgfplots installed in `~/texmf` (doesn't respect environment variables)

context MK IV may need a special environment like `. ~/context/tex/setuptex`

3 Documentation todo

pgfplotstodo.tex:200 **Documentation Todo** [open, Priority 5, Epic ‘’]

document `\boldmath $...$` for pgfplotstable

pgfplotstodo.tex:204 **Documentation Todo** [open, Priority 5, Epic ‘’]

scatter rgb is not documented

pgfplotstodo.tex:221 **Documentation Todo** [open, Priority 5, Epic ‘’]

Document standalone vs. external

- standalone does not work with loops (compare `http://tex.stackexchange.com/questions/191108/how-to-avoid-copied-code-with-the-combination-pgfplots-standalone`)
- generating just one single figure without a document is simpler with standalone
- external integrates seamlessly
- external can be enabled later-on
- external supports label+ref
- external supports make out of the box
- external supports bounding box control (trim)
- ...

pgfplotstodo.tex:226 **Documentation Todo** [open, Priority 5, Epic ‘’]

The new API for stacked plots + nodes near coords is missing, among it the normalized axis cs but see `\pgfplotspointgetnormalizedcoordinates`

pgfplotstodo.tex:231 **Documentation Todo** [open, Priority 5, Epic ‘’]

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

pgfplotstodo.tex:239 **Documentation Todo** [open, Priority 5, Epic ‘’]

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

Außerdem wäre es zum Einstieg für das aus der Datei lesen schön, wenn es zu den Daten auch ein kurzes Beispiel-File für einen Plot gäbe.

pgfplotstodo.tex:245 **Documentation Todo** [open, Priority 5, Epic ‘’]

try a bar plot with individually shaded bars

FIXME: collect details

pgfplotstodo.tex:249 **Documentation Todo** [open, Priority 5, Epic ‘’]

contour: a change label dist

pgfplotstodo.tex:253 **Documentation Todo** [open, Priority 5, Epic ‘’]

document ’execute at begin axis’ and its new variants

pgfplotstodo.tex:257 **Documentation Todo** [open, Priority 5, Epic ‘’]

document how to plot against the coordindex

pgfplotstodo.tex:262 **Documentation Todo** [open, Priority 5, Epic ‘’]

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

pgfplotstodo.tex:268 **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

pgfplotstodo.tex:274 **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!?

4 Bugs/Features in PGF/TikZ

pgfplotstodo.tex:736 **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}
```

TODOsp: I tried all combinations of the lines commented with and without externalization and all give the same corresponding result. Because you didn’t specify *how* the bug shows itself, please recheck yourself. I’d say: Close it as solved.

pgfplotstodo.tex:746 **pgfbug** [open, Priority 5, Epic ‘’]

external does not work
together with tcolorbox `http://tex.stackexchange.com/questions/319647/tikz-package-external-causes-an-unclosed-tikzpicturetcbhooked-environment`

TODOsp: An accepted solution states to use `\tcbset{shield externalize}` which solves the issue. Maybe, if an error occurs, it can be looked if `tcolorbox` is loaded and if so, state a hint that the above could help to solve the problem?

pgfplotstodo.tex:765 **pgfbug** [open, Priority 5, Epic ‘’]

topath in plots is broken since PGF 3.0.0: the following should be a single connected line (as it used to be for 2.10):

```
\documentclass{standalone}
\usepackage{tikz}
\begin{document}
    \begin{tikzpicture}
        \draw (-1,-1) -- (0,0)
            -- plot [id=x,domain=0.7:1.0] function {x};
    \end{tikzpicture}
\end{document}
```

TODOsp: This seems to be fixed. I get only one line (but since I don’t have PGF 3.0.0 I cannot test, what happened with this version.)

pgfplotstodo.tex:774 **pgfbug** [open, Priority 5, Epic ‘’]

external lib does not properly recompile unresolved references in external files, see `http://texwelt.de/wissen/fragen/7948/markierung-in-pgfplotlegende-fehlerhaft-ubernommen?Seite=1#7973`

TODOsp: In the comment below your answer there is a link that provides a possible patch/solution to that problem. Please have a look at that.

pgfplotstodo.tex:784 **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)

TODOsp: Please a the email of at least an MWE to see, if that bug still exists.

pgfplotstodo.tex:810 **pgfbug** [open, Priority 5, Epic ‘’]

When using externalize function together with a transform canvas, the result is somehow cropped. See this example, compare output with deactivated 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}
```

pgfplotstodo.tex:867 **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:
http://sourceforge.net/mailarchive/forum.php?thread_name=4C0F342B.5040008%40ins.uni-
bonn.de&forum_name=pgf-users

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:876 **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`)

TODOsp: created feature request for that `https://sourceforge.net/p/pgf/feature-requests/103/`

pgfplotstodo.tex:880 **pgfbug** [open, Priority 5, Epic ‘’]

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

pgfplotstodo.tex:884 **pgfbug** [open, Priority 5, Epic ‘’]

implement `\pgfmathfloattocount`

pgfplotstodo.tex:889 **pgfbug** [open, Priority 5, Epic ‘’]

active ’`|`’ characters result in compilation bugs (`\usepackage{program}`)

pgfplotstodo.tex:893 **pgfbug** [open, Priority 5, Epic ‘’]

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

pgfplotstodo.tex:902 **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.

5 Bugs in PGFPlots

pgfplotstodo.tex:1607 **Bug** [open, Priority 10, Epic ‘fillbetween’]

fill between fails to work with PGF 2.10

work-around: use the following patch

```
\documentclass[10pt]{beamer}
\usepackage{tikz}
\usepackage{pgf}
\usepackage{pgfplots}
    \usepgfplotslibrary{fillbetween}
\makeatletter
```

```
\def\tikz@intersect@addto@path@names#1#2{%
  \edef\tikz@marshal{#2\expandafter\noexpand\csname tikz@intersect@path@name@#1\endcsname}
  \expandafter\expandafter\expandafter\def\expandafter\expandafter\expandafter\tikz@marsha
    \expandafter\expandafter\expandafter{\expandafter\tikz@marshal\expandafter{\tikz@inter
  \expandafter\pgfutil@g@addto@macro\expandafter\tikz@intersect@namedpaths\expandafter{\ti
}
\pgfkeys{%
  /tikz/name path global/.code={%
    \expandafter\global\expandafter\let\csname tikz@intersect@path@name@#1\endcsname=\rela
      \tikz@key@name@path@new{#1}{\gdef}%
  },
  /tikz/name path local/.code={%
    \pgfkeys{/tikz/name path={#1}}%
  },
  /tikz/name path/.code={%
      \tikz@key@name@path@new{#1}{\def}%
  },
}
\makeatother


\begin{document}
\begin{frame}

\begin{tikzpicture}
\begin{axis}[%
axis x line*=bottom,
ymin=0,
ymax=1,
]

\addplot[name path=h,domain=-0.01:1,blue,line width=2pt] {5.5*x^4 - 8.4*x^3 + 3.2*x^2 + 0.
\path[name path=axis] (axis cs:0,0) -- (axis cs:1,0);

\only<2->{
\addplot [fill=green, fill opacity=0.2]
    fill between[of=h and axis, soft clip={domain=0.0:0.4}];
}

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

pgfplotstodo.tex:1180 **Bug** [open, Priority 5, Epic ‘units’]

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

```
\documentclass[border=5pt]{standalone}
\usepackage{pgfplots}
    \usetikzlibrary{pgfplots.units}
\begin{document}
    \begin{tikzpicture}
        \begin{axis}[
            change x base,
%            x SI prefix=kilo,   % <-- using this works fine
            x SI prefix=none,   % <-- using this causes the error:
            x unit=m,           %     Choice 'milli' unknown in choice key
            y SI prefix=milli,  %     '/pgfplots/x SI prefix'
            y unit=N,           %                ^ (why x and not y where 'milli' is used?
            xlabel=Distance,
            ylabel=Force,
        ]
            \addplot coordinates {
                (1000,1)
                (2000,1.1)
                (3000,1.2)
                (4000,1.3)
            };
        \end{axis}
    \end{tikzpicture}
\end{document}
```

pgfplotstodo.tex:1186 **Bug** [open, Priority 5, Epic 'polar']

polar plots do not show all tick lines if only parts of the range is displayed, see Christians answer on http://tex.stackexchange.com/questions/323462/can-one-generate-a-boxed-polar-plot-with-pgfplots-and-polaraxis/326741#326741

pgfplotstodo.tex:1232 **Bug** [open, Priority 5, Epic '']

regression in 1.14: point meta number format is in internal FPU representation, no longer a standard number http://tex.stackexchange.com/questions/317365/colorbar-with-definite-number-of-discrete-colours-with-matching-ticks-placement/325584#325584

```
\documentclass[tikz,border=3pt]{standalone}
\usepackage{pgfplots}
    \pgfplotsset{compat=1.14}
\usepackage{expl3}
\ExplSyntaxOn
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% \fpEval{<expression>}
% expandably evaluate floating point <expression>
```

```
\let\fpEval\fp_eval:n
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
\ExplSyntaxOff

\begin{document}
\begin{tikzpicture}
\begin{axis}[
  title={min: \pgfkeysvalueof{/pgfplots/point meta min}, max: \pgfkeysvalueof{/pgfplots/po
  domain=-2:2,
  view={0}{90},
  colormap={CM}{
  samples of colormap=(13 of hot)},
  colormap access=piecewise constant,
  colorbar right,
% nachfolgende Berechnungen gehen nicht mehr:
%  colorbar style={%
%     ytick={%
%       \fpEval{\pgfkeysvalueof{/pgfplots/point meta min}},%
%       \fpEval{\pgfkeysvalueof{/pgfplots/point meta min}+(\pgfkeysvalueof{/pgfplots/point
\pgfkeysvalueof{/pgfplots/point meta min})/13},%
%       ...,%
%       \fpEval{\pgfkeysvalueof{/pgfplots/point meta max}*2} %increase limit to ensure pla
%     }
%  }
]
\addplot3[surf,shader=interp]
{exp(-x^2-y^2)*x};
\end{axis}
\end{tikzpicture}
\end{document}
```

pgfplotstodo.tex:1241 **Bug** [open, Priority 5, Epic ‘’]

it seems as if the layer configuration does not really work:
`set layers={axis on top}` has a different effect than `axis on top`.

It seems as if I need `\pgfplotsgetlayerforstyle` somewhere when drawing the individual entities – setting `on layer` is not enough.

pgfplotstodo.tex:1252 **Bug** [open, Priority 5, Epic ‘fillbetween’]

fillbetween / intersection lib bug: see minimal at
`http://tex.stackexchange.com/a/181224/95441`

TODOsp: is this related to
`https://sourceforge.net/p/pgfplots/bugs/111/`? (I took also a note there that it is listed here). If it *is* related, please remove this item here to avoid to have duplicates.

pgfplotstodo.tex:1267 **Bug** [open, Priority 5, Epic ‘’]

external bug: the external lib must keep pgfpictureid, otherwise it mixes remembered positions!

see `http://tex.stackexchange.com/questions/72781/problem-tikz-pgfplots-and-external-coordinates-using-overlay/72804#comment782265_72804` for a MWE

TODOsp: Is this fixed now? At least using default ‘mode’ there is no difference with and without externalization.
Note to SP: → check again with ‘mode=list and make’!

I could still verify it using 1.14-74-g5452a45

pgfplotstodo.tex:1274 **Bug** [open, Priority 5, Epic ‘’]

fpu is unable to compute `exp(-4.08893e7)` (runs into overflow somewhere).

To do: replace it by 0

pgfplotstodo.tex:1283 **Bug** [open, Priority 5, Epic ‘’]

document `http://tex.stackexchange.com/questions/65045/pgfplots-axis-xmode-ymode-in-user-defined-style/65069?noredirect=1#comment721421_65069` in a suitable way

pgfplotstodo.tex:1288 **Bug** [open, Priority 5, Epic ‘’]

external lib screws up prefix option? `http://tex.stackexchange.com/questions/243935/error-using-tikz-externalize-cant-write-md5-file`

pgfplotstodo.tex:1293 **Bug** [open, Priority 5, Epic ‘’]

pgfmathparse no longer compatible with heightof etc. of calc package?
`http://tex.stackexchange.com/questions/289551/how-to-resolve-conflict-between-versions-of-texlive-and-pgf`

pgfplotstodo.tex:1319 **Bug** [open, Priority 5, Epic ‘’]

colormap access=piecewise constant can produce visual artifacts:

```
\documentclass{standalone}
\usepackage{pgfplots}
    \usepgfplotslibrary{colorbrewer}
    \pgfplotsset{compat=1.14,lua debug}
\begin{document}
```

```
\begin{tikzpicture}
    \begin{axis}[
        view={0}{90},
        colormap/Purples-6,
        colorbar,
        colormap access=piecewise constant,
    ]
    \addplot3[surf,shader=interp,] {x*y^2};
    \end{axis}
\end{tikzpicture}
\end{document}
```

This has an artifact right in the middle. Acrobat reader has a slightly different one compared to xpdf

pgfplotstodo.tex:1323 **Bug** [open, Priority 5, Epic ‘’]

axis description styles contain “on layer”, but they ignore it!

pgfplotstodo.tex:1339 **Bug** [open, Priority 5, Epic ‘’]

syntax highlighting issue:

```
\begin{quote}
    |edge node=node [every edge quotes,|\meta{options}|]{|\meta{text}|}|
\end{quote}
```

results in

```
    edge node=node [every edge quotes] options ]{ text }
```

pgfplotstodo.tex:1363 **Bug** [open, Priority 5, Epic ‘’]

tikz bug: using `-- cycle` exchanges start/end positions of nodes

commented minimal

pgfplotstodo.tex:1369 **Bug** [open, Priority 5, Epic ‘z buffer’]

z buffer=sort appears to cause wrong results – must use min(depths) instead of mean(depths)! `http://tex.stackexchange.com/questions/281953/perspective-error-in-pgfplots-surface`

pgfplotstodo.tex:1377 **Bug** [open, Priority 5, Epic ‘’]

pgfplotstable: dec sep align does not work if someone uses custom `column type` or `assign column name`.

See `http://tex.stackexchange.com/questions/276395/pgfplotstable-formatting-problems/277284#277284`

pgfplotstodo.tex:1382 **Bug** [open, Priority 5, Epic ‘’]

add sanity check to lua backend before sending mapped coordinates to TeX: `http://tex.stackexchange.com/questions/269656/pgfplots-with-big-data-dimension-too-large`

pgfplotstodo.tex:1407 **Bug** [open, Priority 5, Epic ‘’]

Using `\thisrow{col}` merely expands to the value, but does not implicitly add parenthesis around the value.

Consequently, something like `\thisrow{col}^2` will produce unexpected results for rows like -2 `http://tex.stackexchange.com/questions/262774/pgfmath-misbehaving-when-reading-negative-values-from-table-in-pgfplots`

However, textual columns should still be processed as usual. And: it used to work to write `\dimen0=\thisrow{col}pt` which would break if parenthesis is added automatically

Idea: something like

```
\def\thisrow#1{-2}
\let\thisrowX=\thisrow
\def\thisrow#1{(\thisrowX{#1})}%
\edef\XXX{\thisrow{col}^2}
\message{EXPANDED: \XXX^^J}
```

before evaluating table expressions?

pgfplotstodo.tex:1413 **Bug** [open, Priority 5, Epic ‘precision’]

inaccuracy when determining minor ticks? `http://tex.stackexchange.com/questions/262662/pgfplots-sometimes-minor-ticks-dont-show`
RELATED TO FOLLOWING TICKET

pgfplotstodo.tex:1430 **Bug** [open, Priority 5, Epic ‘precision’]

Tick labels suffer from inexact arithmetics in Tikz foreach:

TODOsp: Is this the same root cause as the previous bug? → combine them

KAPUTT:

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

OK:

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

pgfplotstodo.tex:1456 **Bug** [open, Priority 5, Epic 'precision']

... and rounding:

Generell erscheint mir die ...-Syntax nicht robust. Bei kleinen, durchaus nicht ungewöhnlichen, 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}
```

Führt dazu, dass die Hilfslinien mit zunehmender Größe immer weiter von den Hauptgitterlinien verschoben sind, obwohl diese aufeinander liegen 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 genießen.

TODOsp: Is this the same root cause as the previous bug? → combine them

pgfplotstodo.tex:1464 **Bug** [open, Priority 5, Epic 'v2']

consider using `unbounded coords=jump` as default for mesh/surface plots
`http://tex.stackexchange.com/questions/259012/glitch-in-3d-surface-plot-using-pgfplots-not-solved-by-z-buffer`

TODOsp: feature for v2? YES (epic=v2)

pgfplotstodo.tex:1479 **Bug** [open, Priority 5, Epic '']

fpu bug:

```
\begingroup
\tracingmacros=2 \tracingcommands=2
```

```
\pgfkeys{/pgf/fpu}
\pgfmathparse{exp(-3.4814e4)}\pgfmathresult
\endgroup
```

fails to compile. http://tex.stackexchange.com/questions/251496/fermi-dirac-in-pgfplots-dimension-too-lage

pgfplotstodo.tex:1485 **Bug** [open, Priority 5, Epic 'z buffer']

view dir=reverse appears to result in a wrong 3d view directions and screws up z buffering, compare http://tex.stackexchange.com/questions/250825/axis-dir-reverse-and-colormap-bug-in-tikz

pgfplotstodo.tex:1493 **Bug** [open, Priority 5, Epic '']

3d axes and y dir = reverse seems to corrupt the z ordering, see http://texwelt.de/wissen/fragen/6131/pgfplots-3d-oberflachenproblem

TODOsp: This is the same bug as the previous one, right? → combine them to one item

pgfplotstodo.tex:1523 **Bug** [open, Priority 5, Epic 'smithchart']

smithchart admittance graph has wrong sign

```
This Mail is in German, since your name suggests, that you speak German.
If that is not the case, please let me know, and I will write send you
this mail again in English.

Sehr geehrter Herr Feuersänger,

ich schreibe momentan meine Bachelor-Arbeit und muss dafür Antennen
vermessen und die Messergebnisse graphisch darstellen. Ich habe mit
Begeisterung festgestellt, dass ich mit PGFPlot recht direkt s1p- und
s2p-Dateien plotten kann.
Vielen Dank dafür!

Allerdings gibt es ein paar Darstellungsprobleme:

-> Bei Admitanzgraphen, also smithchart mirrored=true werden die
Koordinaten auf dem unteren Halbkreis mit negativen Vorzeichen
dargestellt (dies ist auch in dem Beispiel im Manual so)
Jedoch sind bei jedem Admitanz-Smith-Graph die Vorzeichen umgekehrt (da
der Admitanz-Graph per Definition 1/Z_L=1/(a+b*i)=a/(a^2+b^2)-
ib/(a^2+b^2)

Gibt es eine Option das einzurichten?
```

```
    y dir=reverse
bewirkt nichts.
```

pgfplotstodo.tex:1549 **Bug** [open, Priority 5, Epic 'smithchart']

smithcharts: -> Ab einer Größe von 14cm funktionieren yticklabel around circle, few smithchart ticks etc. nicht mehr. Das lässt sich manuell hinbekommen:

```
    \begin{smithchart}[
        title=\text{title},
        smithchart mirrored=true,
        xticklabel shift=-19pt,
        grid style={blue},
        ticklabel style={blue},
        width=15.2cm,
        xtick={0.2,0.5,1,2,5},
        ytick={0,0.2,0.5,1,2,5,-0.2,-0.5,-1,-2,-5},
        xgrid each nth passes y={2},
        ygrid each nth passes x={2},%ytick align=center,
        ytick align=center,
        yticklabel style={
            rotate=180,
        },
    ]
```

pgfplotstodo.tex:1665 **Bug** [open, Priority 5, Epic 'smithchart']

substitute axis cs fails with smithcharts:

```
\documentclass{standalone}
\usepackage{pgfplots}
    \usepgfplotslibrary{smithchart}
    \pgfplotsset{compat=newest}
\begin{document}
   \begin{tikzpicture}
     \begin{smithchart}
%     \pgfplotsset{is smithchart cs}
%     \begin{scope}[/pgfplots/is smithchart cs]
       \draw [black!40,dashed] (0,0) arc (0:360:.5);
%     \end{scope}
     \end{smithchart}
   \end{tikzpicture}
\end{document}
```

pgfplotstodo.tex:1698 **Bug** [open, Priority 5, Epic 'substitute axis cs']

substitute axis cs fails to work with `++ (0,1cm)` (probably for `++` every time)

workaround: use `++ (0cm,1cm)` or `compat=1.10`.

```
\documentclass{article}
\usepackage{pgfplots}
\begin{document}
    \pgfplotsset{compat=1.9}
\begin{tikzpicture}
\begin{axis}[
    xmin=-1,xmax=1,
    ymin=-1,ymax=1
]
    \draw (axis cs:0,0) -- ++(0,1cm);
\end{axis}
\end{tikzpicture}

    \pgfplotsset{compat=1.12}
\begin{tikzpicture}
\begin{axis}[
    xmin=-1,xmax=1,
    ymin=-1,ymax=1
]
    \draw (axis cs:0,0) -- ++(0,1cm);
\end{axis}
\end{tikzpicture}
\end{document}
```

pgfplotstodo.tex:1804 **Bug** [open, Priority 5, Epic 'fillbetween']

error bars + fill between do not work together (per email of Jerzy Wrobel Jan 28, 2015):

```
\documentclass[12pt,border=8pt]{standalone}
%\usepackage[utf8]{inputenc}
%\usepackage{mathtools}
\usepackage{pgfplots}
%\usepackage[T1]{fontenc}
%\usepackage{fouriernc}
    \pgfplotsset{
        compat=newest,
        every axis/.append style={line width=0.03cm,axis line style={line width=0.02cm}},
        every tick/.append style={line width=0.02cm,color=black},
        every axis label/.append style={font=\small},
        tick label style={font=\small},
        legend style={font=\small,draw=none},
```

```
        title style={font=\small},
        axis on top=false,
        table/col sep=comma,
    }
    \tikzset{
        every pin/.style={font=\small},
    }
    \pgfkeys{/pgf/number format/set thousands separator = }
    \usepgfplotslibrary{fillbetween}
    \usetikzlibrary{patterns}
\begin{document}
\begin{tikzpicture}
\begin{axis}[
    width=70mm,
    height=70mm,
    axis x line=box,
    axis y line=box,
    scale only axis,
    enlarge x limits=true,
    enlarge y limits=true,
    grid=none,
    xlabel=wave length (microns),
    ylabel=Experiment1916 (a.u.),
    extra y ticks={0},
    extra y tick labels={},
    extra y tick style={
        grid=major,major grid style={line width=0.02cm,color=black,densely dashed},
    },
]
\addplot+ [
    name path=1,
    line join=round,
    every mark/.append style={solid,fill opacity=1.0},
    mark repeat=1,
    mark phase=1,
    error bars/.cd,
    y dir=both,
    y fixed=10,
] table [
    skip first n=4,
    x index=0,
    y index=1,
] {Coblentz1919.csv} ;
\path[name path=B] (0.4,0) -- (0.75,0);
\addplot[pattern=north west lines,forget plot] fill between[of=1 and B];
\end{axis}
\end{tikzpicture}
\end{document}
file:
```

```
#Spectral energy distribution of a cylindrical acetylene flame.
# T = 2360 K
# W. W. Coblentz, Scientific Papers of the Bureau of Standards, Vol.15, 639 (1920).
wave length (microns),Experiment1916 (a.u.),Revised1919 (a.u.)
0.400,5.9,5
.425,8.2,7
.440,10.0,8.5
.450,11.5,10.0
.460,13.0,11.8
.475,16.0,15.0
.500,21.9,20.9
.520,27.9,27.5
.525,29.5,29.2
.540,35.0,34.6
.550,38.9,38.9
.560,42.9,42.9
.575,49.8,49.8
.580,52.2,52.2
0.600,62.1,62.5
.620,73.0,73.3
.625,75.7,76.1
.640,84.7,85.0
.650,91.1,91.2
.660,97.4,97.6
.675,107.5,107.5
.680,110.9,110.9
.700,124.6,124.1
.720,138.5,137.5
.725,141.9,141.0
.740,152.0,151.0
.750,158.9,157.9
```

pgfplotstodo.tex:1809 **Bug** [open, Priority 5, Epic ‘’]

spurious shifts/displacement when using group plots and discontinuities:
http://tex.stackexchange.com/questions/183399/
wrong-placement-and-or-size-of-picture-plot-with-groupplot-of-table-and

pgfplotstodo.tex:1815 **Bug** [open, Priority 5, Epic ‘’]

get rid of this boolean “apply@datatrafo@”. It is a synonym for “is linear”. I would even say: apply the trafo to log axes as well, perhaps with no-op configuration

pgfplotstodo.tex:1845 **Bug** [open, Priority 5, Epic ‘’]

REGRESSION:

```
\documentclass{article}
\usepackage{pgfplots}
    \pgfplotsset{compat=newest}
\begin{document}
\begin{tikzpicture}
    \begin{axis}[ybar,
        symbolic x coords={foo,bar,baz},
        x=2cm,
        enlarge x limits={abs=bar}, xtick=data,
    ]
        \addplot coordinates { (foo,1) (bar,3) (baz,2) };
    \end{axis}
\end{tikzpicture}
\end{document}
```

broke sometime between 1.6 (OK) and 1.7 (compile error).

The problem is the definition of `\pgfkeysdef{/pgfplots/@enlargelimits/abs value}` which attempts to decide if the argument contains a unit. If so, it skips the standard coordinate mapper. In this case, 'abs=bar' has no unit, but the math parser bails out because the argument is no math expression at all... how can I fix that!?

pgfplotstodo.tex:1893 **Bug** [open, Priority 5, Epic 'dateplot']

combination of dateplot and x tick label as interval does not work well together:

- one can only format one of the dates
- the description applies to the *right* end rather than the left end

commented minimal, because it raises an error when TeXing

pgfplotstodo.tex:1898 **Bug** [open, Priority 5, Epic '']

xtick key still relies on PGF's basic level foreach – and the numbers inside of it are not parsed as it seems.

pgfplotstodo.tex:1916 **Bug** [open, Priority 5, Epic 'fillbetween']

fillbetween + softclip: soft clip should return the input path if the input path is contained in the soft clip region

see `http://tex.stackexchange.com/questions/195678/pgfplots-using-the-fillbetween-library/195697?noredirect=1#comment453086_195697`

TODOsp: Is this related to
https://sourceforge.net/p/pgfplots/bugs/111/ or has this bug another root cause?

pgfplotstodo.tex:1971 **Bug** [open, Priority 5, Epic ‘’]

extra ticks produce unwanted shifts of labels

```
\documentclass[border=5pt,varwidth]{standalone}
\usepackage{pgfplots}
    \pgfplotsset{compat=newest}
\begin{document}

I'm using pgfplots 1.10 and observe the following problem. The x label
shifts down a bit if I introduce extra x ticks, even if their labels are
empty. The following example produces the observed behavior:

\begin{tikzpicture}
\begin{axis}[
  width=4cm,
  xlabel=Test,
  axis x line=bottom,
  axis y line=left,
]
\addplot [black] {x^2};
\end{axis}
\begin{axis}[
  width=4cm,
  xlabel=Test,
  axis x line=bottom,
  axis y line=left,
  extra x ticks={0},
  extra x tick labels={},
]
\addplot [black] {x^2};
\end{axis}
\end{tikzpicture}

It does not happen with the old, static, label placement (e.g. if I
remove the compat setting). I was not able to test with 1.11, but
there's nothing related mentioned in the release notes, if I did not
overlook it. Any ideas what I'm missing or if it is a bug and how to
work around it? I need two adjacent graphs in subfloats but only one of
them has extra ticks, so the spacing should be equal to look reasonable.
Adding an extra tick on the other plot helps, but the spacing looks
better without, so that's not an option. And switching back to the old
behavior means manually tweaking the distance, what makes no sense to me
either.
```

```
\end{document}
```

root cause: `\pgfplots@ticklabel@maxtickdimen@finish` is called twice: once for the normal ticks and once for the extra ticks (node that the `*reset` routine is invoked just once). However, the `@finish` routine adds some `@extrashift` – and that is done twice as well. The extra shift belongs to the tick offset

pgfplotstodo.tex:1976 **Bug** [open, Priority 5, Epic ‘’]

composed plot handlers (hist, boxplot, contour) need access to coordinate filters etc. during their first survey phase

pgfplotstodo.tex:1995 **Bug** [open, Priority 5, Epic ‘’]

pgfplots fails to handle very small number ranges on logarithmic plots (i.e. if, say, xmin and xmax are very very close to each other).

see `http://www.mrunix.de/forums/showthread.php?76399-tikz-PGFplots-Fehler-dimension-too-large&p=358818#post358818`

pgfplotstodo.tex:2005 **Bug** [open, Priority 5, Epic ‘nodes near coords’]

ybar interval does not work well with nodes near coords

ybar interval has an artificial last coordinate. This will also receive nodes near coords.

See `http://tex.stackexchange.com/questions/181061/pgfplots-clipping-everything-outside-a-specific-area`

pgfplotstodo.tex:2014 **Bug** [open, Priority 5, Epic ‘nodes near coords’]

ybar interval + nodes nears coords: nodes are displaced.

The shift does not work here.

See `http://tex.stackexchange.com/questions/181061/pgfplots-clipping-everything-outside-a-specific-area`

pgfplotstodo.tex:2025 **Bug** [open, Priority 5, Epic ‘’]

Coordinate filtering + error bars do not work.

Expected behavior: error coordinates should be computed and coord filters should be applied to the result. Currently, filters are evaluated first and then error coordinates.

See `http://tex.stackexchange.com/questions/154084/how-to-scale-both-data-and-error-bars-in-pgfplots`

pgfplotstodo.tex:2030 **Bug** [open, Priority 5, Epic ‘’]

`legend to name` appears to be broken if the legend is empty (empty matrix?) TODO: verify that empty legends do not cause problems here!

pgfplotstodo.tex:2034 **Bug** [open, Priority 5, Epic ‘’]

cube/size x has no effect if used in a 2d axis

pgfplotstodo.tex:2179 **Bug** [open, Priority 5, Epic ‘’]

copy paste from the manual does not copy empty lines – which is quite bad for the 3d surface examples

This appears to be a known issue for pdf, with few available workarounds. The listings package suffers from it as well (at least regarding trailing spaces/tabs); there are a couple of related questions at tex.sx

A way might be to use `\usepackage{accsupp}` with something like

```
\documentclass{article}
\usepackage{accsupp}
\begin{document}
\long\def\XXX{A
    ^^J
  ^^J
B}
\BeginAccSupp{method=escape,ActualText=\XXX}
First line

second line
\EndAccSupp{}

\end{document}
```

Expected result: copy-pasting that segment should yield A (newline) B. But it does not work with newlines as it seems

One entry on tex.sx mentioned that Microsoft products can insert empty lines in copy-paste output (suggested solutions there was to generate such a pdf and include it as graphics unless I am mistaken)

I managed to get this up and running so far – but the result is unusable! both acrobat and xpdf get utterly confused unless one hits exactly on some invisible lines Problems here:

commented minimal, because it raises an error when TeXing

- one has to mark the right segments, and these can only be found by trial and error
- hitting the wrong one will copy nothing or only parts
- in the manual, I often had the wrong content in my clipboard (some older state of the clipboard, no idea where that came from)

pgfplotstodo.tex:2184 **Bug** [open, Priority 5, Epic 'fillbetween']

tikz intersections: the "store in macro" methods do not use global macros. Should they!?

pgfplotstodo.tex:2190 **Bug** [open, Priority 5, Epic 'fillbetween']

fill between:
`addplot [draw=green!80,fill=green!20] fill between [of=lower and axis, soxft clip={(inters`
`\numintersections) rectangle (axis cs:\pgfkeysvalueof{/pgfplots/xmax},0)}]`
produces no "unknown key soxft clip" error

pgfplotstodo.tex:2231 **Bug** [open, Priority 5, Epic 'fillbetween']

fill between / intersections lib bug: spurious intersections
`http://tex.stackexchange.com/questions/180127/`
`stacking-plots-in-animation-using-fill-between-library-with-dynamic-calculation/`
`180299#180299`

analysis in `~/tmp/bug_intersectionlibs.tex`: problem seems to be the duplicate detection

problem: almost parallel lines:

```
\begin{tikzpicture}
    \makeatletter
     \def\A{%
         \pgfsyssoftpath@movetotoken{15.60165pt}{0.60783pt}%
        \pgfsyssoftpath@linetotoken{0pt}{0pt}%
    }%
     \def\B{%
         \pgfsyssoftpath@movetotoken{15.60165pt}{0.81604pt}%
        \pgfsyssoftpath@linetotoken{0.0pt}{0.33199pt}%
    }%
    \draw[blue] \pgfextra{\pgfsetpathandBB\A};
```

```
\draw[red] \pgfextra{\pgfsetpathandBB\B};
\pgfintersectionofpaths%
    {%
      \pgfsetpath\A
    }%
    {%
      \pgfsetpath\B
    }%

\pgfmathloop%
\ifnum\pgfmathcounter>\pgfintersectionsolutions\relax%
\else%
\draw[red] \pgfextra{\pgftransformshift{\pgfpointintersectionsolution{\pgfmathcounter}
node[anchor=center] {\pgfmathcounter/\pgfintersectionsolutions};
\repeatpgfmathloop%

 \end{tikzpicture}
```

pgfplotstodo.tex:2236 **Bug** [open, Priority 5, Epic ‘’]

pgfplotstable appears to have problems with `#` in col names, compare `http://tex.stackexchange.com/questions/175395/how-can-csv-files-be-put-into-latex-without-having-to-adjust-much`

pgfplotstodo.tex:2371 **Bug** [open, Priority 5, Epic ‘’]

`ytick=data` does not work together with stacked plots because, by design, it merely considers coordinates of the *first* plot. That’s nonsense for stacked plots.

pgfplotstodo.tex:2392 **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.

pgfplotstodo.tex:2400 **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`

pgfplotstodo.tex:2442 **Bug** [open, Priority 5, Epic ‘boxplot’]

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

pgfplotstodo.tex:2451 **Bug** [open, Priority 5, Epic ‘’]

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

pgfplotstodo.tex:2463 **Bug** [open, Priority 5, Epic ‘’]

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

pgfplotstodo.tex:2487 **Bug** [open, Priority 5, Epic ‘’]

auto tick label assignment can sometimes produce strange results:

commented minimal, because it raises an error when TeXing

See also the examples for boxplots in the manual

pgfplotstodo.tex:2504 **Bug** [open, Priority 5, Epic ‘’]

the 3d clip path is sometimes bad: perhaps it should be the bounding box instead!?

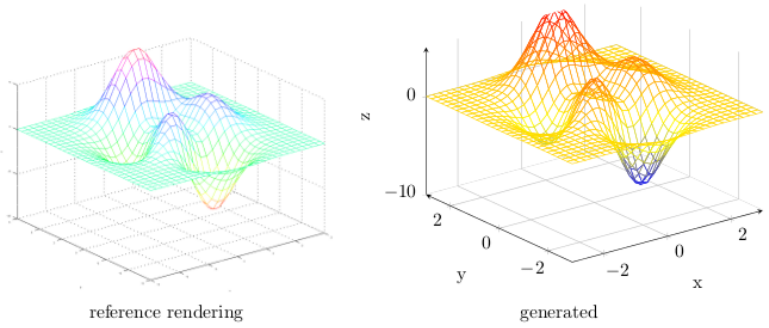

Figure 44: Mesh plot. `meshPlot`

pgfplotstodo.tex:2520 **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:2524 **Bug** [open, Priority 5, Epic ‘’]

`ybar legend` does not contain `ybar` (sourceforge 3482770)

pgfplotstodo.tex:2529 **Bug** [open, Priority 5, Epic ‘’]

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

pgfplotstodo.tex:2537 **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:2545 **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`.

pgfplotstodo.tex:2549 **Bug** [open, Priority 5, Epic ‘quiver’]

quiver plots: the clip path does not respect arrow paths

pgfplotstodo.tex:2577 **Bug** [open, Priority 5, Epic ‘’]

Alignment bug: axis x line=middle combined with a yshift shifts the xlabel incorrectly: Using `xlabel style = {yshift=3cm}` in the plot will correctly position the x label (to its default position).

commented minimal, because it raises an error when TeXing

pgfplotstodo.tex:2584 **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.

pgfplotstodo.tex:2591 **Bug** [open, Priority 5, Epic ‘’]

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

(See `https://tex.stackexchange.com/q/433059`.)

pgfplotstodo.tex:2649 **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:2742 **Bug** [open, Priority 5, Epic ‘’]

groupplots + extra braces or foreach are incompatible.

pgfplotstodo.tex:2746 **Bug** [open, Priority 5, Epic ‘’]

numplotspertype and forget plot and ybar interval yields errors.

pgfplotstodo.tex:2752 **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:2783 **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}
```

```
\documentclass[border=5pt]{standalone}
\usepackage{pgfplots}
\begin{document}
\begin{tikzpicture}
\begin{axis}
    \addplot+ [
        samples at={
%            0,10,...,360                % <-- works
            0,10,...,180,200,...,360    % <-- only evaluates up to ``180''
%            % both following lines cause an error
%            % (what could be an advantage of using `\foreach' instead of
%            %  directly using the above syntax?)
%            \foreach \x in {0,20,...,340} {\x, }360
%            \foreach \x in {0,10,...,180,200,...,340} {\x, }360
        },
    ] ( {sin(x*2)}, {cos(x)} );
\end{axis}
\end{tikzpicture}
\end{document}
```

pgfplotstodo.tex:2792 **Bug** [open, Priority 5, Epic ‘groupplots’]

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

pgfplotstodo.tex:2796 **Bug** [open, Priority 5, Epic ‘’]

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

pgfplotstodo.tex:2804 **Bug** [open, Priority 5, Epic ‘’]

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

pgfplotstodo.tex:2809 **Bug** [open, Priority 5, Epic ‘’]

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

could you be more specific here?

pgfplotstodo.tex:2813 **Bug** [open, Priority 5, Epic ‘ternary’]

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

pgfplotstodo.tex:2818 **Bug** [open, Priority 5, Epic ‘’]

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

pgfplotstodo.tex:2829 **Bug** [open, Priority 5, Epic ‘’]

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

pgfplotstodo.tex:2854 **Bug** [open, Priority 5, Epic ‘quiver’]

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:2859 **Bug** [open, Priority 5, Epic ‘’]

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

pgfplotstodo.tex:2880 **Bug** [open, Priority 5, Epic 'Usability']

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?

DUPLICATE: the nasty dimension-to-large message could probably be avoided if pgfplots would simply clip the results to TeX's range

pgfplotstodo.tex:2887 **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>).

could you give an example here, too, which shows the bug?

pgfplotstodo.tex:2893 **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:2903 **Bug** [open, Priority 5, Epic '']

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

pgfplotstodo.tex:2907 **Bug** [open, Priority 5, Epic '']

FPU: atan doesn't check for unbounded inputs.

pgfplotstodo.tex:2912 **Bug** [open, Priority 5, Epic '']

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

pgfplotstodo.tex:2920 **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:2929 **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:2934 **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:2938 **Bug** [open, Priority 5, Epic ‘’]

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

pgfplotstodo.tex:2944 **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:2948 **Bug** [open, Priority 5, Epic ‘’]

log samples in plot expression for 3D plots

pgfplotstodo.tex:2952 **Bug** [open, Priority 5, Epic ‘gnuplot’]

different log bases and gnuplot

pgfplotstodo.tex:2964 **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:2971 **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:2976 **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:2982 **Bug** [open, Priority 5, Epic ‘nodes near coords’]

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

pgfplotstodo.tex:2986 **Bug** [open, Priority 5, Epic ‘’]

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

pgfplotstodo.tex:2990 **Bug** [open, Priority 5, Epic ‘’]

axis equal for semilog plots is not correct (?)

pgfplotstodo.tex:2996 **Bug** [open, Priority 5, Epic ‘’]

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

pgfplotstodo.tex:3000 **Bug** [open, Priority 5, Epic ‘’]

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

pgfplotstodo.tex:3008 **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.

pgfplotstodo.tex:3014 **Bug** [open, Priority 5, Epic ‘’]

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

Is this still current? There has been a major revision some time ago...

pgfplotstodo.tex:3049 **Bug** [open, Priority 5, Epic ‘’]

addplot3 vs addplot:

```
  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}
    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:2285 **Bug** [open, Priority 4, Epic ‘’]

adding a post action with decoration to a pgfplots style results in an error if markers are active

pgfplotstodo.tex:1904 **Bug** [open, Priority 3, Epic ‘’]

TikZ wastes time during its point evaluations (i.e. expressions of sorts `(1,2)`): it calls `tikz@checkunit` in order to check for units and throws the math result away (and computes it again afterwards).

pgfplotstodo.tex:1980 **Bug** [open, Priority 3, Epic ‘’]

contour plot handler does not allow ’name path’.

pgfplotstodo.tex:2348 **Bug** [open, Priority 3, Epic ‘’]

Usability issues around histograms: `http://tex.stackexchange.com/questions/151411/histogram-frequention-polygon/151551#151551`

commented minimal, because it raises an error when TeXing

Problems:

- point meta uses the wrong values (try out nodes near coords)
- ybar interval is active for the second plot
- interval=false should be deduced automagically

pgfplotstodo.tex:2437 **Bug** [open, Priority 3, Epic 'v2']

units and square brackets

```
Dann habe ich noch eine Anmerkung bzgl. der Verwendung von Einheiten in
Achsenbeschriftungen, Punkt "5.10 Units in Labels". Generell sehr hilfreich,
allerdings möchte ich darauf hinweisen, dass die Verwendung von eckigen
Klammern um Einheiten falsch und unzulässig ist. Auch wenn man es leider immer
wieder (falsch) selbst in (schlechten) Lehrbüchern sieht. Indes, falsch bleibt
falsch und mit diesem schlechten Usus sollte gebrochen werden. Hierzu 'DIN
1313: Größen', die Anmerkung zu Punkt 4.3

------ Zitat
ANMERKUNG;
Die eckigen Klammern dürfen nicht um Einheitenzeichen gesetzt werden. Angaben
wie [kg] sind nicht zu verwenden, auch nicht zur Beschriftung von
Koordinatenachsen in graphischen Darstellungen (siehe DIN 461).
------ Ende Zitat

'DIN 461: Graphische Darstellung in Koordinatensystemen' gibt Beispiele, wie es
gemacht werden soll.

Hintergrund: formaler Zusammenhang ist X={X}[X]. Das heißt: X=5N --> {X}=5 und
[X]=N. Unsinnig ist [N] da "[...]" für "Einheit von ..." steht und was ist die
Einheit einer Einheit?! Nachzulesen auch in jedem guten Physik-
Lehrbuch.

Wie gesagt: leider eine sehr schlechte Angewohnheit und den meisten unbekannt
oder schlicht egal. Anders ist die allzu häufige Verwendung nicht zu erklären.
Deshalb sollte zumindest ein Hinweis zu lesen sein, dass diese Syntax
eigentlich falsch/unzulässig ist.
```

While I understand the problem, I am inclined to keep it as-is... I cannot break backwards compatibility now, and I hear few complaints. I suppose the units stuff could be reviewed at some time

pgfplotstodo.tex:2455 **Bug** [open, Priority 3, Epic 'gnuplot']

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

pgfplotstodo.tex:2497 **Bug** [open, Priority 3, Epic ‘’]

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

pgfplotstodo.tex:3061 **Bug** [open, Priority 3, Epic ‘’]

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

... but nobody has ever complaint so far

pgfplotstodo.tex:1644 **Bug** [open, Priority 2, Epic ‘lua’]

The following fails to compile with lualatex (due to the double quotes apparently). It works in pdflatex

Low prio because the point meta is not actually used – and if it is used correctly, it works in both lualatex and pdflatex

```
\documentclass{standalone}
\usepackage{pgfplots}
    \pgfplotsset{%
        compat=newest,
    }
\begin{document}
\begin{tikzpicture}
\begin{axis}[
    %point meta=explicit symbolic,
    %nodes near coords=\pgfplotspointmeta,
]
\addplot table[x index={1}, y expr=\coordindex, header=false, meta index={0}] {
"i"  62.099
"i"  62.046
"i"  56.304
"i" -44.258
"2" -28.826
"i"  18.740
"i" -14.653
"i"  14.402
"i" -12.907
"i"  12.295
};
\end{axis}
\end{tikzpicture}
\end{document}
```

pgfplotstodo.tex:2271 **Bug** [open, Priority 2, Epic ‘’]

context + color mix produces spurious shifts

```
I'm using pgfplots to create some bar charts with error bars in ConTeXt,
and I want to use custom colors for them.  This works, but if I mix a
custom color with another color, it results in a gap between the
error bar line and the error bar mark.  If two custom colors are
mixed, the gap seems to be doubled.  Mixing with predefined colors
works normally.  MWE:

\usemodule[tikz]
\usemodule[pgfplots]

\definecolor    [Blue]    [h=3465a4]
\definecolor    [White]    [h=ffffff]

\pgfplotsset{compat=newest}

\starttext
\starttikzpicture
    \startaxis[ybar, error bars/y dir=both, error bars/y explicit]
        \addplot+[blue!50!white] coordinates {(1,5) +- (0,3)};
        \addplot+[Blue] coordinates {(1,5) +- (0,3)};
        \addplot+[Blue!50!white] coordinates {(1,5) +- (0,3)};
        \addplot+[Blue!50!White] coordinates {(1,5) +- (0,3)};
    \stopaxis
\stoptikzpicture
\stoptext

Here you can see the result:
http://666kb.com/i/cl86t0mfpin35tht7.png
```

pgfplotstodo.tex:2280 **Bug** [open, Priority 2, Epic ‘’]

Adding user decorations to axis lines might interfere with the discontinuities which are also decorations, compare
http://tex.stackexchange.com/questions/160936/random-decoration-of-plot-axis-messes-up-direction-of-axis-arrow/161025#comment368659_161025

I already added Jake’s workaround to the code, but might still need more cleanup (and for non-centered axis lines as well)

pgfplotstodo.tex:2354 **Bug** [open, Priority 2, Epic ‘’]

Loading a pgfplots library via `\usetikzlibrary{pgfplots...}` fails unless pgfplots is loaded. This should be reported properly (or perhaps load pgfplots implicitly?)

pgfplotstodo.tex:2387 **Bug** [open, Priority 2, Epic ‘’]

The combination of 3d unit vectors and `scale mode=stretch to fill` is useful, but unsupported.

I made a brief experiment to verify that it does something useful if

- the data scaling is disabled in setunitvector
- the first if branch of `...@prepare@plotbox@limits` is activated.

References: `http://tex.stackexchange.com/questions/139686/controlling-orientation-in-3d-pgf-plots`

pgfplotstodo.tex:2446 **Bug** [open, Priority 2, Epic 'boxplot']

boxplots: predefined legend entries need to be improved

pgfplotstodo.tex:2459 **Bug** [open, Priority 2, Epic 'gnuplot']

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

pgfplotstodo.tex:2493 **Bug** [open, Priority 2, Epic 'layers']

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=...}`)

pgfplotstodo.tex:2656 **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

pgfplotstodo.tex:2788 **Bug** [open, Priority 2, Epic 'external']

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

pgfplotstodo.tex:2824 **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 ...

pgfplotstodo.tex:2835 **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?

pgfplotstodo.tex:2845 **Bug** [open, Priority 2, Epic ‘’]

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

pgfplotstodo.tex:2849 **Bug** [open, Priority 2, Epic ‘’]

contour external doesn’t handle the ordering flag correctly.

pgfplotstodo.tex:2864 **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:3020 **Bug** [open, Priority 2, Epic ‘’]

interp shader is displayed transparently in evince

Bug in evince (for sure). What is to be done?

pgfplotstodo.tex:1147 **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 `\
=13 \edef -{\string -}\fi \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?

TODOsp: With "xcolor 2016/05/11 v2.12" it is still present.

Are you sure that this is caused by 'xcolor'? It seems that this messages are caused by the `.` in the value of `mark list fill`. If you provide a value without a dot these messages won't appear.

If you are sure that this could be solved in 'xcolor' let me know and I'll contact the corresponding author and ask him for help.

pgfplotstodo.tex:1851 **Bug** [open, Priority 1, Epic ''']

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

pgfplotstodo.tex:1985 **Bug** [open, Priority 1, Epic ''']

contour plot handler does not allow to specify draw and fill – which might be useful if just one contour is drawn/filled

pgfplotstodo.tex:2365 **Bug** [open, Priority 1, Epic ''']

`\usepackage[gray]{xcolor}` does not seem to work as good as for cmyk... in particular, the default color map is not being converted correctly!? Probably because xcolor does not convert the colors if they are RGB

This appears to be irrelevant because RGB gray colors can be translated losslessy to scalar gray values!?

pgfplotstodo.tex:2508 **Bug** [open, Priority 1, Epic ''']

Internal coordmath framework: it is not used everywhere

pgfplotstodo.tex:2515 **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:2678 **Bug** [open, Priority 1, Epic ‘’]

CRASH:

commented minimal, because it raises an error when TeXing

pgfplotstodo.tex:2726 **Bug** [open, Priority 1, Epic ‘’]

Using 0 in pgfplots coordinate systems does not necessarily mean “no offset”. This is misleading. Bug sourceforge #3168030:

commented minimal, because it raises an error when TeXing

pgfplotstodo.tex:2732 **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:2737 **Bug** [open, Priority 1, Epic ‘’]

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

pgfplotstodo.tex:2840 **Bug** [open, Priority 1, Epic ‘’]

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

pgfplotstodo.tex:2898 **Bug** [open, Priority 1, Epic ‘’]

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

pgfplotstodo.tex:3054 **Bug** [open, Priority 1, 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:3066 **Bug** [open, Priority 1, Epic ‘’]

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

pgfplotstodo.tex:3082 **Bug** [open, Priority 1, Epic ‘’]

javascript: incompatibility 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.

6 Feature Proposals PGFPlots

pgfplotstodo.tex:7511 **Feature Proposal** [open, Priority 6, Epic ‘’]

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

... perhaps combined with a matrix-like input file as in
`http://tex.stackexchange.com/questions/123719/drawing-a-large-binary-matrix-as-colored-grid-in-tikz`

imagesc

`http://tex.stackexchange.com/questions/255596/imagesc-in-tikz-with-non-symmetric-matrices-and-squared-axis`

`http://tex.stackexchange.com/questions/124276/matlab2tikz-imagesc-tikz-pgfplots-equivalent`

`http://tex.stackexchange.com/questions/255432/plotting-matrix-image-data?lq=1`

`http://tex.stackexchange.com/questions/154051/table-vs-plot-how-to-print-table-of-values-with-colors-representing-values?lq=1`

- 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.

TODOsp: Is that `matrix plot*` and thus this feature request can be closed or are there some points left?

`pgfplotstodo.tex:5155` **Feature Proposal** [open, Priority 5, Epic ‘Usability’]

parse elements of tick positions with math parser

`http://tex.stackexchange.com/questions/187115/`
`with-pgfplots-how-to-manually-enter-ticks-as-fractions`

`http://tex.stackexchange.com/questions/249953/`
`locating-tick-marks-at-integral-multiples-of-pi-2`

`pgfplotstodo.tex:5163` **Feature Proposal** [open, Priority 5, Epic ‘’]

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

TODOsp: This is the same as the previous feature request, right? → merge them

pgfplotstodo.tex:5185 **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!

TODO: check which ones are missing. Some of them have been added in the meantime

TODOsp: This is the same as the previous feature request, right? → merge them

pgfplotstodo.tex:5191 **Feature Proposal** [open, Priority 5, Epic ‘fillbetween’]

fillbetween: accept `soft path={inner x range}` or something like that, compare `http://tex.stackexchange.com/questions/180127/stacking-plots-in-animation-using-fill-between-library-with-dynamic-calculation/180299?noredirect=1#comment416484_180299`

pgfplotstodo.tex:5202 **Feature Proposal** [open, Priority 5, Epic ‘fillbetween’]

allow predicates styles of sorts `every intersection below y={0}` *or, even better:* `every lower intersection segment` – after all, “upper” and “lower” can be identified by means of scalar productions with same directed vector. The directed vector can be plugged in from the axis → think about solutions

see `http://tex.stackexchange.com/questions/207042/pgfplots-addplot-color-depending-on-sign/207157#comment535403_207157`

pgfplotstodo.tex:5211 **Feature Proposal** [open, Priority 5, Epic 'Usability']

Given some PGF point, allow to access the high-level coordinates.

see `http://tex.stackexchange.com/questions/174404/converting-unit-to-coordinate-works-for-x-coordinate-but-not-for-y-coordnate/174443#174443` for a use-case and note that there is a work-around which works for 'pin' in the pgfplots manual - and that is awkward.

pgfplotstodo.tex:5239 **Feature Proposal** [open, Priority 5, Epic 'Usability']

pgfplots math expressions always yield internal FPU formats which confuses users.

Return SCI representation instead.

TO BE CLARIFIED: which operations should be replaced? Replacing 'x filter' and its friends might actually make things worse because FPU functions will no longer accept the argument.

It might be useful to improve the FPU such that declarefunction for a function which does *not* expect FPU arguments accept it.

This could be done by patching `\pgfmath@stack@push@operation` – and check if the argument is a function and that function has a known FPU implementation (its float-backup exists). If not: generate a dummy which converts to fixed points. Use the function's arity!

see `http://tex.stackexchange.com/questions/171263/plotting-a-function-defined-with-pgfmathdeclarefunction/171471#171471`

see `http://texwelt.de/wissen/fragen/3960/fraktale-mit-pgfplots?Seite=1#3993`

I started to work on a solution for the second one, see the uncommented call to `\pgfplots@expression@normalize@floats`

pgfplotstodo.tex:5247 **Feature Proposal** [open, Priority 5, Epic 'Usability']

The distinction into survey phase and visualization phase is technically reasonable, but causes confusion.

Is there a way to simplify loop operations during the survey phase as in `http://tex.stackexchange.com/questions/172917/pgfplots-and-using-axis-cs-to-add-points`?

pgfplotstodo.tex:5260 **Feature Proposal** [open, Priority 5, Epic 'z buffer']

Pgfplots requires user input to distinguish between parameterized 3d plots and matrix-like 3d plots.

Why can't PGFPlots determine automatically if `z buffer=sort` is adequate?

`http://tex.stackexchange.com/questions/172753/pgfplots-and-gnuplot-with-addplot3`

ATTENTION: what about 3d sampled line plots!? this would break with z buffer sort.

pgfplotstodo.tex:5271 **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:5280 **Feature Proposal** [open, Priority 5, Epic ‘polar’]

polar plots should allow to paint generate tick labels for the radius on *both* sides (symmetrically)

see `http://tex.stackexchange.com/questions/323462/can-one-generate-a-boxed-polar-plot-with-pgfplots-and-polaraxis/326741#326741` for an application

pgfplotstodo.tex:5284 **Feature Proposal** [open, Priority 5, Epic ‘’]

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

pgfplotstodo.tex:5290 **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)

pgfplotstodo.tex:5295 **Feature Proposal** [open, Priority 5, Epic ‘polar’]

boxed polar plots `http://tex.stackexchange.com/questions/323462/can-one-generate-a-boxed-polar-plot-with-pgfplots-and-polaraxis/326741#326741`

pgfplotstodo.tex:5313 **Feature Proposal** [open, Priority 5, Epic ‘v2’]

it would be nice to have a way to overcome weaknesses of the old pgfplots without breaking everything.

Idea: `\usepackage{pgfplots2}` ... or `compat=2.0` ?

This affects

- changes covered already by compat level
- key naming conventions, compare
  `http://tex.stackexchange.com/questions/315812/is-there-a-general-rule-when-to-write-pgfplots-keywords-together`
- trig format plots
- plotting tables which have just one column
- ?

pgfplotstodo.tex:5318 **Feature Proposal** [open, Priority 5, Epic ‘’]

add support for `list and make` under windows. Perhaps powershell script or bat file?

pgfplotstodo.tex:5354 **Feature Proposal** [open, Priority 5, Epic ‘’]

Loading matrix data which contains just Z coordinates is quite involved.

- streamline defaults which allow to read scaline number and the index in the current scanline as x- resp y-coordinates
- think about a ”plain” matrix input routine

The current implementation already contains support for

```
\begin{tikzpicture}
    \begin{axis}
    % this yields a 3x4 matrix:
    \addplot3[surf] table[x expr=\pgfplotsscanlinecurrentlength,y expr=\pgfplotsscanlinein
 0
 0
 0
 0

 0
 0.6
 0.7
 0.5

 0
 0.7
 0.8
```

```
 0.5
        };
    \end{axis}
\end{tikzpicture}
```

pgfplotstodo.tex:5362 **Feature Proposal** [open, Priority 5, Epic ‘’]

One question that I got was about support for hexbin plots. Something along the lines of this: `https://cran.r-project.org/web/packages/hexbin/vignettes/hexagon_binning.pdf` Gadfly.jl also supports it: `http://dcjones.github.io/Gadfly.jl/geom_hexbin.html`

pgfplotstodo.tex:5387 **Feature Proposal** [open, Priority 5, Epic ‘Stefan Pinnow CM’]

simplify visualization of barymetric surface data

Example date can be retrieved from

`http://ferret.pmel.noaa.gov/NVODS/UI.vm#panelHeaderHidden=false;differences=false;autoCont`
`10146;globalMax=6096;xCATID=CAFA99223222D59FB6BF29705FAFDDBE;xDSID=dbdb5_top_bathy_nc;vari`

`http://pyhogs.github.io/colormap-bathymetry.html`

To do:

- test matrix input routine
- document matrix input routine
- finalize ideas about ‘colormap to legend’
- test ‘target pos min’ and its friends
- document ‘target pos min’ and its friends
- styles for logarithmic samples?

TODOsp: What of these items is already implemented/finished?

pgfplotstodo.tex:5426 **Feature Proposal** [open, Priority 5, Epic ‘Stefan Pinnow CM’]

add support to build a *nonuniform* colormap of a colormap

- on input, the user provides a list of non-uniform *positions* and (optionally) a colormap name

- pgfplots samples a suitable number of colors (one per position) *uniformly* from the source colormap
- each uniformly sampled number is assigned to the associated *target* position in the resulting map

Related, but not the same:

- on input, the user provides a list of non-uniform *positions* and (optionally) a colormap name
- pgfplots uses this list of positions for both *source* and *target*
- this will be necessary for contour filled

look-and feel:

```
    colormap={name}{colors of colormap={of xyz, target pos=-
1000,-300,-200, range=-2000:+20000}}}
    colormap={name}{colors of colormap={of xyz, target pos=0,300,800}}}
    ... I suppose that this key-value syntax _needs_ an ' of ' statement somewhere:
    colormap={name}{colors of colormap={of current, target pos=0,300,800}}}
    colormap={name}{colors of colormap={{-1000,-300,-200} of xyz, range=-
2000:+20000}}}

    colormap={name}{colors of colormap={{-1000,-300,-200}, range=-
2000:+20000}}}
    or perhaps it also needs ' of ':
    colormap={name}{colors of colormap={-1000,-300,-200 of current, range=-
2000:+20000}}}
```

TODOsp: What of these items is already implemented/finished?

pgfplotstodo.tex:5430 **Feature Proposal** [open, Priority 5, Epic 'Stefan Pinnow CM']

allow colormaps of size 1

pgfplotstodo.tex:5461 **Feature Proposal** [open, Priority 5, Epic 'Stefan Pinnow CM']

allow to define colormaps with arbitrary positions which are then mapped to [0,1000]

something like

```
    colormap={name}{color(-1000)=(red) color(1e6)=(blue)}
```

would have the same effect as

```
colormap={name}{color(0pt)=(red) color(1pt)=(blue)}
```

Ideas:

- eliminate restriction that it must start at 0 ✓
- eliminate pgfmathapproxequalto
- eliminate TeX arithmetics in reinterpolate
  is no problem: it only applies TeX arithmetics to the fractional part
- activate FPU for the parsing procedure (only)

TODOsp: What of these items is already implemented/finished?

pgfplotstodo.tex:5539 **Feature Proposal** [open, Priority 5, Epic 'Stefan Pinnow CM']
Allow to build colormaps based on what we see in a colorbar (not map!)

something like

```
colormap={name}{colors of colorbar={-1000,-4000} of xyz}}
```

Ideas:

- color interpolation is based on the current value of `colormap access`
- uses the current `point meta min` and `max`

Idea: also add

```
    colormap={name}{color of colormap=-1000 of xyz using -
2000:+20000}}
    colormap={name}{color of colormap=-1000 using -2000:+20000}}
    colormap={name}{color of colormap=-1000 using -2000:+20000 of xyz}}
```

or (better!):

```
    colormap={name}{color of colormap={-1000 of xyz using range=-
2000:+20000}}}
    colormap={name}{color of colormap={-1000 using range=-
2000:+20000}}}
```

I suppose “range” is better than “domain” in this context

the general form could be

```
  colormap={name}{color of colormap={<arg> using <keys>}}}
  colormap={name}{color of colormap={<arg> of <name> using <keys>}}}
  or even
  colormap={name}{of colormap={<keys>}}
  Example:
  color of colormap={500 of xyz}
->of colormap={name=xyz, access=map, selection=500},

  index of colormap={5 of xyz}
->of colormap={name=xyz, access=direct, selection=5},

  indices of colormap={5,7 of xyz}
->of colormap={name=xyz, access=direct, selection={5,7}},

  const colors of colormap={550,700 of xyz}
->of colormap={name=xyz, access=piecewise constant, selection={550,700}},
```

valid `<keys>` could be

```
  colormap access
  range
  ?
```

TODO:

- document that curly braces *and* round braces are supported
- document new syntax elements
- provide consistent feature for both colormap and cycle list
- document new `of colormap={initial}`
- implement that positions are in sync with the target positions of a const plot. This is non-trivial

TODOsp: What of these items is already implemented/finished?

pgfplotstodo.tex:5556 **Feature Proposal** [open, Priority 5, Epic ‘Stefan Pinnow CM’]

`colorbar style={ytick data}`

`http://pyhogs.github.io/colormap-bathymetry.html`

TODO:

- implement ✓
- test
- document
- document `samples at colormap pos`

TODOsp: What of these items is already implemented/finished?

pgfplotstodo.tex:5576 **Feature Proposal** [open, Priority 5, Epic 'Stefan Pinnow CM']
color of colormap mit piecewise const

Ideas:

- add a dedicated input method of sorts `colormap={name}{const of colormap={520}}`
  ✓
  To be done: tests
- inside of `color of colorbar` (see above): deduce interpolation order automatically from the settings of the colorbar

TODOsp: What of these items is already implemented/finished?

pgfplotstodo.tex:5580 **Feature Proposal** [open, Priority 5, Epic '']
implement waterfall charts (?)

pgfplotstodo.tex:5631 **Feature Proposal** [open, Priority 5, Epic '']
finish + document `plot yshift auto` (idea: allow something like bar shift, but for all plots and including marker support)

finish + document `axis plot except legend style`

Idea: use that in order to also allow markers in

```
\documentclass{standalone}

\usepackage{pgfplots}

\pgfplotsset{compat=1.12}

\pgfplotstableread{
```

```
Year    Cat1  Cat2    Cat3   Cat4
2005    10     50      -10     30
2006    -40     60      -15     90
2007    -20     60      -15     60
}\mytable

\begin{document}%

  \begin{tikzpicture}
  %\tracingmacros=2 \tracingcommands=2
    \begin{axis}[
      xbar stacked,
     % area legend,
      xmajorgrids,
      legend pos=outer north east,
      bar width=10pt,
      bar shift auto,
      nodes near coords,
      nodes near coords style={font=\tiny},
      enlarge y limits=0.3,
      extra x ticks={0},
      extra x tick style={grid style={black},xticklabel=\empty},
      ]
      \addplot+[mark=*] table [x index=1,y=Year] {\mytable};
      \addplot+[mark=otimes] table [x index=2,y=Year] {\mytable};
      \addplot+[mark=square*] table [x index=3,y=Year] {\mytable};
      \addplot table [x index=4,y=Year] {\mytable};
      \legend{Cat1,Cat2,Cat3,Cat4}
    \end{axis}
  \end{tikzpicture}

\end{document}
```

pgfplotstodo.tex:5636 **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:5641 **Feature Proposal** [open, Priority 5, Epic ‘’]

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

pgfplotstodo.tex:5657 **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

TODOsp: Is this already implemented? Is it related to `clip mode=individual`?

pgfplotstodo.tex:5707 **Feature Proposal** [open, Priority 5, Epic ‘’]

optimization ideas:

- replace `\pgfpointscale` with a 'q' version ⇝ 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)
- 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:5712 **Feature Proposal** [open, Priority 5, Epic '']

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

pgfplotstodo.tex:5718 **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:5723 **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:5730 **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. $\rightsquigarrow$ it might be useful to use pgfmathparse for any numerical input argument as well.

pgfplotstodo.tex:5772 **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:5778 **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:5782 **Feature Proposal** [open, Priority 5, Epic '']

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

pgfplotstodo.tex:5789 **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:5822 **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 two-dimensional 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 implemented 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!

pgfplotstodo.tex:5832 **Feature Proposal** [open, Priority 5, Epic 'polar']

polar axes and `axis x line shift` (i.e. shift the circles).

Ideas:

modify the `#1` argument of `\pgfplots@drawgridlines@onorientedsurf@fromto@polar`. Furthermore, try to inject a "b axis shift" into the associated tick labels ...

pgfplotstodo.tex:5846 **Feature Proposal** [open, Priority 5, Epic '']

add expressions 'row' and 'col' which expand to the current row/col index of an input matrix.

Implementation is ready on branch `rowcolgetters`, with the following todo list left:

1. LUA backend
2. docu
3. `empty line=jump` is broken!

pgfplotstodo.tex:5851 **Feature Proposal** [open, Priority 5, Epic 'fillbetween']

fillbetween: add support to style the "positive" and "negative" region, respectively.

pgfplotstodo.tex:5872 **Feature Proposal** [open, Priority 5, Epic '']

`http://tex.stackexchange.com/questions/247973/customize-length-of-y-major-grids-in-pgfplots`

tufte range frames and easier adoptions of grid lines

`http://tex.stackexchange.com/questions/165766/tufte-style-range-frames-for-three-dimensional-plots/165889#165889`

`http://tex.stackexchange.com/questions/125444/draw-reduced-graph-with-pgfplot`

TODOsp: Up to here this should already be implemented as `axis line shift`, right?

see experimental git branch `range_frames` and the mail of Juernjakob Dugge

related:

`http://tex.stackexchange.com/questions/264222/is-there-a-way-to-automatically-force-xticks-on-first-and-last-data-point-in-pgf`

pgfplotstodo.tex:5883 **Feature Proposal** [open, Priority 5, Epic '']

Add option of sorts "shift along outer normal vector of current axis" while generating paths for axes, ticks, tick labels, and perhaps even grid lines.

`http://tex.stackexchange.com/questions/165766/tufte-style-range-frames-for-three-dimensional-plots`

TODOsp: not sure what you are writing here, but the link is the same as one in the previous feature request. Is it closed?

pgfplotstodo.tex:5900 **Feature Proposal** [open, Priority 5, Epic '']

implement special math formulas for cube roots `http://tex.stackexchange.com/questions/69411/pgfplots-cant-plot-some-usual-mathematical-functions`. More generally: solve

$$y(x) = x^{1/n}$$

for odd n and negative x: in this case, it holds that

$$y(x) = -|x|^{1/n}$$

also implement feature requests to PGF with gamma function etc

`http://tex.stackexchange.com/questions/281865/part-of-function-domain-omitted-using-pgfplots`

pgfplotstodo.tex:5906 **Feature Proposal** [open, Priority 5, Epic ‘’]

allow a simple way to connect coordinates of different axes, i.e. something like `(axis1 cs:1,1) -- (axis 2 cs:3,4)` (Tim Esser, per Mail Apr. 28, 2015)

pgfplotstodo.tex:5911 **Feature Proposal** [open, Priority 5, Epic ‘’]

add `enlarge limits=to next tick label`, see `http://tex.stackexchange.com/questions/229551/enlarge-limits-to-nearest-tick`

pgfplotstodo.tex:5950 **Feature Proposal** [open, Priority 5, Epic ‘’]

allow `#` inside of inline tables: the following does not work in any pgfplots version:

```
\documentclass[border=5pt]{standalone}
\usepackage{pgfplots}
    \pgfplotsset{
        compat=newest,
    }
\begin{document}
\begin{tikzpicture}
    \begin{axis}[
        % point meta=explicit symbolic,
    ]
        \addplot table [
            x index={1},
            y expr=\coordindex,
%            header=false,      % <-- SP: that cannot word then, can it?
%            meta index={0},    % <-- SP: what should that be good for?
        ] {
            #Re  750K
            "i"  62.099
            "i"  62.046
            "i"  56.304
            "i" -44.258
            "2" -28.826
```

```
            "i"  18.740
            "i" -14.653
            "i"  14.402
            "i" -12.907
            "i"  12.295
        };
    \end{axis}
\end{tikzpicture}
\end{document}
```

pgfplotstodo.tex:5962 **Feature Proposal** [open, Priority 5, Epic ‘’]

write a script which applies the entire “release todo” over night:

- clone into tmp repo, compile manual from scratch with latest PGF
- clone into different tmp repo, compile tests against latest PGF
- same for all supported PGF versions
- improve reporting somehow (log files and summaries somewhere)

pgfplotstodo.tex:5969 **Feature Proposal** [open, Priority 5, Epic ‘’]

stream plots: contour plots in which each contour line is actually of type `mesh` with a specific value of `point meta`

`http://stackoverflow.com/questions/8296617/how-to-plot-a-streamlines-when-i-know-u-and-v-components-of-velocitynumpy-2d`

pgfplotstodo.tex:5973 **Feature Proposal** [open, Priority 5, Epic ‘’]

lua backend support for `restrict expr to domain`

pgfplotstodo.tex:6187 **Feature Proposal** [open, Priority 5, Epic ‘’]

improve the outcome of `log ticks with fixed point` if the data range is of order $O(1)$.

`Hi Benjamin Voigt,`

`thanks for the detailed email!`

`A appreciate the level of detail for this feature request, and I am glad that my package proves to be useful to you!`

You asked for preferences: if possible, I always prefer reduced (optimally: minimal) examples to work with. In your case, the suggested list of tick labels is quite a good start.

The feature request(s) are, of course, closely related. The overall theme is "improve log ticks with fixed point if the outcome would either have almost no ticks or would have strange fractional numbers", right? In fact, it sounds a lot like switch everything to integer exponents or no exponents at all -- together with the well known feature from linearly scaled plots, i.e.\@ scaled ticks (extract common factors), SI units. Your last item "getting into a dream" (use 1M if the majority of other ticks has that format as well) is actually known to me and I started to implement some suitable style some time ago. But it never made it into axis formatting (it ended up in some other area, perhaps pgfplotstable).

I believe I understand most of them, although I would need to think through it more carefully once I would start with an implementation (see below for "if", "when" and "how").

First, let me make some small comments.

1. I believe the requirement list for an algorithm which chooses tick positions + labels is as follows

   - generate tick labels automatically
   - make it clear that they are in log format
   - ensure that there are enough tick labels to fill the axis
   - the algorithm has to work reasonably on _every_ possible order of magnitude
   - ensure that the tick labels have some commonly recognized form

   You see that I made rather abstract formulations. The current implementation in pgfplots which results in something like
   10^{-3},10^{-2},10^{-1},10^0
   10^{-3},10^0,10^3,10^6
   10^{-1.3},10^{-1},10^{-0.7},10^{-0.3},10^0
   addresses these issues using the standard exponential notation.

   Why do I say that? Well, improving the quality of log ticks with fixed point for O(1) number ranges is a good idea, so I accept the feature request. On the other hand, the existing algorithm does a quite good job at satisfying the general requirements (I understand that the results for log ticks with fixed point are quite bad in your data regime). That makes it more of a border case, i.e.\@ I have feature requests which appear to have higher priority.

2. This is really just a side note: you may want to take a look at the existing key "log identify minor tick positions" in the reference manual. There is also some tick fine tuning in section "4.15.4 Tick Fine-
Tuning" (the number
   may vary depending on your version).

3. The current tick placement algorithm has no predefined lists of real numbers. It operates on a more general way such that it can satisfy the requirement to work on all possible orders of magnitude. The routine works quite well, although it has become somewhat dusty... . It is in pgfplotsticks.code.tex and is called \pgfplots@assign@default@tick@foraxis . In contains lots of uncommented debug messages along with comments, so there is a chance to see what's going on. Note that pgfplots currently just has either linear axes or log axes, and if the boolean "is linear" is false, it is assumed to be logarithmic.

My current priorities in the development of pgfplots are as follows:
Release 1.11 is the current public stable.

Release 1.12 will have the theme "scalability and performance". It will come with a new backend for lualatex which is better suited for huge data sets; it will also contain important maintenance topics (bug fixes).

Release 1.13 (or whatever it will be called) will probably become a feature release to address hundreds of open requests around bar plots (and maintenance as usual).

I typically take six months per release; and 1.12 is planned for end of this year. That means: it might take some time until I can look into this feature request; some might be up to one year from now.

The effort as such is probably quite low; it's just that I have to make a strong focus to address pgfplots items in my limited spare time.

In other words: if you are willing to invest time on this special project, I would be willing to test and accept code contributions.

Entry points would be
http://tex.stackexchange.com/questions/12668/where-do-i-start-latex-programming/27589#27589
and the routine mentioned in Note 3 above. It might be worth a try. Adding additional keys to pgfplots is relatively simple; if in doubt, the pgfmanual contains an exhaustive reference section on pgfkeys.

I will copy this email to my todo list (I assume your suggestions and thought in this mail can be viewed by anybody) and will come back to it according to my time schedule outlined above.

What do you think?

Kind regards

Christian

PS

```
Your name sounds like german origins - or some other european country. Do you
come from here (meaning: germany)?

Am 26.10.2014 04:08, schrieb richardvoigt@gmail.com:
>
> Hi Dr. Feuersänger,
>
> To being with, thanks *very* much for all the hard work you've done on
> pgfplots already, and for making that available for free.  Without that
> package, I wouldn't dare try to make plots from inside LaTeX, everything would
> be exported as EPS or PNG from MATLAB, and not looking nearly as good.
>
> This is related to the Tex.SE post "pgfplot log axis more than one tick label
> per decade (1,2,5,10)"
> http://tex.stackexchange.com/questions/208891/pgfplot-log-axis-
more-than-one-tick-label-per-decade-1-2-5-10
>
> Great idea moving to email, among other advantages I can share unredacted
> pieces of my thesis privately with you, while I can't post those in public
> prior to publication.  Please let me know your preferences, whether you prefer
> to work with reduced examples, or would take direct excerpts (complete
> preamble, selected data files, and a couple unedited paragraphs/plots from the
> thesis).
>
> My suggestion is that the following sets of labels be made available:
>
> 10^{-3},10^{-2},10^{-1},10^0      current default
> 10^{-3},10^0,10^3,10^6            current default, when range is huge
>
> 0.001, 0.01, 0.1, 1               currently gotten from "log ticks with fixed point"
> 0.1,1,10,100,1000                 currently gotten from "log ticks with fixed point"
> 0.1, 0.2, 0.5, 1, 2, 5            given in my Tex.SE answer
> 0.001,0.002,0.005,0.01,0.02       given in Tex.SE answer
> 50,100,200,500,1000,2000          given in Tex.SE answer
> 0.001,0.003,0.01,0.03,0.1,0.3,1   can use technique from Tex.SE answer
> 10,30,100,300,1000,3000           can use technique from Tex.SE answer
>
> 10^{-1.3},10^{-1},10^{-0.7},10^{-0.3},10^0    use technique from Tex.SE
>                                               answer, removing "log ticks
>                                               with fixed point" / shown in
>                                               your answer -- but, I do not
>                                               like this one because the tick
>                                               pattern does not look
>                                               logarithmic
>
> 1\mu,2\mu,5\mu,10\mu,20\mu,50\mu,100\mu       this would be nice, for values where
> 100k, 200k, 500k, 1M, 2M, 5M, 10M same        "log ticks with fixed point" gives
> 1m, 1, 1k, 1M same, when range is huge        too many zeros
>
> 1 \cdot 10^{-6}, 2 \cdot 10^{-6}, 5 \cdot 10^{-6}, 1 \cdot 10^{-
```

```
5}, 2 \cdot 10^{-5}
> maybe call this "log ticks with integer exponent"
> 1 \cdot 10^{-6}, 3 \cdot 10^{-6}, 1 \cdot 10^{-5}, 3 \cdot 10^{-
5}, 1 \cdot 10^{-4}, 3 \cdot 10^{-4}
> same idea, wider range
>
> Since the magnitude of my data is reasonably close to 1, I prefer the second
> group, and the only problem with the solution shown in my answer is that I have
> to change code for each graph to control whether it is 1, 2, or 3 ticks per
> decade.  Just auto-selecting between 1-2-5-10, 1-3-10, 1-10-
100, and .01-1-100
> would be ideal for me. But I agree that something like scientific notation,
> engineering notation (scientific but the exponent is always a multiple of 3),
> or SI units are needed to keep the labels for very large or very small numbers
> readable.
>
> I just don't like fractional exponents.  In my mind, log plots are for
> presenting data which varies over a wide range, without thinking of the
> logarithm.  For cases where the logarithm has meaning, perhaps because the
> equation involves powers, then the data is converted to logarithmic units (dB)
> and those are plotted linearly.
>
> I also don't have to let my data get far away from zero, because I can label
> my axis as "Time (microseconds)" rather than having 10^{-
6} on every tick.
>
> But I realize that other people may see their data differently, or work with
> different data.  So I've tried to include other "nice" styles in the list of
> labels above, even though I don't anticipate needing them myself.
>
> As you commented, there really are two discrete things here: tick placement,
> and label formatting.
>
> I propose something like "log rational ticks" to switch to whichever of
> 1-10,1-3-10,1-2-5-10 provides the most appropriate number of ticks.
> (rational in the sense of rational numbers)
>
> And then formats of default (mantissa = 1 always, exponent varies), "log
> ticks with fixed place" as already implemented, "log ticks with integer
> exponent".  Bonus would be "log ticks with SI factor" and "log ticks with
> engineering exponent" which force the exponent to a multiple of 3.  Getting
> into dreams -- there should be a heuristic that prevents just a single label
> from having a different exponent, e.g. 10^6 should be written 1000k if the
> other labels are 10k,30k,100k,300k, but 1M if the other labels are 2M,5M,10M.
>
> It seems like pgfplots has a lot of customization points with a path of
> tag/.code  I don't know if any affect tick selection; didn't see anything in
> the manual. But if you point me in the right direction I'm willing to help try
> things.
>
```

```
> Again, thanks for the awesome work that is pgfplots 1.9.  When I say that "I
> doubt anyone is very happy with the results |log ticks with fixed point
> |currently gives for intra-decade ticks" I don't mean to take away from the
> fact that pgfplots does amazingly well, by default, for most graphs.
>
> R Benjamin Voigt
```

pgfplotstodo.tex:6194 **Feature Proposal** [open, Priority 5, Epic ‘’]

Add support for a transformation which allows TIME input data (i.e. without date)

see `http://tex.stackexchange.com/q/79252/18401`

pgfplotstodo.tex:6202 **Feature Proposal** [open, Priority 5, Epic ‘’]

slanted and sloped text (projected onto some axis plane)

see `http://tex.stackexchange.com/questions/212699/text-projection-onto-plane-in-3d-pgf-plots`

this also has math formulas to set up the trafo

pgfplotstodo.tex:6209 **Feature Proposal** [open, Priority 5, Epic ‘external’]

external lib: rerun externalization in case of unresolved references.

see `http://www.texwelt.de/wissen/fragen/9476/labels-an-pgfplots/9527` for a sketch and use-case

pgfplotstodo.tex:6217 **Feature Proposal** [open, Priority 5, Epic ‘’]

allow some simply style of sorts “xtick should have a distance of 10. Get it done.”

TODOsp: This should be implemented as `xtick distance`, right? → close it.

pgfplotstodo.tex:6227 **Feature Proposal** [open, Priority 5, Epic ‘’]

add styles to format seconds using some time format

see `http://tex.stackexchange.com/questions/198706/convert-gnuplot-script-to-pgfplots-using-raw-gnuplot-option`

see `http://tex.stackexchange.com/questions/118676/increasing-dateplots-resolution-to-seconds/219447#219447`

pgfplotstodo.tex:6234 **Feature Proposal** [open, Priority 5, Epic ‘’]

table package: Add style “modify content for the following row indices”

can be copy-pasted from
`http://tex.stackexchange.com/questions/187067/how-to-have-different-colormaps-for-different-columns-in-the-same-heatmaps-table/187099#187099`

pgfplotstodo.tex:6260 **Feature Proposal** [open, Priority 5, Epic ‘’]

ich bin es nochmal. Ich habe nochmal eine Anmerkung zu den decorations. Ich bin gerade dabei, meine Plots mit Pfeilen zu versehen wie ich es in dem Bild zuvor bereits getan habe. Leider kann ich für die decorations nur folgende Angabe machen:

```
mark = at position 0.15 with {\arrow [scale=1]{stealth}},
```

Jedoch kommt es bei Kurven häufiger vor, dass ich nicht gut abschätzen kann, ob es sich dabei um Position 0.1 oder 0.15 oder dergleichen handelt. Ich habe es so verstanden, dass der Compiler die Strecke der Kurve vermisst und bei bspw.

```
mark = at position 0.5 with {\arrow [scale=1]{stealth}},
```

eine Dekoration bei der Hälfte der Kurve macht. Doch wo ist die Hälfte der Kurve, wenn diese gekrümmt und gewunden ist? Nun ist es umständlich immer diese Dekorationen anzupassen und ich würde lieber eine x oder y Koordinate verwenden und sagen können:

```
mark = at x position 40 with {\arrow [scale=1]{stealth}},
```

So kann ich mir sicher sein, dass der Pfeil dann der x-Koordinate 40 zugeordnet wird und auf der Kurve landet.

pgfplotstodo.tex:6270 **Feature Proposal** [open, Priority 5, Epic ‘’]

Unbounded point meta data should be filtered out.

`http://tex.stackexchange.com/questions/164250/pgfplots-surf-plot-dont-draw-nan?lq=1`

However, it seems as if this “filter out” should not be done as for coordinates (which would filter out all adjacent patch segments). It should merely filter out the current one.

pgfplotstodo.tex:6278 **Feature Proposal** [open, Priority 5, Epic ‘graphics’]

addplot graphics: support “scale to natural dimensions” of the input graphics

there is a half-ready solution in
http://tex.stackexchange.com/questions/175618/addplot-graphics-maintaining-images-aspect-ratio-despite-different-scaling-of/175726#175726

pgfplotstodo.tex:6301 **Feature Proposal** [open, Priority 5, Epic ‘’]

bullet graphs:

http://tex.stackexchange.com/questions/146538/how-to-create-vertical-bullet-graphs-with-pstricks

Inspiration:

http://img35.imageshack.us/img35/1061/snap2134.png

I started a prototype. To do items:

- make sure that color schemes can be exchanged in a simple way
- prepare the package for up to 3 discriminative markers
- regarding data files: TO CLARIFY

branch ‘bulletplots’

perhaps this would be a good contribution together with
http://tex.stackexchange.com/questions/29293/is-there-a-package-that-provides-graphing-in-the-style-of-ed-tufte/29311#29311

pgfplotstodo.tex:6309 **Feature Proposal** [open, Priority 5, Epic ‘’]

add “stackable tick labels” (tick labels with vertical shifts if they are too close).

compare http://tex.stackexchange.com/questions/164673/positioning-even-or-odd-x-axis-tick-labels-in-pgfplots

pgfplotstodo.tex:6319 **Feature Proposal** [open, Priority 5, Epic ‘’]

Allow to shift (x) tick labels vertically if they are too dense.

Sometimes one needs lots of tick labels. It would be cool if every second would be shifted vertically, perhaps with an edge to the original location

TODOsp: This is the same as the previous feature request, right? → merge them

pgfplotstodo.tex:6354 **Feature Proposal** [open, Priority 5, Epic ‘’]

suppress tick-line cross in 3d for view directions in which the cross degenerates to a very thick line

compare

```
\documentclass[tikz,12pt]{standalone}
\usepackage{tikz,pgfplots,pgfplotstable}

\pgfplotsset{compat=1.10}

\pgfplotsset{every axis/.append style={tick style={line width=0.7pt}}}

\pgfplotstableread{
a b
-0.1 0.2
0.1 0.5
}\testdata

\begin{document}
\foreach \h in {5,10,...,360} {
\begin{tikzpicture}

\begin{axis}[title=\h,axis lines=center, ymin=-0.22,ymax=0.22, xmin=-
0.2,xmax=0.2,xlabel=x ,view/h=\h ]
    \addplot3 [color=blue,] table[x expr=0,y=a,z=b] {\testdata};
\end{axis}

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

pgfplotstodo.tex:6379 **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:6391 **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:6398 **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-tikzpicture

pgfplotstodo.tex:6413 **Feature Proposal** [open, Priority 5, Epic ‘v2’]

improve grid line placement for grouped bar plots: grid lines should not pass through the middle of the bar group

http://tex.stackexchange.com/questions/248807/draw-grid-lines-for-grouped-bar-chart-diagram

TODOsp: I also think it should be the default that ticks should be *in between* the bars instead of in the middle of the bars. Maybe this could easiest be implemented by using the minor ticks for that and just don't draw any major ticks.

This should also have then no impact if one also wants to draw some other stuff that are not bars, e.g. lines.

pgfplotstodo.tex:6421 **Feature Proposal** [open, Priority 5, Epic ‘Bar plots’]

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

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

(stacked and clustered bar charts)

pgfplotstodo.tex:6430 **Feature Proposal** [open, Priority 5, Epic ‘Bar plots’]

staggered tick labels

http://peltiertech.com/Excel/Charts/Staggered.html

implement a feature which applies a “suitable” `yshift` to every second tick label such that they can be denser

pgfplotstodo.tex:6468 **Feature Proposal** [open, Priority 5, Epic 'Bar plots']

interrupted bar plots

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.

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`

pgfplotstodo.tex:6473 **Feature Proposal** [open, Priority 5, Epic 'Bar plots']

bar plots: Introduce significance stars (see GROUP BARS on page 103)

pgfplotstodo.tex:6763 **Feature Proposal** [open, Priority 5, Epic 'Bar plots']

GROUP BARS

Bar plots: simplify grouped bars

related: suppress bar shift if one group element is missing `http://tex.stackexchange.com/questions/71350/two-level-labels-in-bar-plot`

related: Nested groups: `http://tex.stackexchange.com/questions/198943/unclutter-a-bar-plot-with-pgfplot`

Hi Joshua,

as already mentioned, your old mails from January have slipped my discipline...
I am sorry for not answering right away. I should have written "sorry, I am
busy please also try tex.stackexchange" after there was no answer for some
time.

You asked for a simpler way to
a) position groups of bars without having to compute 'bar shift' manually
b) a simpler way to get "significance stars".

First, grouped bar charts are an open feature request in pgfplots, and your
input will eventually prove to be valuable as definition of requirements. I
haven't thought much on significance stars so far; I fear your current solution
which fixes the y location is the best at hand. There may be ways to write
complicated styles which implement that feature, but that's clearly advanced.

The first request is actually possible, assuming that you always provide them
in the sequence as you did: I copied the default definition of 'bar shift' and
adopted it.

The default is

```
    /pgfplots/ybar/.style={
        /pgf/bar shift={%
            % total width = n*w + (n-1)*skip
            % -> subtract half for centering
            -0.5*(\numplotsofactualtype*\pgfplotbarwidth + (\numplotsofactualtype-
1)*#1)  +
            % the '0.5*w' is for centering
            (.5+\plotnumofactualtype)*\pgfplotbarwidth + \plotnumofactualtype*#1},%
    }
```

and my modification substitutes every index and 'n' by half of it:

```
\pgfplotsset{
    % #1 = separation between bars
    bar shift for half number plots/.style={%
        /pgf/bar shift={%
            % total width = n*w + (n-1)*skip
            % -> subtract half for centering
            -0.5*(\numplotsofactualtype/2*\pgfplotbarwidth + (\numplotsofactualtype/2-
1)*#1)  +
            % the '0.5*w' is for centering
            (.5+div(\plotnumofactualtype,2))*\pgfplotbarwidth + div(\plotnumofactualtype,2
        },%
    },%
    bar shift for half number plots/.default=2pt,
}
```

With this definition, you do not have to write 'bar shift' in your styles.

The whole figure becomes

```
\usemodule[pgfplots]
\pgfplotsset{compat=newest}

\pgfplotsset{HeartControl/.style=
{
    red, fill=red!33!white,
    %bar shift=-0.1667
}}
\pgfplotsset{HeartDiabetes/.style=
{
    red!33!black, fill=red!66!white,
    %bar shift=+0.1667
}}
\pgfplotsset{KidneyControl/.style=
{
    orange, fill=orange!33!white,
    %bar shift=-0.1667
}}
\pgfplotsset{KidneyDiabetes/.style=
{
    orange!33!black, fill=orange!66!white,
    %bar shift=+0.1667
}}

\pgfplotsset{/pgfplots/ybar legend/.style=
{
    /pgfplots/legend image code/.code={%
        \draw[
            ##1, /tikz/.cd,
            bar width=0.25em,
            yshift=-0.27em,
            bar shift=0pt
        ]
        plot coordinates {(0pt,0.8em)};
    }
}}

\pgfplotsset{
    % #1 = separation between bars
    bar shift for half number plots/.style={%
        /pgf/bar shift={%
            % total width = n*w + (n-1)*skip
            % -> subtract half for centering
            -0.5*(\numplotsofactualtype/2*\pgfplotbarwidth + (\numplotsofactualtype/2-
1)*#1)  +
            % the '0.5*w' is for centering
```

```
            (.5+div(\plotnumofactualtype,2))*\pgfplotbarwidth + div(\plotnumofactualtype,2
        },%
    },%
    bar shift for half number plots/.default=2pt,
}

\starttext

\starttikzpicture
    \startaxis
    [
        bar width=0.3,
        ybar,
        bar shift for half number plots=5pt,
        xtick=data,
        ylabel={mRNA level},
        ymin=0,
        xmin=0.333, xmax=2.667,
        xtick={1,2}, xticklabels={Heart,Kidney},
        error bars/y dir=both,
        error bars/y explicit,
        legend columns=2,
        legend pos=outer north east,
        legend cell align=left
    ]

    \addplot+[HeartControl] coordinates {(1,1) +- (0,0.1)};
    \addplot+[KidneyControl] coordinates {(2,1) +- (0,0.12)};
    \addplot+[HeartDiabetes] coordinates {(1,1.1) +- (0,0.2)};
    \addplot+[KidneyDiabetes] coordinates {(2,0.8) +- (0,0.05)};

    \draw (axis cs:2,1.15) +(-2.2em,0) -- +(2.2em,0);
    \node[anchor=south, yshift=-1ex] at (axis cs:2,1.15) {*};

    \legend{{\kern-0.1em}, Control, {\kern-0.1em}, Diabetes}

    \stopaxis
\stoptikzpicture
\stoptext
```

I suppose you could even simplify the styles by means of a cycle list or whatever.

I also experimented with symbolic x coords, but how would you write "xmin=0.3333" or "bar width=0.3" (which are axis units)? I believe the solution with arbitrary constants is better (i.e.\@ as you have it right now).

You can use

```
\def\heartUnit{1}
```

```
\def\kidneyUnit{2}

to introduce constants - this might make it more readable.


I am aware of the fact that this solution comes way too late. Perhaps it proves
to be useful for someone eventually.

Kind regards

Christian



Am 21.01.2014 13:29, schrieb Joshua Krämer:
> Dear pgfplots developers!
>
> First, thanks a lot for your great package.  I'm using it to create
> diagrams in ConTeXt.  Please consider the following (M)WE.  I hope you
> can run it, otherwise, you can see the output here:
> http://666kb.com/i/cl5sdm34i4ig69mkk.png
>
> \usemodule[pgfplots]
> \pgfplotsset{compat=newest}
>
> \pgfplotsset{HeartControl/.style=
> {
>     red, fill=red!33!white,
>     bar shift=-0.1667
> }}
> \pgfplotsset{HeartDiabetes/.style=
> {
>     red!33!black, fill=red!66!white,
>     bar shift=+0.1667
> }}
> \pgfplotsset{KidneyControl/.style=
> {
>     orange, fill=orange!33!white,
>     bar shift=-0.1667
> }}
> \pgfplotsset{KidneyDiabetes/.style=
> {
>     orange!33!black, fill=orange!66!white,
>     bar shift=+0.1667
> }}
>
> \pgfplotsset{/pgfplots/ybar legend/.style=
> {
>     /pgfplots/legend image code/.code={%
>         \draw[
```

```
>             ##1, /tikz/.cd,
>             bar width=0.25em,
>             yshift=-0.27em,
>             bar shift=0pt
>         ]
>         plot coordinates {(0pt,0.8em)};
>     }
> }}
>
> \starttext
>
> \starttikzpicture
>     \startaxis
>     [
>         ybar,
>         xtick=data,
>         ylabel={mRNA level},
>         ymin=0,
>         xmin=0.333, xmax=2.667,
>         xtick={1,2}, xticklabels={Heart,Kidney},
>         error bars/y dir=both,
>         error bars/y explicit,
>         bar width=0.3,
>         legend columns=2,
>         legend pos=outer north east,
>         legend cell align=left
>     ]
>
>     \addplot+[HeartControl] coordinates {(1,1) +- (0,0.1)};
>     \addplot+[KidneyControl] coordinates {(2,1) +- (0,0.12)};
>     \addplot+[HeartDiabetes] coordinates {(1,1.1) +- (0,0.2)};
>     \addplot+[KidneyDiabetes] coordinates {(2,0.8) +- (0,0.05)};
>
>     \draw (axis cs:2,1.15) +(-2.2em,0) -- +(2.2em,0);
>     \node[anchor=south, yshift=-1ex] at (axis cs:2,1.15) {*};
>
>     \legend{{\kern-0.1em}, Control, {\kern-0.1em}, Diabetes}
>
>     \stopaxis
> \stoptikzpicture
>
> \stoptext
>
> As you can see, I have two groups (organs: hearts, kidneys), sometimes
> more, and two conditions (control, Diabetes).  To make it easier to
> compare the diagrams (there are many), I want to use consistent colors
> for the same organs, and two brightnesses for the two conditions.  The
> code above works, but automatic positioning would be much nicer, so I
> could just set something like "bar width=..." and "bar seperation=..."
> and let the bars be positioned automatically.  This would also allow me
```

```
> to use symbolic coordinates, avoiding the pseudo coordinates (1 and
> 2).  Is there a better way to get the desired result than what I've
> done?
>
> I also hope there is a better way to create significance stars.  If two
> values are significantly different, there is a horizontal line to be
> added which spans the two bars, and a symbol above it.  The symbols
> usually are one till three stars (depending on the degree of
> significance), sometimes other symbols are used in the literature.  At
> least it would be nice if I could define the coordinates for the
> horizontal line with something like "max(errormark1, errormark2) +a",
> ie, the higher of the two error marks involved plus some separation.
>
> Kind regards,
> Joshua Krämer
>
>
>
> ------------------------------------------------------------------------------
> CenturyLink Cloud: The Leader in Enterprise Cloud Services.
> Learn Why More Businesses Are Choosing CenturyLink Cloud For
> Critical Workloads, Development Environments & Everything In Between.
> Get a Quote or Start a Free Trial Today.
> http://pubads.g.doubleclick.net/gampad/clk?id=119420431&iu=/4140/ostg.clktrk
> _____________________________________________
> Pgfplots-features mailing list
> Pgfplots-features@lists.sourceforge.net
> https://lists.sourceforge.net/lists/listinfo/pgfplots-features
```

pgfplotstodo.tex:6801 **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-Definition die Bedingung ybar angibt. Ich finde diese Ausrichtung sehr unflexible, da ich sehr oft über das Problem stolpere, dass ich in meinem gruppierten Säulendiagramm noch eine waagrechte Linie oder ähnliches 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 hätte ich 2 Vorschläge. Zum einen die Legende "freier" zu gestalten. Also so, dass man beliebig (ggf. auch ohne Plot) ein Legendenelement hinzufügen kann und vlt. noch das zugehörige Symbol festlegen kann. (Bis jetzt habe ich das Problem, das ich mit tricksen zwar meine Obere Toleranzgrenze in die Legende bekomme, dann jedoch mit einem Säulenzeichnen davor). Der andere Vorschlag ist, dass Säulendiagramm anders zu definieren. 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!

grouped + stacked bar: `http://tex.stackexchange.com/questions/71350/two-level-labels-in-bar-plot`

TODOsp: similar to the previous feature request (at least the link is also in there).

pgfplotstodo.tex:6811 **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:6816 **Feature Proposal** [open, Priority 5, Epic 'Bar plots']

Allow 3d bar plots (see also `http://tex.stackexchange.com/questions/176347/2d-bar-chart-in-3d-space/176363#176363`)

pgfplotstodo.tex:6837 **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 dependent on point meta (like scatter plots)

There is some preparation key `at begin bar` combined with `at end bar` in the bar plot handlers. It can be used as low–level backend, but it still needs to be worked out (see `unittest_bar_shade_atbeginbar.tex`)

http://tex.stackexchange.com/questions/116765/particular-bar-plot-with-pgfplots-bar-color-category

pgfplotstodo.tex:6849 **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.

In particular, the default settings should never hide a bar (especially not half of a bar)
http://w3facility.org/question/spacing-in-pgfplots-bar-plot/

http://tex.stackexchange.com/questions/147945/pgfplots-bar-graph-axis-distance

pgfplotstodo.tex:6874 **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?

see also http://www.faqssys.info/pgf-plots-barplot-irregular-spacing-between-barscolumns/

TODOsp: This seems to be related to the previous feature request. *to* can you merge them?

pgfplotstodo.tex:6886 **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' (?)

TODOsp: This seems to be related to the previous feature request. *to* can you merge them?

pgfplotstodo.tex:6892 **Feature Proposal** [open, Priority 5, Epic ‘’]

allow better default styles for bar plots, compare

`https://www.queryxchange.com/q/24_239801/fontawesome-scaling-deedy-resume-pgfplots-axis-xetex-luatex-issues/`

pgfplotstodo.tex:6899 **Feature Proposal** [open, Priority 5, Epic ‘polar’]

allow to rotate polar plots and fix rotation of tick labels.

see `http://tex.stackexchange.com/questions/116830/polar-plot-x-and-y-ticks-and-units`

pgfplotstodo.tex:6906 **Feature Proposal** [open, Priority 5, Epic ‘polar’]

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

TODOsp: Which feature request shall this be?

pgfplotstodo.tex:6911 **Feature Proposal** [open, Priority 5, Epic ‘Bar plots’]

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

pgfplotstodo.tex:6925 **Feature Proposal** [open, Priority 5, Epic ‘Bar plots’]

bar plots:

- bar interval plot handler which *assumes* uniform distances. This allows to eliminate the last, superfluous 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

pgfplotstodo.tex:6930 **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:6935 **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:6944 **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

TODOsp: This is already closed, right?

pgfplotstodo.tex:6953 **Feature Proposal** [open, Priority 5, Epic 'layers']

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/31713#31713` for a motivation (the tick lines are hidden by the surface)

pgfplotstodo.tex:6961 **Feature Proposal** [open, Priority 5, Epic '']

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

TODOsp: This seems to be implemented, right? → close it

pgfplotstodo.tex:6972 **Feature Proposal** [open, Priority 5, Epic 'external']

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 descriptions)

TODOsp: I think this is related to `http://tex.stackexchange.com/questions/350106/selectively-use-png-or-pdf-for-externalized-tikz-pictures`

pgfplotstodo.tex:6989 **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 would be fixed, one could provide `colorbar style={ymode=log,disablelogfilter}` and would get a proper logarithmic colorbar. Perhaps even combined with `log basis` ...?

TODOsp: Is this still open?

pgfplotstodo.tex:6993 **Feature Proposal** [open, Priority 5, Epic ‘’]

Cases-statement in math parser

pgfplotstodo.tex:6998 **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:7003 **Feature Proposal** [open, Priority 5, Epic ‘’]

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

pgfplotstodo.tex:7007 **Feature Proposal** [open, Priority 5, Epic ‘’]

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

pgfplotstodo.tex:7016 **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:7021 **Feature Proposal** [open, Priority 5, Epic ‘’]

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

pgfplotstodo.tex:7026 **Feature Proposal** [open, Priority 5, Epic ‘’]

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

pgfplotstodo.tex:7030 **Feature Proposal** [open, Priority 5, Epic ‘’]

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

pgfplotstodo.tex:7035 **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:7039 **Feature Proposal** [open, Priority 5, Epic ‘’]

hist does not allow modifications to the data range

pgfplotstodo.tex:7043 **Feature Proposal** [open, Priority 5, Epic ‘’]

feature to replicate axis descriptions on both sides

pgfplotstodo.tex:7050 **Feature Proposal** [open, Priority 5, Epic ‘external’]

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:7055 **Feature Proposal** [open, Priority 5, Epic ‘’]

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

pgfplotstodo.tex:7084 **Feature Proposal** [open, Priority 5, Epic ‘groupplots’]

groupplots: group-wide axis labels

TODOsp: I guess this is either `https://sourceforge.net/p/pgfplots/feature-requests/19/` or `https://sourceforge.net/p/pgfplots/feature-requests/48/`, right? → If so, delete this entry

pgfplotstodo.tex:7095 **Feature Proposal** [open, Priority 5, Epic ‘’]

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

pgfplotstodo.tex:7121 **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 proud 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....

pgfplotstodo.tex:7130 **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 write a coord filter which checks for `\tikz@plot@handler`. Or write pgfplots styles which set them.

pgfplotstodo.tex:7152 **Feature Proposal** [open, Priority 5, Epic ‘polar’]

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.

pgfplotstodo.tex:7156 **Feature Proposal** [open, Priority 5, Epic ‘’]

patch visualization: provide displacement input format

pgfplotstodo.tex:7181 **Feature Proposal** [open, Priority 5, Epic 'quiver']

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 for 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:7185 **Feature Proposal** [open, Priority 5, Epic '']

plot expression: make the sampling parameters available within survey phase

pgfplotstodo.tex:7189 **Feature Proposal** [open, Priority 5, Epic '']

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

pgfplotstodo.tex:7194 **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:7239 **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]  % grün
\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:7249 **Feature Proposal** [open, Priority 5, Epic '']

add something like

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

pgfplotstodo.tex:7262 **Feature Proposal** [open, Priority 5, Epic 'ternary']

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:7275 **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:7284 **Feature Proposal** [open, Priority 5, Epic ‘’]

filled contour plots (prototype is 10% ready)

works by means of gnuplot if the outer region is extended artificially. However, color data needs to be compensated etc.

TODOsp: I guess this is more than 10% ready in the meantime, right?

pgfplotstodo.tex:7293 **Feature Proposal** [open, Priority 5, Epic ‘’]

DUPLICATE 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:7315 **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 artificial path from the data limits.

– idea: in case I know the corner values, I'd know which contour plateau requires the artificial path.
– other idea: I could implement some sort of even-odd rule in TeX. This should also yield the information.

pgfplotstodo.tex:7320 **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:7331 **Feature Proposal** [open, Priority 5, Epic ‘pgfplotstable’]

table package and axes should improve their communication. Namely:

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

pgfplotstodo.tex:7336 **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:7343 **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:7367 **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, (???)
   ```

3. the comma separated list in the legend command's argument applies successively to the two addplot.
4. the colors of the two plots are given by the color cycle list.

pgfplotstodo.tex:7373 **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.

pgfplotstodo.tex:7382 **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.

TODO: activate
`pgfplotsforeachungroupeduniform@loop@mathengine@PRECISE@CURRENTLY@UNUSED@`

pgfplotstodo.tex:7387 **Feature Proposal** [open, Priority 5, Epic ‘’]

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

pgfplotstodo.tex:7391 **Feature Proposal** [open, Priority 5, Epic ‘’]

interpolate missing coordinates for stacked plots.

pgfplotstodo.tex:7395 **Feature Proposal** [open, Priority 5, Epic ‘z buffer’]

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

pgfplotstodo.tex:7432 **Feature Proposal** [open, Priority 5, Epic ‘pgfplotstable’]

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.

pgfplotstodo.tex:7523 **Feature Proposal** [open, Priority 5, Epic ‘’]

imagesc combined with input of sorts

```
     -3   -2   -1   0    1    2    3
0    0    1    2    3    4    5    6
1    1    2    3    4    5    6    7
2    2    3    4    5    6    7    8
3    3    4    5    6    7    8    9
```

pgfplotstodo.tex:7567 **Feature Proposal** [open, Priority 5, Epic ‘’]

Ein weiteres wünschenswertes Feature aus meiner Email ist, dass man die "‘diverging"’ Colorbars von `http://colorbrewer2.org/#` symmetrisch um 0 (also z.B. von −60 bis +60) zuweist, dann aber nur in der Colorbar Werte von −30 bis +60 zeigt, weil eben keine tieferen Werte als −30 vorhanden sind. Damit würde die 0 nach wie vor weiß bzw. gelb bleiben (in den Colorbars von `http://colorbrewer2.org/#`). Der Feature-Request gilt übrigens allgemein für die Colorbar und nicht zwingend für die `colorbar sampled`. Ich glaube, dass war eins der Features, die du aus meiner Email nicht verstanden hast. Wenn dem immer noch so sein sollte, kann ich auch gerne eine

Dummy-Colorbar erstellen und daran zeigen, was ich meine. Manchmal sagt ein Bild ja mehr als 1000 Worte.

TODOsp: I think this feature can already be realized with the following MWE. So from my point of view this can be deleted. Did you have something different/else in mind?

```
\documentclass[border=5pt]{standalone}
\usepackage{pgfplots}
    \usepgfplotslibrary{colorbrewer}
    \pgfplotsset{compat=1.14}
\begin{document}
    \begin{tikzpicture}
        \begin{axis}[
            view={0}{90},
            ymin=-5,
            colormap/RdYlBu,
            colorbar,
            colorbar style={
                ymin=-20,
            },
            domain=-2:5,
        ]
            \addplot3 [
                point meta min=-5,
                surf,
            ] {y};
        \end{axis}
    \end{tikzpicture}
\end{document}
```

pgfplotstodo.tex:7695 **Feature Proposal** [open, Priority 5, Epic 'colormap']

finish `colormap access=direct`

- add colorbar style
- add example of a bitmap image with palette

```
\documentclass{standalone}
\usepackage{pgfplots}
    \pgfplotsset{compat=1.9}
    \definecolor{YlGn-B}{RGB}{255,255,204}
    \definecolor{YlGn-E}{RGB}{194,230,153}
    \definecolor{YlGn-G}{RGB}{120,198,121}
    \definecolor{YlGn-I}{RGB}{49,163,84}
    \definecolor{YlGn-K}{RGB}{0,104,55}
```

```
    \pgfplotsset{colormap={brewer}{color=(YlGn-B) color=(YlGn-
E) color=(YlGn-G) color=(YlGn-I) color=(YlGn-K)}}
\begin{document}
    \begin{tikzpicture}[baseline]
        \begin{axis}[anchor=center,enlargelimits=false,
            colorbar sampled={
                surf,
                samples=\pgfplotscolormapsizeof{\pgfkeysvalueof{/pgfplots/colormap name}}+
                domain=0:\pgfplotscolormapsizeof{\pgfkeysvalueof{/pgfplots/colormap name}}
            },
            colorbar style={
                point meta min=0,
                point meta max=\pgfplotscolormapsizeof{\pgfkeysvalueof{/pgfplots/colormap
                y tick label as interval,
            },
        ]
        \addplot[matrix plot,
            nodes near coords=\coordindex,mark=*,
            point meta=explicit,
            colormap access=direct,
        ]
        coordinates {
        (0,0) [0] (1,0) [1] (2,0) [2] (3,0) [NaN]

        (0,1) [3] (1,1) [4] (2,1) [5] (3,1) [99]

        (0,2) [3.5] (1,2) [-1] (2,2) [2.99] (3,2) [3]
        };
        \end{axis}
    \end{tikzpicture}
    %
    \def\showcolorandname#1{%
        \showcolor{#1}~\texttt{#1}%
    }%
    \def\showcolor#1{%
        \tikz \draw[black,fill={#1}] (0,0) rectangle (2em,1.6em);%
    }%
    %
    \begin{minipage}[c]{5cm}
        \showcolorandname{YlGn-B}\\
        \showcolorandname{YlGn-E}\\
        \showcolorandname{YlGn-G}\\
        \showcolorandname{YlGn-I}\\
        \showcolorandname{YlGn-K}
    \end{minipage}
\end{document}
```

or

```
\documentclass{standalone}
\usepackage{pgfplots}
    \pgfplotsset{compat=1.9}
    \definecolor{YlGn-B}{RGB}{255,255,204}
    \definecolor{YlGn-E}{RGB}{194,230,153}
    \definecolor{YlGn-G}{RGB}{120,198,121}
    \definecolor{YlGn-I}{RGB}{49,163,84}
    \definecolor{YlGn-K}{RGB}{0,104,55}
    \pgfplotsset{colormap={brewer}{color=(YlGn-B) color=(YlGn-
E) color=(YlGn-G) color=(YlGn-I) color=(YlGn-K)}}
\begin{document}
    \begin{tikzpicture}[baseline]
        \begin{axis}[anchor=center,enlargelimits=false,
            colorbar sampled={
                surf,
                samples=\pgfplotscolormapsizeof{\pgfkeysvalueof{/pgfplots/colormap name}}+
                domain=0:\pgfplotscolormapsizeof{\pgfkeysvalueof{/pgfplots/colormap name}}
            },
            colorbar style={
                point meta min=0,
                point meta max=\pgfplotscolormapsizeof{\pgfkeysvalueof{/pgfplots/colormap
                ytickmax=\pgfplotscolormaplastindexof{\pgfkeysvalueof{/pgfplots/colormap n
                %y tick label as interval,
            },
        ]
        \addplot[matrix plot,
            nodes near coords=\coordindex,mark=*,
            point meta=explicit,
            colormap access=direct,
        ]
        coordinates {
        (0,0) [0] (1,0) [1] (2,0) [2] (3,0) [NaN]

        (0,1) [3] (1,1) [4] (2,1) [5] (3,1) [99]

        (0,2) [3.5] (1,2) [-1] (2,2) [2.99] (3,2) [3]
        };
        \end{axis}
    \end{tikzpicture}
    %
    \def\showcolorandname#1{%
        \showcolor{#1}~\texttt{#1}%
    }%
    \def\showcolor#1{%
        \tikz \draw[black,fill={#1}] (0,0) rectangle (2em,1.6em);%
    }%
    %
    \begin{minipage}[c]{5cm}
        \showcolorandname{YlGn-B}\\
        \showcolorandname{YlGn-E}\\
```

```
        \showcolorandname{YlGn-G}\\
        \showcolorandname{YlGn-I}\\
        \showcolorandname{YlGn-K}
    \end{minipage}
\end{document}
```

TODOsp: I think one of them was an intermediate example for the already implemented feature `colormap access=direct` or `colorbar as legend`, right?

pgfplotstodo.tex:7699 **Feature Proposal** [open, Priority 5, Epic 'legend']

support `\multicolumn` for legends

pgfplotstodo.tex:7704 **Feature Proposal** [open, Priority 5, Epic 'legend']

it appears line breaks in legend descriptions are a problem (?) $\rightsquigarrow$ bug in pgf: \\ is overwritten and won't be restored.

pgfplotstodo.tex:7708 **Feature Proposal** [open, Priority 5, Epic 'pgfplotstable']

pgfplotstable file open protocol: provide public listener interface

pgfplotstodo.tex:7739 **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.

pgfplotstodo.tex:7743 **Feature Proposal** [open, Priority 5, Epic ‘’]

`\addplot coordinates {\macro};`

pgfplotstodo.tex:6373 **Feature Proposal** [open, Priority 4, Epic ‘’]

regression line computation: also generate macros containing the coefficient of determination

`http://en.wikipedia.org/wiki/Coefficient_of_determination`

`https://tex.stackexchange.com/questions/147249/pgfplots-linear-regression-mean-square-error`

pgfplotstodo.tex:7135 **Feature Proposal** [open, Priority 3, Epic ‘Performance’]

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

pgfplotstodo.tex:6364 **Feature Proposal** [open, Priority 2, Epic ‘’]

faceted allows to draw the entire rectangle.

It would allow cool effects if one could draw only those lines along the x direction (or y direction).

Compare `http://tex.stackexchange.com/questions/166768/draw-a-surface-from-scattered-curves`

pgfplotstodo.tex:5266 **Feature Proposal** [open, Priority 1, Epic ‘fillbetween’]

Optimization: if the same plot contains `name intersections` and `fill between` or `intersection segments`: compute the intersection points just once

pgfplotstodo.tex:7075 **Feature Proposal** [open, Priority 1, 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:7091 **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”

pgfplotstodo.tex:7750 **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.