

Agenda

- Célok: különböző módszerek előadás slidok írásra
- Történelem: miért vált ez most nekem aktuálissá
- Feladat: Getting Started with Nics
- Nics demó és feature összefoglaló
- Nics ötletek, kód walkthrough
- De közben mindig figyelve az érdekes T_EX tudás megbeszélésére
- Bug fixing in nics, live coding

Népszerű módszerek

WYSIWYG

- Microsoft PowerPoint
 - nagyon gyorsan lehet haladni
 - szép a kimenet
 - de nem időtálló a kimenet (a PDF export igen)
 - problémás greppelni és copy-pastelni
 - problémás és munkás kódot színezni
 - problémás a branching és a teamwork
 - személyes preferencia: hate everything WYSIWYG
- Google Slides
 - all the same: még a teamwork sem jobb sokkal
 - személyes preferencia: love Google soooooo much

Népszerű módszerek

Markdown, AsciiDoc, RST, Emacs Org mód

- Markdown + Pandoc
 - van forrás kód, amit gitbe lehet tolni
 - nagyon népszerű, meetupokon ezt használják
 - a kód színezés is valamennyire meg van oldva
 - nagyon-nagyon WYGIWYG, nem flexibilis at all
 - pl. nem lehet cím slidera extra kontentet írni
- Nilcons status quo: Emacs Org mód + org-reveal(js)
 - valamennyire flexibilis
 - diagrammok még mindig problémásak \Rightarrow inkább nem csinállok
 - HTML-t generál ami kicsit flaky: resolution és böngésző függő
 - build stability: dockerből futtatjuk a batch emacsot

Aktualitás

Miért lett ez most érdekes

- Kubernetes + Cég: viszonylag sok óra
- Nehéz téma, sok buktatóval, amit sokszor kell előadnom
- Szeretném mindig kb. ugyanazt a minimumot mondani
- Adtak slidokat, amikből két verzió van
 - Régi motorosok: Google Slidesba konvertált PowerPoint
 - Full of bugs that nobody updates
 - Persze én elkezdhetem javítani, de history, merging?
 - Új ifjú titánok: Markdown PDF és HTML outputtal (Google Slidesből)
 - Visual bugs: slide kilóg, scrollbar screenshot in PDF (demó)
 - PowerPoint konverzió: szó-szó, amit nehéz volt, azt inkább nem
 - Diagrammok screenshotokká váltak: no updates/fixes possible

Aktualitás

Véleményváltozás: egy kép többet mond 100 szónál

- A PowerPoint slidokat annak idején még meg is szépíttette a Cég
- Meggyőzőtt: diagrammok egy CS előadásban jók, szöveg nem elég
- Mindig lehet táblára rajzolni, de az időhúzás, amit nem szeretek
- A slidokat szeretem jóra csinálni: nincs jegyzetelés
- Oké, de mi a baj a \LaTeX Beamer-rel?
 - Nagyon lassú fordítás, 100 slide kb. 0.5-1 perc
 - Nagyon sok mindenhez kell StackOverflowzni
 - Szép kinézethez csomagok tömkelegét kell összeházasítani
 - Random spacing issues, random megjósolhatatlan működés
 - Ha valami nekem random, akkor másnak impossible
 - Everyone will hate me, including future myself

Getting started

Download, compile, view

- <https://nics.nilcons.com/>
- <https://github.com/nilcons/nics>
- Debian buster: lucky you, go to docker branch and do the apt-get
- Linux: just make sure Docker hello-world works
- osx: Docker Desktop for Mac
- windows 10: WSL + debian buster + apt-get install

Features

Ötletek, requirements, implementation tricks

- Demó: feature demonstration pdf
- Demó: source kódja, nagyon struktúrált, explicit
- Adjuk a build rendszert
 - lehet caching, meg exporting
 - lehet bonyolult kód színezés
 - lehet gyors (T_EX format)
- Memory is cheap: mindent kirenderelünk boxokba, no triple-compile
- Slidok közti automatikus tördelés nem kell: shipout
- Mi kontrollálunk mindent: absolute positioning easy
- Absolute positioning is all you need: disable margins all
- Full screen pictures are easy after this

Features

Build system

- Assumption: you use git
- Kell egy submodulelet betenned a nicshez
- Így könnyen tudsz upgradelni (és ha valamit elcsesztem, rollbackelni)
- De az előadásaid privátok maradhatnak
- Your Makefile: specify nicsroot + include my nics makefile
- Reproducibility: Docker (és a T_EXmagától is elég jó)
- Confidence: MD5 a PPM konverzióról laponként (TODO: oldalszám nélkül)
- A PDF-et bekommitolni controversial, esetleg Git LFS?

Implementation

LuaTeX is awesome

- A TeX-nek nincs programozási nyelve
- Csak egy makrónyelve, ami versenyzik az m4-gyel jószágban
- LuaTeX: olyan TeX engine, amiben lehet luában is kódolni
 - és úgy egyébként is modern: unicode, PDF, etex
 - elég jó dokumentáció, de érteni kell az alapokat: TeXBook
 - az átjárás és call convention lehetne jobb
 - aktív fejlesztés alatt, de már most nagyon jó
- LuaTeX-ben implementált részek
 - verbatim, ami különben nehéz: it's hard to stop
 - exporting, sub-compilation, caching
 - PDF zooming (adjustbox)

PDF copy-paste, accessibility

Not very good \LaTeX support, anybody knows the PDF primitives?

- Copy-paste is problematic in PDF: no concept of “line”
- Glyphs are just put at the right places
- The viewer “figures out” what is a line for you to copy-paste
- This doesn’t work well with indented source code and double spaces
- Interestingly, xpdf is your best bet for copy-pasting indented code
- Accessibility (screen readers) is also problematic

PDF tools

- Prezikhez használhatok pdfpc-t
 - Két videó kimenetet használ
 - Hallgatók: teljes slide
 - Előadó: slide + következő slide + notes
 - Lehet “lézer pointerrel” mutogatni
 - Lehet rajzolni
 - Lehet freezelni és úgy lapozgatni
- Preview: evince vagy mupdf
- Visual PDF diff: `diffpdf -a`

T_EX run

Basic makró debugging

- `luatex '\relax\scrollmode'`
- * is the T_EX prompt
- Try, explain: `\show\centerline`, then `\show\line`
- Try, explain: `\show\lineskip`, then `\showthe\lineskip`
- Try, explain: `\message{\the\lineskip}`

T_EX hello world

While trying to be minimal

- Kód (hello.tex):

```
Hello World  
\bye
```

- Fordítás: `luatex hello.tex`
- Makródefiníció: `\def\sosem#1{Sosem beszelek #1 orajan}`
- Használjuk háromszor!
- Megoldás: hello.tex

T_EX challenge

írjuk ki ezerszer

Hints:

- `\show\loop`
- `\show\repeat`
- `\show\iterate`
- `\advance\count4142 by 1`
- `\ifnum\count4142 < 1000 almafa\else kortefa\fi`
- Megoldás: `ezerszer.tex`
- Megoldás 2: `ezerszer-lua.tex`
- Megnézni: `tizszer.tex`

T_EX catcodes

Interesting concept, not found elsewhere

- A T_EX tokenizerjét újra lehet programozni catcode-ok mentén
 - 0: escape (\)
 - 1: bgroup ({)
 - 2: eggroup (})
 - 3: math (\$)
 - 4: alignment (&)
 - 5: eol (enter)
 - 6: param (#)
 - 7: superscript (^)
 - 8: subscript (_)
 - 9: ignored (ascii null)
 - 10: space (space)
 - 11: letters (a..z A..Z)
 - 12: other (everything else)
 - 13: active (~)
 - 14: comment (%)
 - 15: invalid (delete)
-
- See catcodes-*.tex! **Once tokenized, stays tokenized!**

T_EX catcodes

Makró neve csak betűkből állhat... Tényleg?

- Common trick: makró névbe kényelmes lenne valami nem betű
- Hogy a `\bazihozzunemertenyiprogramozok` lehessen
`\bazi@hozzu@nev@mert@en@igy@programozok`
- Megoldás: kukacot betűvé kódolni `\def` előtt, aztán vissza
- Szokás: internal nevek tartalmazzanak kukacot, publikok nem
- L^AT_EX csomag elején beváltunk
- Ez elterjedt, ezért L^AT_EX-ben: `\makeatletter`, `\makeatother`
(A L^AT_EX úgy általában NAGYON NAGY és nem fér bele sehogy 2 órába.)
- nics catcodes: `nics/src/nics-slide.tex` vége

T_EX boxes

Overview

- Minden vboxokban lévő hboxok sokasága (demó: boxes.log)
- Expliciten is lehet őket készíteni: `\{v,h\}box{content}`
- Implicit dobozolásnak köze van a módokhoz:
 - Alapból vertical módban vagyunk, oldalakat tördelünk
 - Ha jön valami betű amit renderelni kell, akkor horizontal módba váltunk és sorokat tördelünk (ez a váltás implicit)
 - `\par` visszavált, a két enter rak egy `\par`-t
 - Ha matekba váltunk, arra van egy külön matek mód
- Amikor expliciten dobozolunk, akkor “restricted” h vagy v mód
- Restricted módban nincs tördelés (sor vagy oldal)
- Amikor non-restricted vmódban kész egy oldal, akkor shipout

nics boxes

A trükkök, amiket használunk

- `dimen`: távolság, pl. `1in`, `5cm`, `10pt`, use emacs to convert
- `glue`: rugó jobb név lenne,
flexibilis hármás (`dimen`, nyúlás, összenyomódás)
- Demó: `glue.tex`
- A `\hss` és `\vss` nagyon useful és sokrétű
- Mostmár meg tudjuk érteni a `\centerline` makrót

T_EX grouping

Ugyanaz, mint más nyelvekben a szkóp

- Ami kapcsosok között történik, az kapcsosok között marad
- `count`, `def`, `dimen`, `stb.`, kivéve, ha prefixed with `\global`
- Nem minden kapcsos group: pl. `def` vagy `{v,h}box` kapcsosa
- L^AT_EX-ben minden `begin...end` environment usage egy group
- Hosszú makrók problémásak, common trick: `\show\centerline`
- Kapcsosok kapcsos nélkül: `\bgroup` és `\egroup`
- L^AT_EX savebox demó: `savebox.tex`

T_EX documentation

Kicsit szétszórt

A dokumentáció for historical reasons nagyon szétszórt:

- T_EXBook for the basics, must read (but not all chapters)
- eT_EX: `etex_man.pdf`
- LuaT_EX: `luatex.pdf`
- Helper package: `luacode.pdf`
- L^AT_EX, ha muszáj valamit tudni: `source2e.pdf`
- TikZ, ha rajzolni kell: `pgfmanual.pdf`
- Egy nagyon érdekes dolog L^AT_EX utálóknak: `opmac`
- Ha jól állunk idővel, itt legyen TikZ demó és tanítás (basics)

nics code reading

- Mostmár mindent tudunk, hogy olvassunk egy kis kódot
 - `nics-cached.tex`
 - `nics-slide.tex`
 - `nics-content.tex`
- `nics-extern.tex` + `nics-extern.lua`
- `nics-zoom` + `nics-aftersetbox`

Live coding

Bug fixing

- Bug report: ha üres a második paraméter a title slideon, akkor is van sötétítő buborékja, pedig akkor inkább ne legyen!
- Write a test! Mentsük el az MD5-jét!
- Fix the bug! Csak a teszt MD5-je változhat