

The NTS Project

EuroT_EX – Heidelberg DE – 1999

Jiří Zlatuška

Philip Taylor

Karel Skoupý

Joachim Lammarsch

Hans Hagen

History

Choices

Results

Future

Dante e.V.

Unknown Donator

Peter Breitenlohner



History

Choices

Results

Future

The History of NTS

Joachim Lammarsch

we saw ...

features of T_EX making it into other programs

that providing a decent user interface
to T_EX was nearly impossible

the T_EX community was not that good at proselytising

the number of users was diminishing
(or at least, not growing that much)

we saw ...

so ...

we had to improve $\text{T}_{\text{E}}\text{X}$ and come up with something better

which means that we had to make $\text{T}_{\text{E}}\text{X}$ more user friendly

and in the same time we had to make $\text{T}_{\text{E}}\text{X}$ more attractive

we saw ...

so ...

but ...

$\text{T}_{\text{E}}\text{X}$ was frozen by Donald Knuth, and was nearly bugfree, very stable and highly portable

we didn't want to lose that stability and portability

there were many opinions on how to continue

we saw ...

so ...

but ...

and then ...

the discussion around a New kind of
Typesetting System became more organized

nevertheless, commercial vendors were
not that interested in a successor of T_EX

and publishers were not that interested too

e.V. Dante decided to provide the
basic funding for real developments

NTS had become a fact instead of fiction

NTS: Programming Languages and Paradigms

Jiří Zlatuška


```
graph LR; A(Objectives) --> B(A new, more powerful successor to TeX-the-typesetter.); A --> C(An effort to re-program TeX-the-program.);
```

Objectives

A new, more powerful successor
to TeX-the-typesetter.

An effort to re-program TeX-the-program.

A procedural language based on a formal syntax and well-defined semantics.

Objectives

Structured programming as a methodology.

Pascal

Building large programs from manageable smaller components.

Abstract data structures instead of just the data types provided by architectures.

Portability to different platforms.

Introducing the world to literate programming.

Extending Pascal by pre-processing the source code.

Tangling a WEB and weaving the documentation.

Arithmetic calculations and memory management built in to guarantee portability.

Objectives

Pascal

Knuth's choices

Extending $\text{T}_{\text{E}}\text{X}$ is not that easy.

Successors of Pascal didn't make it.

Even Knuth has moved to C.

Hardware is no longer a constraint.

$\text{T}_{\text{E}}\text{X}$ data structures have become time bombs.

$\text{T}_{\text{E}}\text{X}$'s monolithic, interwoven character has become a burden.

We need to re-create $\text{T}_{\text{E}}\text{X}$ using modern methodologies and technologies.

Objectives

Pascal

Knuth's choices

20 years later

Removing all constraints and most complexity.

Adding modularity and clear interfaces.

Functional programming would suit
T_EX's basic manipulations: Lisp?

Logic programming would help us specify
and solve typographic problems: Prolog?

Procedural programming along the
lines T_EX is coded now: C and C++?

The portable compromise, procedural
programming while using objects: Java!

Objectives

Pascal

Knuth's
choices

20 years later

Options

Provide a high level of structure
using objects and methods.

Anchor NTS in the World Wide Web.

Support as many platforms as possible.

Use standardized interfaces.

Move from WEB to WWWeb.

Objectives

Pascal

Knuth's
choices

20 years later

Options

NTS in Java

The Implementation of NTS

Karel Skoupý & Philip Taylor

NTS: The Structure

the implementation language of NTS is Java

the program code is encapsulated in classes

objects are instances of (sub)classes

classes are clustered in packages

The initialization

```
\def \initialisation
  {\nonstopmode

  \input init.inc

  \tracingcommands = 0 \tracingonline = 0 \tracingparagraphs = 0

  \time = 750 \showboxdepth = 100 \showboxbreadth = 1000000

  \baselineskip = 12pt \lineskiplimit = 0pt \lineskip = 1pt

  \def \NTS {{\tenit NTS}}

  \font \tenrm = cmr10 \font \tenit = cmti10 \tenrm
}
```

A normal paragraph

```
\def \normalpar  
  {\parindent = 0 pt  
  %\adjdemerits = 10  
  \pretolerance = 300  
  \tolerance = 300  
}
```

A narrow paragraph

```
\def \narrowpar  
  {\hspace = 0,5\hspace  
   \tolerance = 9999  
   \leftskip = 0.2\hspace  
  }
```

A centered paragraph

```
\def \centeredpar
  {\leftskip = 0.5\hsize plus 1 fil
  \rightskip = \leftskip
  \parindent = 0 em
  \parfillskip = \parindent
  \hsize = 2\hsize
}
```

A few rules

```
\def \divider #1%
  {\ifcase #1
    \message {No zeroth divider class}
  \or
    \par \hrule \par
  \or
    \par
    \vskip 10pt
    \hrule height 1 pt depth 1 pt
    \vskip 10pt
    \par
  \else
    \message {No divider class > 2}
  \fi
}
```

The text

```
\def \text
```

```
{Other authors in this series of papers on \NTS\ have explained the reasons for the creation of the \NTS~project (Joachim Lammarsch), the rationale behind the choice of programming language (Ji\v{r}\'\{i} Zlatu\v{s}ka), and future directions in which the project may develop (Hans Hagen). This paper addresses the rather more detailed area of the actual implementation itself, and is intended to provide the reader with as much detail as can reasonably be accommodated in a paper which is intended to appear in the Conference Proceedings. A considerably more detailed version of the paper will eventually be available as an accompaniment to (or possibly integrated in) the JavaDoc documentation which will accompany the released version of \NTS.
```

```
}
```

The typesetting

```
\setbox 1 = \hbox {\text}
```

```
\shipout \vbox
```

```
{\normalpar
```

```
\unhcopy 1
```

```
\divider 1
```

```
\narrowpar
```

```
\unhcopy 1
```

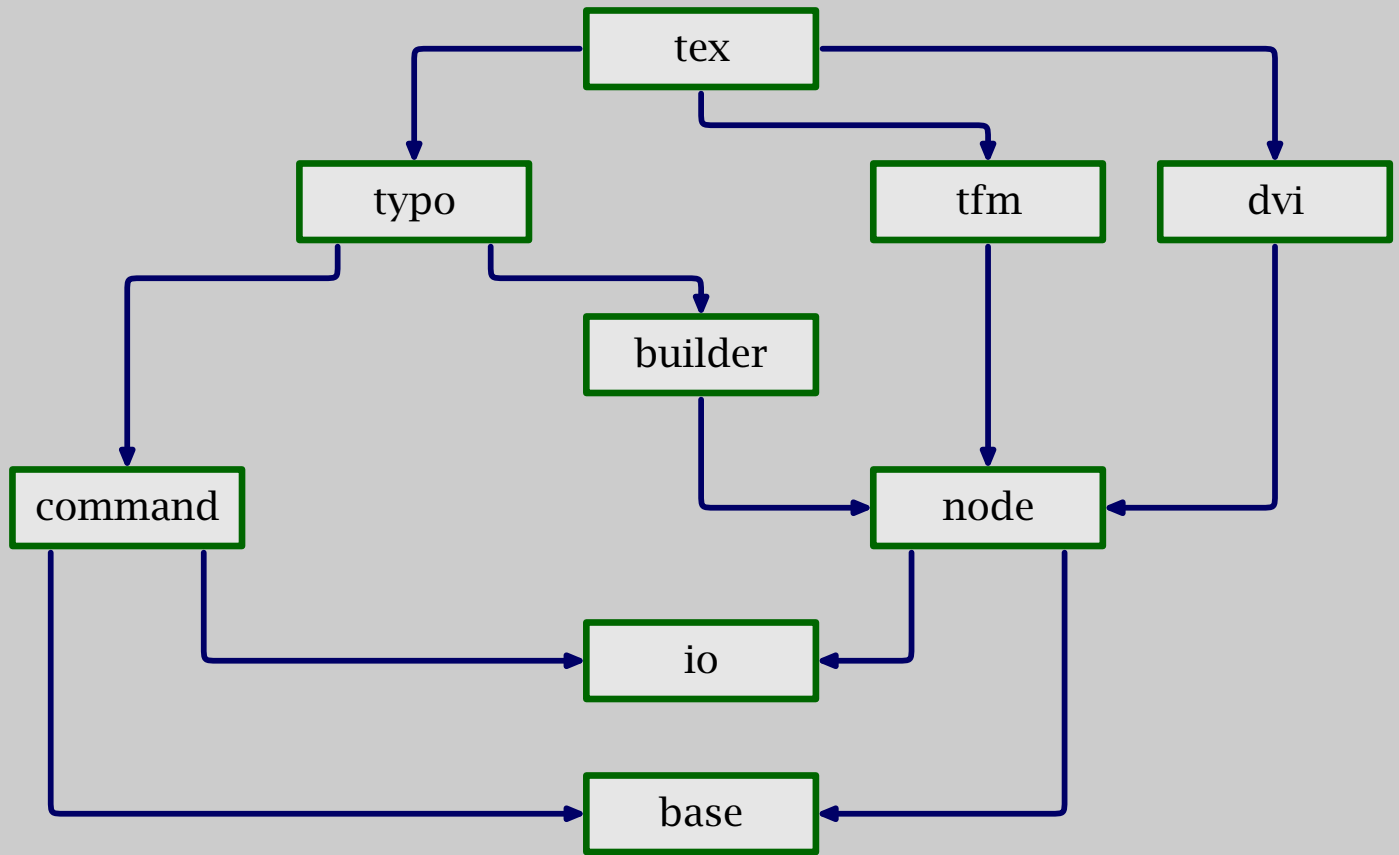
```
\divider 2
```

```
\centeredpar
```

```
\unhcopy 1
```

```
}
```

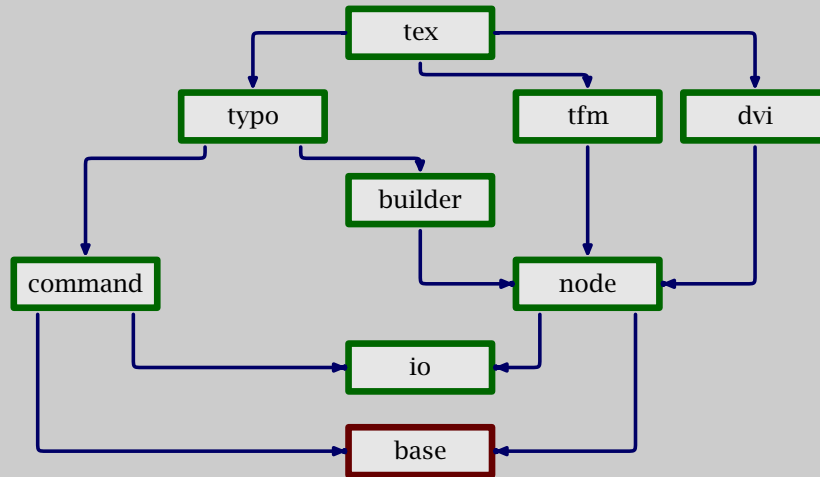
```
\end
```



NTS Java packages

base

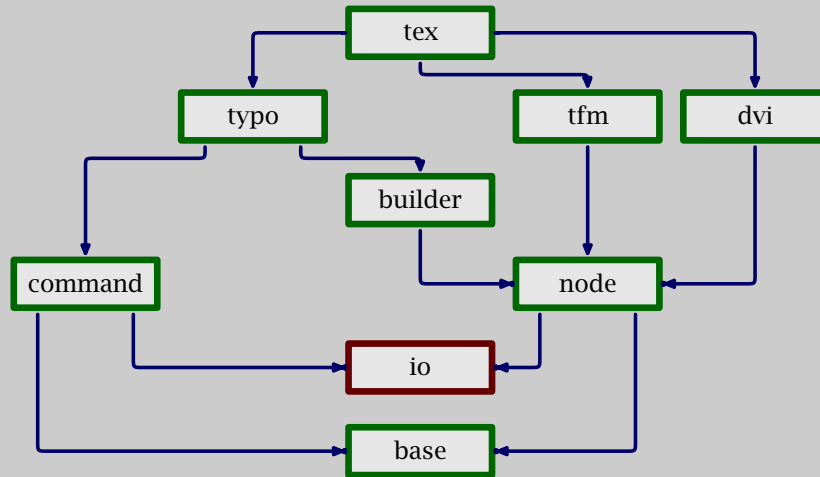
implements elementary data types



Dimen Glue Num LevelEqTable

io

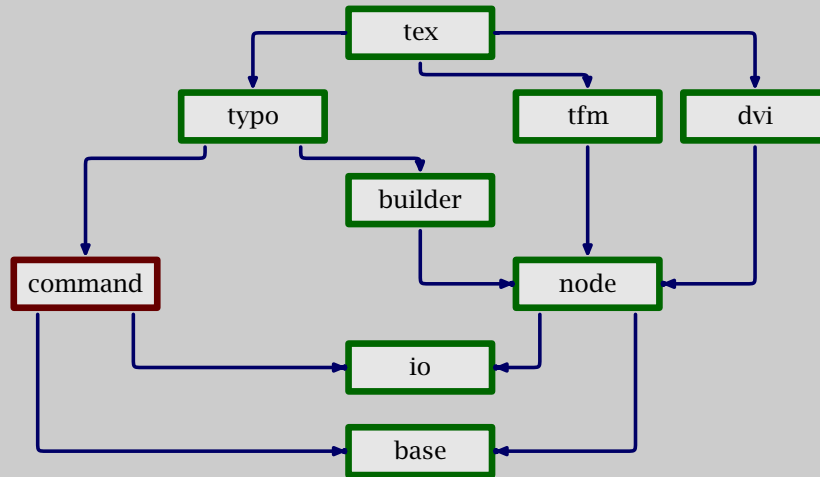
handles reading from input and writing to the log file



CharCode Name InputLine Log Loggable

command

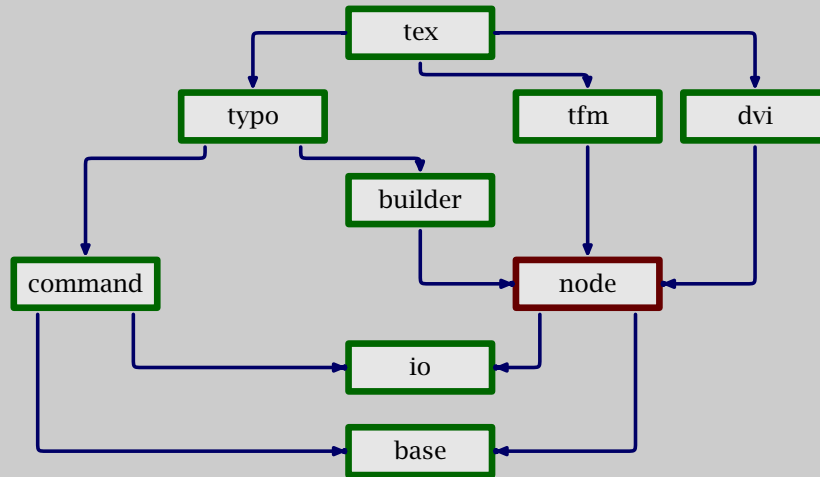
interprets the $\text{T}_\text{E}\text{X}$ input language



Token Tokenizer Command CommandBase

node

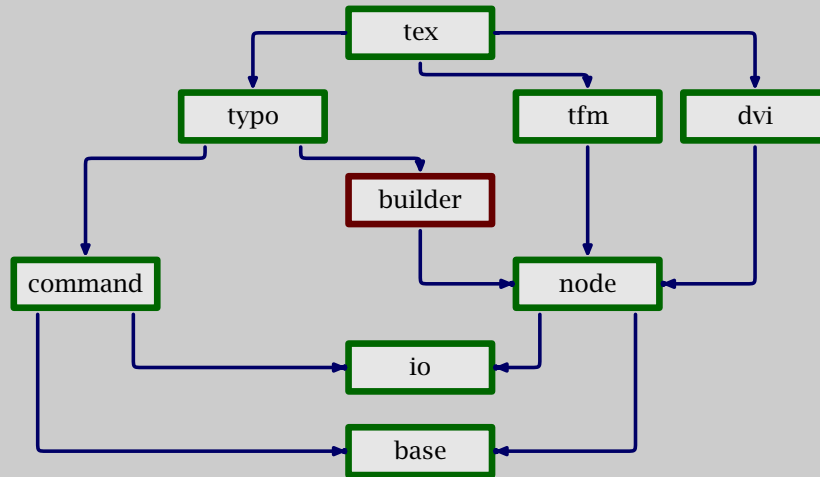
represents the material to be typeset



Node Packer FontMetric TypeSetter

builder

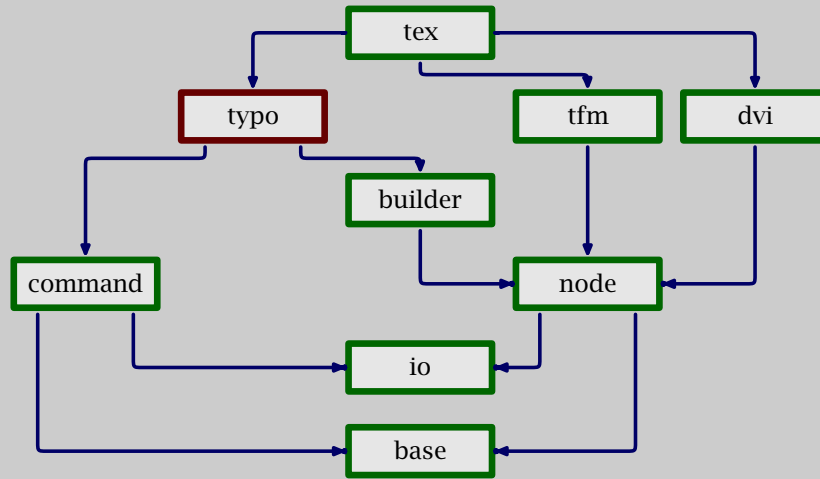
takes care of mode-related things



Builder

typo

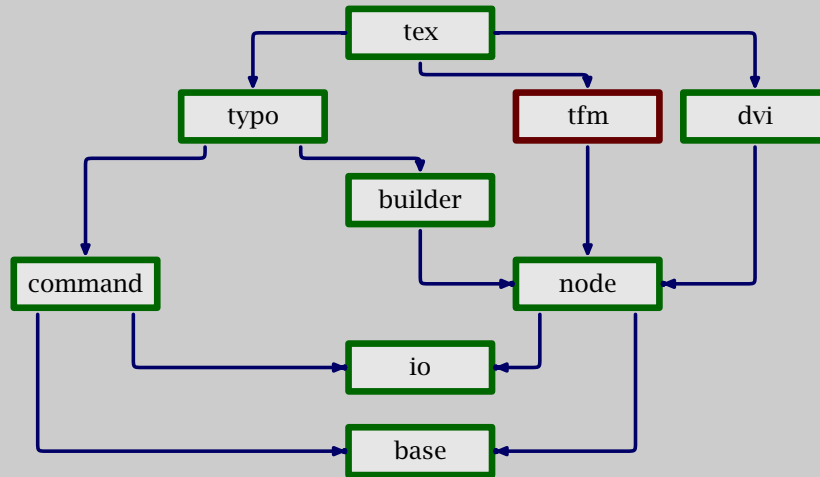
deals with typesetting



TypoCommand BuilderCommand Group

tfm

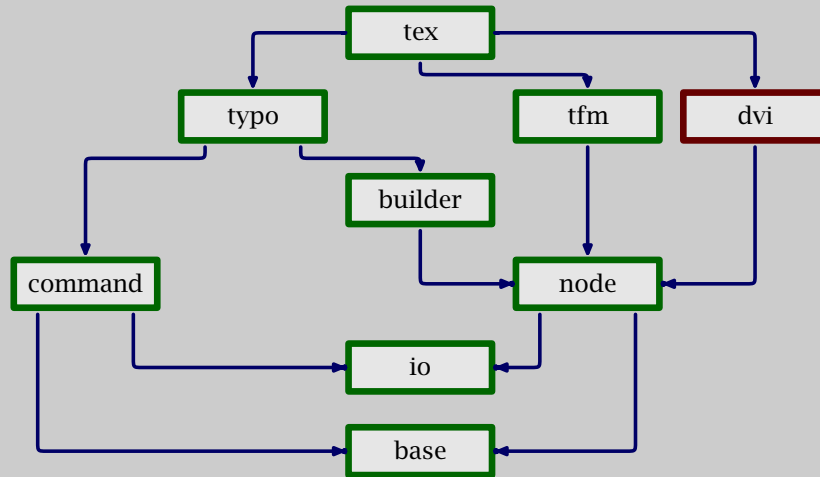
handles the natural $\text{T}_{\text{E}}\text{X}$ font metrics



TeXFm TeXFontMetric

dvi

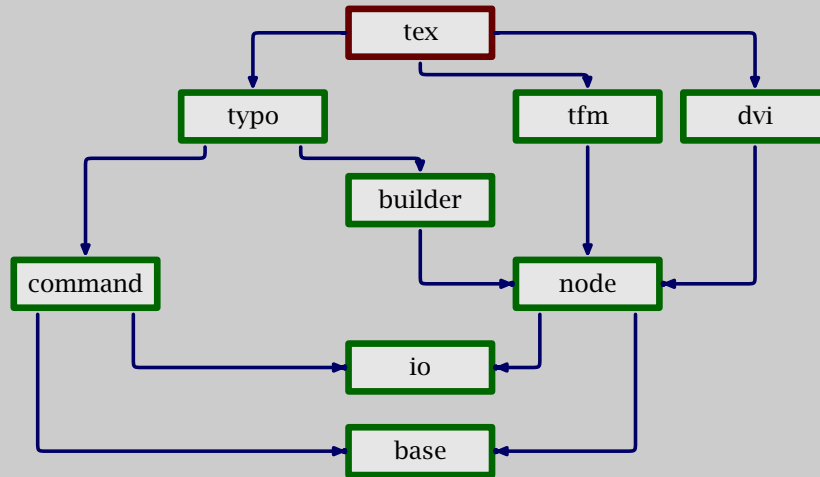
takes care of $\text{T}_{\text{E}}\text{X}$'s native output format



DviFormatSetter DviTypeSetter

tex

glues everything into a program



A few concluding remarks

compatibility with T_EX is important to gain user confidence and widespread acceptance

quite some time has been invested in obtaining strict compatibility with T_EX

in spite of extensive documentation of T_EX-the-program, much in-depth study of the code was needed

Java still has kept its promise, but the heavy use of objects will have a performance penalty

The Future of NTS

Hans Hagen

The predecessor
 ϵ - $\text{T}_{\text{E}}\text{X}$ provides:

extensions in the
spirit of Knuth

limitations in
registers removed

some low level
optimizations

protection
against expansion

more conditionals
and feedback

control over
expansion

bi-directional
typesetting


The stimulator
pdf_TE_X has
brought:

a new high
quality backend

removing
the post-
processing stage

additional
paragraph
optimization

some tantilizing
things to
come ...

Three blue circles are arranged horizontally across the page. Each circle has a thick blue border and a light blue fill. The text inside each circle is in a black serif font. The first circle on the left contains the text 'The challenger eeTeX provided a playground for:'. The middle circle contains 'extensive list manipulations'. The third circle on the right contains 'an alternative input parser'.

The challenger
eeTeX provided a
playground for:

extensive list
manipulations

an alternative
input parser

Together ϵ -TeX,
pdfTeX, eeTeX
and others:

have smoothed
the path to NTS

have opened the
road to a more
drastic deviation

have shown that
we cannot neglect
alternative

have
demonstrated
that TeX can be
extended in a

made clear that
many people
depend on
macro writers

have learned
that change
should be guided
and guarded

After TUG2000:

NTS will provide a tested and robust starting point

the team will have set up an infrastructure to handle changes

new functionality and interfaces can be prototyped

we need to discuss standards that will guarantee stable environment

all old wishes will be looked into

macro packages will become NTS aware

something like this ...

Columns and Grids
Overall Appearance
Embedded Graphics
Layers in Text

Do we need NTS?

First of all we want our columns
to be perfectly balanced.
This is trivial for pure text,
but imagine lots of white
space, like display math.

Columns and Grids

We want floats to be moved to the best available location. Of course we want floats to span more than one column, and even spanning one and a halve column with a text flowing around the figure should be possible.

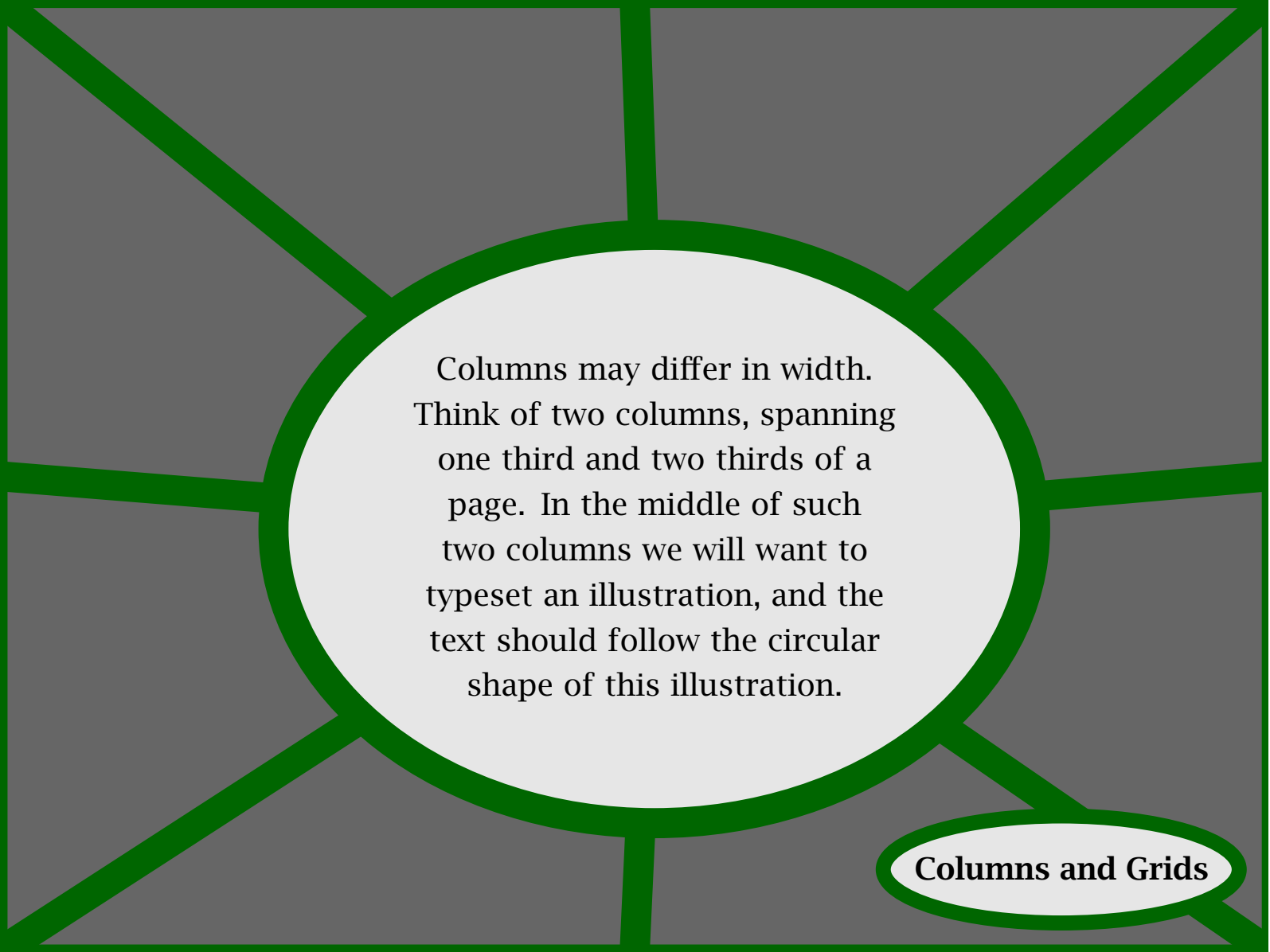
Columns and Grids

In double sided output, we want lines to align on the opposing pages (spread). When we hold the paper towards a bright source of light, we want the lines to align too.

Columns and Grids

We definitely don't want to end up with a few lines or words on the last page. Why not apply a small percentage of glyph scaling in such a way that we get full pages? Of course we will need more than paragraph and page optimization for this: we are dealing with the document as a whole.

Columns and Grids



Columns may differ in width. Think of two columns, spanning one third and two thirds of a page. In the middle of such two columns we will want to typeset an illustration, and the text should follow the circular shape of this illustration.

Columns and Grids

Talking of illustrations,
instead of being something
with fixed dimensions, the
scale may be adapted, of
course consistently, to suit the
overall document appearance
(grid, spread, and more).

Columns and Grids

Are you still thinking from left to right? Text can go in all directions, and will be mixed too. The width of columns may change in the meantime. Anyone who has seen traditional jewish religion documents, will see the challenge in nested columns with (foot)notes flowing around partial columns.

Columns and Grids

A case study of columns.

The diagram is a mind map with a central white oval containing the text "A case study of columns." in green. Eight green lines radiate from this central oval to the corners of the page. A smaller white oval is attached to the bottom-right branch, containing the text "Columns and Grids" in black.

Columns and Grids

Typesetting is more than
manipulating metrics. Don't
we need a typesetting system
that looks at the glyphs
themselves, the small graphics?

Overall Appearance

People tend to disagree on what looks best, but experts often agree on what looks worse. Why not build in expert knowledge, or even better, build a system that learns from the user's rating?

Overall Appearance

How is greyness calculated? Does NTS act upon the internal lists of glyphs, or does it first build a bitmap? At least then it knows how the pages comes out. Is the validation a function of an output device? Will the shape of glyphs depends on the rating? Will $\text{T}_{\text{E}}\text{X}$ and METAFONT become one?

Overall Appearance

Is, in validating the appearance,
a model of the page needed, in
terms of meaningful areas? If so,
how is such a model defined? Do
we need pattern recognition?

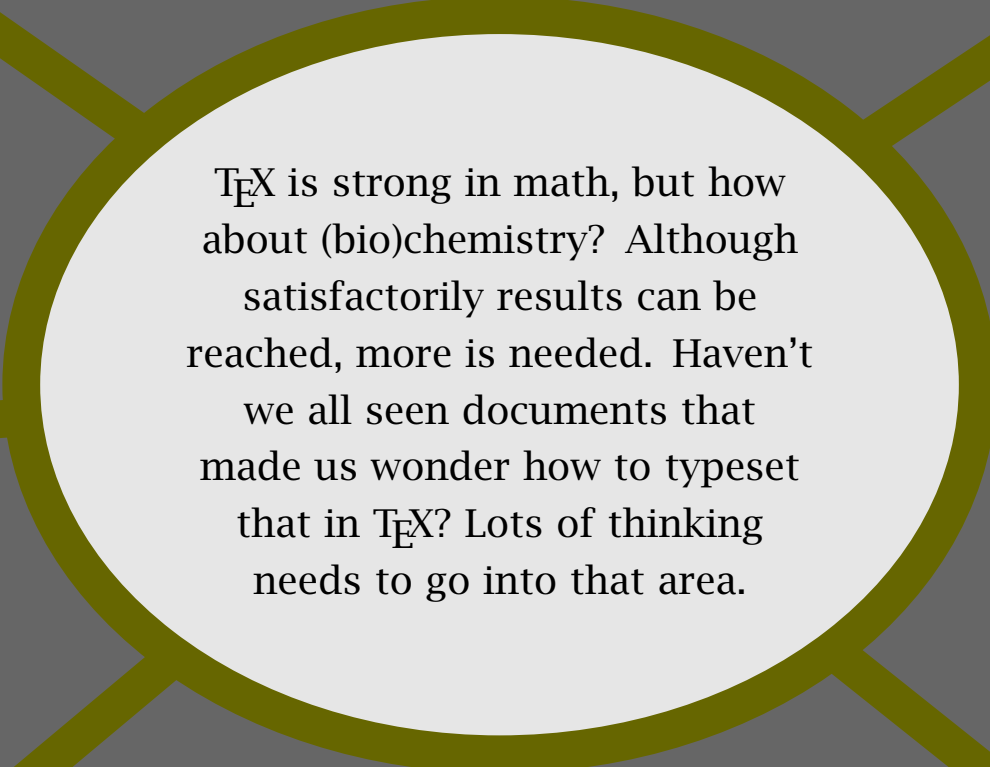
Overall Appearance

NTS needs a graphics engine, or maybe even several. Models for exchange of information between processes dealing with pure typesetting and drawing shapes need to be developed. Such mechanisms should cooperate naturally with the paragraph and page breaking as well.

Embedded Graphics

Typesetting along curves,
turning shapes into outlines,
and applying arbitrary filling
and shading, it all makes sense.

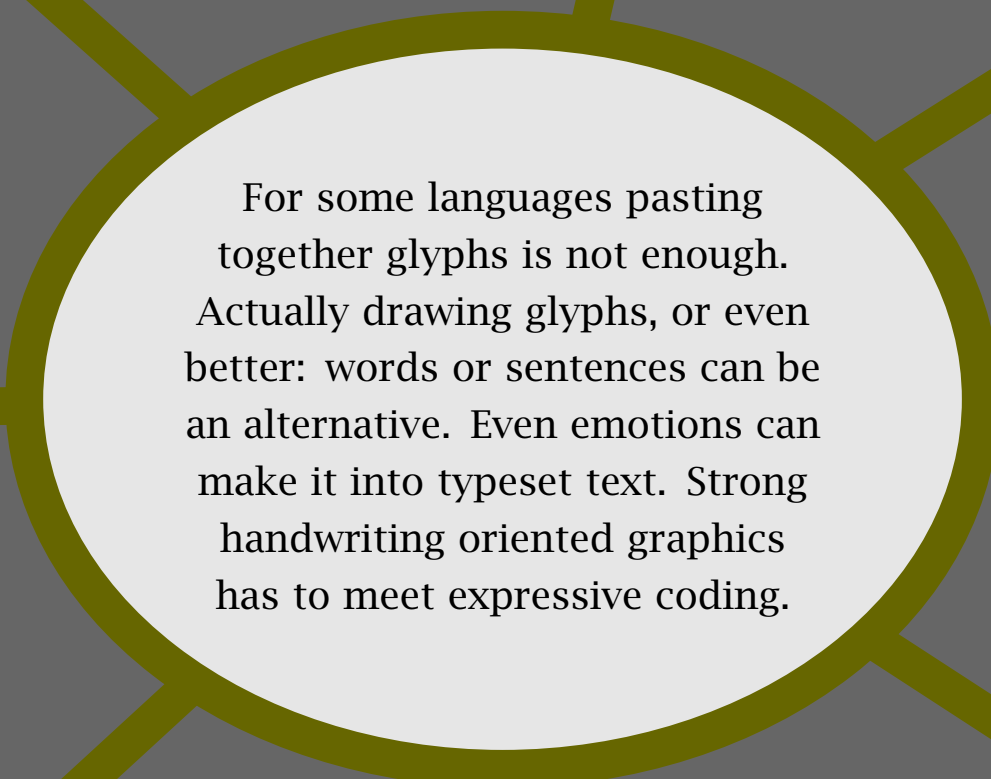
Embedded Graphics



$\text{T}_{\text{E}}\text{X}$ is strong in math, but how about (bio)chemistry? Although satisfactorily results can be reached, more is needed. Haven't we all seen documents that made us wonder how to typeset that in $\text{T}_{\text{E}}\text{X}$? Lots of thinking needs to go into that area.



Embedded Graphics



For some languages pasting together glyphs is not enough. Actually drawing glyphs, or even better: words or sentences can be an alternative. Even emotions can make it into typeset text. Strong handwriting oriented graphics has to meet expressive coding.



Embedded Graphics

**Some examples of
embedding graphics.**

Embedded Graphics

First of all, the new system
needs some more understanding
about the typeset text.
Support for UNICODE, unified
glyph names is mandate.

Layers in Text

When searching through a document, some knowledge on what language the text we're dealing with makes sense. Not only the (many) language(s) of a text, but the direction also plays a role. Complicated ligatures should be recognized properly.

Layers in Text

In more dynamic documents, like fill-in-forms, interaction with a typesetting engine is not a luxury, especially not in european and eastern languages. NTS can be such a plug in, but the document itself should contain the information needed to let NTS to do its task. A document is more than a collection of graphics and glyphs, and typesetting more than organizing those.

Layers in Text

As PDF \TeX already demonstrates,
using \TeX to embed typeset
information like pop-up-help and
tool tips is a breeze. Although
heavily dependent of features of
viewers, NTS will benefit from a
decent model of layers on which
we typeset as well as concepts
of information hidden in the
output but showing up at wish.

Layers in Text

líbilo.²⁷Všechno mi bylo předáno od mého Otce; a nikdo plně nepoznává²⁸ Syna, jen Otec, ani Otec nikdo plně nepoznává²⁹, jen Syn a ten, komu by ho Syn chtěl zjevit.²⁸Pojďte ke mně všichni, kteří těžce³⁰ pracujete³¹ a jste³² přetíženi, a já vám dám³³ odpočinek.³⁴Vezměte mé jho na sebe a uče³⁵ se ode mne, neboť jsem tichý a pokorný v srdci; a naleznete³⁶ odpočinutí svým duším.³⁶Vždyť³⁷ mé jho je příjemné³⁸ a mé břemeno je lehké.³⁹

12⁴⁰V ten čas šel Ježíš v sobotu obilím. Jeho učedníci dostali⁴¹ hlad⁴² a začali trhat klasy a jíst.⁴³Když⁴⁴ [je] uviděli⁴⁵ farizeové, řekli mu: „Hle, tvoji učedníci dělají, co se nesmí dělat v sobotu.“⁴⁶On však jim řekl: „Nečetli jste, co udělal David, když vyhladověl, on i ti, kdo byli s ním?⁴⁷ Jak vešel do Božího⁴⁸ domu a snědl⁴⁹ [chleby předložení⁵⁰], které nesměl jíst *ami* on ani ti,

1 E: naučte se; 2 nebo: dobré; 3 var.: snědl; 4 E: *aristis prothesosis*, h. *techem happátim*, dosl. „chleby sváté“; Tyto chleby byly pravidelně předkládány před B. tvář za 12 pokolení Izraele; viz Ex 25,30; Lv 24,5–9; 5 E: uschlou; 6 var.: uzdravit; 7 E: Kdo bude z vás člověk, který bude ml...; 8 nebo: se to dověděl / to poznal; 9 var.: mnozí; 10 E: aby ho neucínil známým;

kdo byli s ním, ale jen kněží?

⁵¹Anebo nečetli jste v Zákoně, že o sobotách kněží v chrámě porušují sobotu, a přece jsou bez viny?

⁵²Právím vám, že zde je někdo větší než chrám.⁵³Kdybyste věděli⁵⁴, co znamená: „Milosrdenství chci, a ne obět“^{55,56}, neodsoudili byste nevinné.⁵⁷Vždyť⁵⁸ Syn člověka je páнем soboty.⁵⁹⁶⁰A když⁶¹ odtamtud odešel⁶², přišel do jejich synagogy.⁶³A hle, byl⁶⁴ tam člověk, který měl odumřelou⁶⁵ ruku. I otázali se ho: „Je dovoleno v sobotu uzdravovat?“⁶⁶ *To proto*, aby jej obžalovali.⁶⁷On jim řekl: „Kdyby měl někdo z vás⁶⁸ jednu ovci a ta by *mu* v sobotu spadla do jámy, což by ji neuchopil⁶⁹ a nevytáhl⁷⁰?

⁷¹Oč je člověk cennější než ovce! Proto je dovoleno v sobotu činit dobře.“⁷²Potom řekl⁷³ tomu člověku: „Natáhni svou ruku.“ Natáhl jí, a byla zase⁷⁴ v pořádku⁷⁵ a zdravá jako ta druhá.⁷⁶Farizeové však vyšli a radili se proti němu,

Facsimile Snapping Breaking Notes

Facsimile Snapping Breaking Notes

23

Matouš 12

líbilo. ²⁷Všechno mi bylo předáno od mého Otce; a nikdo plně^s nepoznává^s Syna, jen Otec, ani Otec nikdo plně^s nepoznává^s, jen Syn a ten, komu by *ho* Syn chtěl zjevit. ²⁸Pojďte ke mně všichni, kteří těžce^s pracujete^s a jste^p přetíženi, a já vám dám^{sf} odpočinek^s. ²⁹Vezměte mé jho na sebe a učte¹ se ode mne, neboť jsem tichý a pokorný v srdci; a naleznete^f odpočinutí svým duším. ³⁰Vždyť¹ mé jho je příjemné² a mé břemeno je lehké.“

12 ¹V ten čas šel Ježíš v so-

kdo byli s ním, ale jen kněží?

⁵Anebo nečetli jste v Zákoně, že o sobotách kněží v chrámě porušují sobotu, a *přece* jsou bez viny?

⁶Pravím vám, že zde je *někdo* větší než chrám. ⁷Kdybyste věděli^p,

co znamená: „Milosrdenství chci, a ne oběť“^{oz 6,6}, neodsoudili byste nevinné. ⁸Vždyť¹ Syn člověka je pánem soboty.“

⁹A když^s odtamtud odešel^s, přišel do jejich synagogy. ¹⁰A hle, *byl tam* člověk,

který měl odumřelou⁵ ruku. I otázali se ho: „Je dovoleno v sobotu uzdravovat⁶?“

To proto, aby jej

uzdravovat⁶?“ *To proto*, aby jej

2

bu mmeno Otec, a nikdo plně nepoznává^s Syna, jen Otec, ani Otec nikdo plně^s nepoznává^s, jen Syn a ten, komu by *ho* Syn chtěl zjevit. ²⁸Pojďte ke mně všichni, kteří těžce^s pracujete^s a jste^p přetíženi, a já vám dám^{sf} odpočinek^s. ²⁹Vezměte mé jho na sebe a učte¹ se ode mne, neboť jsem tichý a pokorný v srdci; a naleznete^f odpočinutí svým duším. ³⁰Vždyť mé jho je příjemné² a mé břemeno je lehké.“

12 ¹V ten čas šel Ježíš v sobotu obilím. Jeho učedníci dostali^s hlad^s a začali trhat klasy a jíst. ²Když^s [*je*] uviděli^s farizeové, řekli mu: „Hle, tvoji učedníci dělají, co se nesmí dělat v sobotu.“ ³On však jim řekl: „Nechetli jste, co udělal David, když vyhladověl, *on* i ti, *kdo byli* s ním? ⁴Jak vešel do Božího^c domu a snědl³ (chleby předložený⁴), které nesměl jíst *ani* on ani ti,

nebo nechtěl^s v Zakone, že v sobotách kněží v chrámě porušují sobotu, a přece jsou bez viny? ⁶Pravím vám, že zde je někdo větší než chrám. ⁷Kdybyste věděli^p, co znamená: ‚Milosrdenství chci, a ne obět‘^{xOz 6,6}, neodsoudili byste nevinné. ⁸Vždyť Syn člověka je pánem soboty.“ ⁹A když odtamtud odešel^s, přišel do jejich synagogy. ¹⁰A hle, *byl tam* člověk, který měl odumřelou⁵ ruku. I otázali se ho: „Je dovoleno v sobotu uzdravovat⁶?“ *To proto*, aby jej obžalovali. ¹¹On jim řekl: „(Kdyby měl^f někdo z vás)⁷ jednu ovci a ta by *mu* v sobotu spadla do jámy, což by ji neuchopil^f a nevytáhl^f? ¹²Oč je člověk cennější než ovce! Proto je dovoleno v sobotu činit dobře.“ ¹³Potom řekl^h tomu člověku: „Natáhni svou ruku.“ Natáhl *ji*, a byla^s zase^s vs pořádku^s a zdravá jako ta druhá. ¹⁴Farizeové však vyšli a radili se proti němu,

1 ř.: naučte se; 2 nebo: dobré; 3 var.: snědl; 4 ř. *artoi tés protheseós*, h. *lechem happáním*, dosl. ‚chléb tváře‘. Tyto chleby byly pravidelně předkládány před B. tvář za 12 pokolení Izraele;

Facsimile Snapping Breaking Notes



Iž botu obilím. Jeho učedníci dostali^s hlad^s a začali trhat klasy a jíst. ²Když^s [je] uviděli^s farizeové, řekli mu: „Hle, tvoji učedníci dělají, co se nesmí dělat v sobotu.“ ³On však jim řekl: „Nechťli jste, co udělal David, když vyhladověl, *on* i ti, *kdo byli s ním*? ⁴Jak vešel do Božího^c domu a snědli³ (chleby předloženi)⁴, které nesměl jíst *ani* on ani ti,

obžalovali. ¹¹On jim řekl: „Kdyby měl někdo z vás⁷ jednu ovci a ta by *mu* v sobotu spadla do jámy, což by ji neuchopil^f a nevytáhl^f? ¹²Oč je člověk cennější než ovce! Proto je dovoleno v sobotu činit dobře.“ ¹³Potom řekl^h tomu člověku: „Natáhni svou ruku.“ Natáhl *ji*, a byla^s zase^s v^s pořádku^s a zdravá jako ta druhá. ¹⁴Farizeové však vyšli a radili se proti němu,

1 ř.: naučte se; 2 nebo: dobré; 3 var.: snědl; 4 ř. *artoi tés protheseós*, h. *lechem happáním*, dosl. „chléb tváře“. Tyto chleby byly pravidelně předkládány před B. tvář za 12 pokolení Izraele; viz Ex 25,30; Lv 24,5–9. 5 ř.: uschlou; 6 var.: uzdravit; 7 ř.: Kdo bude z vás člověk, který bude mít... 8 ► nebo: se to dověděl / to poznal; 9 ► var.: mnozí; 10 ► ř.: aby ho neučinili známým;

Facsimile Snapping Breaking Notes

Zaagh, of Steek-zaagh	Domme-kraght	Mokers
Yzere Wiggen	Brandt-yzer	Hout-haak
Spaander-haak	Een Klaas Jacobzen	Tange
Nagel-hamer	Wigge	Oor-houten
Kluften	Hellen	Steven-haken
Hevels en Klein-touwen	Een Koe-voet	Schot-bouten
Yzere ram	Avegaar	Een hout Ram
Een groote Wigge	Teer-ketel	Slyp-steen
Een Mal	Rye	Hout-bok
Schraagh	Vlotten	Een Slee
Dwars-slee	Kaap-stander	Bytels
Klavaats-hamer	Rabat-yzer	Klavaats-yzer
Spyker-yzer	Werk-bytel	Duim-stok
Schraper	Een Moker	Spyker-hamertje
Een Roffel	Gerf-schaaf	Odief
Ploegen	Handt-zaagh	Klamp-spykers-boor
tien duims Boor	Dissel	Byl

This list is derived from 'De Materie', a musical composition of Louis Andriessen. Long ago, the instruments listed here were used in ship-building.

We thrive in information–thick worlds because of our marvelous and everyday capacity to select, edit, single out, structure, highlight, group, pair, merge, harmonize, synthesize, focus, organize, condense, reduce, boil down, choose, categorize, catalog, classify, list, abstract, scan, look into, idealize, isolate, discriminate, distinguish, screen, pigeonhole, pick over, sort, integrate, blend, inspect, filter, lump, skip, smooth, chunk, average, approximate, cluster, aggregate, outline, summarize, itemize, review, dip into, flip through, browse, glance into, leaf through, skim, refine, engulf, glean, synopsisize, winnow the wheat from the chaff and separate the sheaves.

Intergrating text and graphics graphics in a \TeX – \MetaPost environment not only is thrilling, but also introduced new concepts. But, looking at this list as composed by E. Tufte, humans are capable to deal with those.

Thus, I came to the conclusion that the designer of a new system must not only be the implementer and first large-scale user; the designer should also write the first user manual. The separation of any of these four components would have hurt TeX significantly. If I had not participated fully in all these activities, literally hundreds of improvements would never have been made, because I would never have thought of them or perceived why they were important. But a system cannot be successful if it is too strongly influenced by a single person. Once the initial design is complete and fairly robust, the real test begins as people with many different viewpoints undertake their own experiments.

Wasn't it Donald Knuth who has said this? But what system is he talking about?