

Knihovna `scancsv.lua`

a její následovník

ConT_EXtový modul `t-scancsv.lua`

Radost ze zpracování hromadných dat . . .

Jaroslav Hajtmar

T_EXperience 2011

Železná Ruda 1. října 2011

Abstrakt

Při hromadném zpracování dat se často využívají údaje uložené v textových CSV (comma separated values) souborech. Tyto mohou být zpracovávány bez použití externích nástrojů přímo Plainem, L^AT_EXem či ConT_EXtem. V přednášce bude popsána knihovna ScanCSV.lua, umožňující zpracovávat CSV soubory prostřednictvím LuaT_EXu, LuaL^AT_EXu či ConT_EXtu MkIV. Ukážeme, jak lze velmi jednoduchým a přirozeným způsobem vytvářet hromadné tiskové sestavy (formuláře, vysvědčení, pozvánky, faktury, průkazky, vizitky, oboustranné kartičky atd.).

Základem přednášky je text, který zazněl v angličtině na loňském 4. ConT_EXt Meetingu v Břejlově. Připomínky vývojářů ConT_EXtu, které zazněly po loňské přednášce byly důležitým impulsem k zásadnímu přepracování původní knihovny ScanCSV.lua a k vzniku ConT_EXtového MkIV modulu t-scancsv.lua, který bude nyní poprvé veřejně představen.

Úvod

Knihovna SCANCSV.LUA:

- Jednoduchým způsobem zpracovává v ConT_EXtu (MkIV) (LuaL^AT_EXu, LuaPlainu) textová data uložená v externích CSV souborech.
- Umožňuje jednoduchým a přirozeným způsobem vytvářet v ConT_EXtu dokumenty, které jsou založeny na zpracování hromadných dat (jednoduchých CSV databází).
- Má široké využití: tisk různých formulářů, vysvědčení, pozvánek, průkazek, kartiček (i oboustranných), hromadných dopisů, pozvánek, vizitek, tabulek, animací atd.

Hlavní důraz byl při tvorbě knihovny kladen na:

- jednoduché použití bez znalosti jazyka Lua, univerzální užívání nejen v ConT_EXtu, ale i v LuaL^AT_EXu a LuaPlainT_EXu, zpřístupnění CSV dat prostřednictvím T_EXových maker postavených na knihovních funkcích (tj. bez Lua kódu), motivace dalších uživatelů k používání LuaT_EXu a jazyka Lua.
- Užitečnost a snadné používání ...

Při tvorbě ConT_EXtového modulu `t-scancsv.lua` byl důraz kladen na optimalizaci fungování v ConT_EXtu!

CSV formát a SCANCSV.LUA

- Výměna dat, export do CSV (z MySQL databází), jednodušší alternativa k XLM, lze snadno zpracovávat (třídít a editovat) v tabulkových procesorech (Excel, Calc, Gnumeric, ...)
- Popis CSV formátu obecně
- CSV formát vhodný pro SCANCSV.LUA:
 - soubor musí být kódován v UTF-8! (vyexportované XLS soubory se musí překódovat - nevýhoda)
 - Oddělovače polí: v podstatě libovolné, defaultně použít ';' středník (tj. a la Excel)
 - Vymezovače polí : v podstatě libovolné levý i pravý mohou být různé (nejčastěji bývají " uvozovky), defaultně nepoužito!!!
 - Parsovací algoritmus SCANCSV.LUA je velmi jednoduchý (lze jej však libovolně upravit) => omezení (pokud nastavíme vymezovače, musí být použity všude – obecně to tak být nemusí)

SCANCSV – historie vzniku, inspirace

- 2005 – nalezení makra `scanbase.tex` P. Olšáka. Zpracovává specifické textové soubory.
- Makro `scancsv.tex` - Petr Olšák - modifikace a zobecnění původního makra pro CSV formát. V plainu používáno do r. 2008.
- 2008 – modifikace pro L^AT_EX (Jaromír Kuben) a ConT_EXt (Petr Olšák).

Makro je použitelné v ConT_EXtu MkII i MkIV dodnes.

- 2010 – používám ConT_EXt MkIV - velké problémy s UTF-8 (nyní již vše OK).
- Duben 2010 – jazyk Lua a LuaT_EX. Programování knihovny `scancsv.lua`.

Důvod: zájem o Lua jazyk a o jiný způsob programování TeXových maker - použití v ConT_EXtu MkIV.

- Červen 2010 – první reálně použitelná verze, otestováno ostré nasazení v řadě praktických úloh, včetně agendy pro CTM a TE 2010, vylepšování, ladění a rozšiřování funkčnosti.
- Září 2010 – prezentace `scancsv.lua` na 4. ConTeXt Meetingu v Brejlově u Prahy. Cenné připomínky vývojářů ConTeXt - H. Hagen a W. Schustera.
- Červen 2011 – regulérní (lua)modul pro ConTeXt MkIV. Kompletní přepracování a optimalizace pro ConTeXt. Vylepšení, nové funkce. Otestováno v ostrém provozu.
- ??? - zohlednění připomínek a námětů, vytvoření dokumentace a manuálu.

Ukázka definice TeXového makra pomocí Lua

% ukázka definice makra v ConTeXtu pomocí Lua jazyka

```
\def\fsin#1{
  \directlua{x="#1"; tex.print("\sin("..x..") = "..math.sin(x));}
}
```

```
\def\Fsin#1{
  \startluacode
    x="#1";
    context("\sin("..x..") = "..math.sin(x));
  \stopluacode
}
```

```
\starttext
```

```
 $\fsin{3.14}$
```

```
 $\Fsin{2.431}$
```

```
\stoptext
```


Výsledek :

$$\sin(3.14) = 0.0015926529164868$$

$$\sin(2.431) = 0.65228310240891$$

Princip fungování knihovny

1. Načtení knihovny scancsv.lua (jediný Lua kód ve zdrojovém textu ConT_EXtu).
2. Volitelné nastavení příznaku hlavičky, separátoru a vymezovačů položek (jinak se použijí default hodnoty).
3. Otevření CSV souboru (různými způsoby).
4. Načítání řádku CSV tabulky (manuálně nebo v cyklu)
5. Rozparsování řádku (oddělení sloupcových dat).
6. Načtení sloupcových dat do T_EXových maker.
7. Opakování kroků 4.-6. pro všechny řádky CSV tabulky.

Způsob zpracování 1. řádku tabulky závisí na tom, zda je to „hlavička“ či nikoliv. Po načtení sloupcových dat do maker jsou data přístupná ConT_EXtu. Řádky lze procházet „manuálně“, prostřednictvím standardních cyklů nebo knihovním makrem.

Použití v „manuálním“ režimu

- Načtení knihovny:
`\directlua{dofile("scancsv.lua")}`
- Nastavení příznaku hlavičky (pokud je soubor s hlavičkou):
`\setheader`
(nebo zrušení nastavení - `\resetheader`).
- Otevření CSV souboru: `\opencsvfile{file.csv}`.
- Použití maker `\cA`, `\cB` ... á la Excel, resp. `\cI`, `\cII`, ... á la Roman, nebo `\Firstname`, `\Lastname`, ... pokud 1. řádek obsahuje hlavičku. Tato makra obsahují sloupcové hodnoty aktuálního CSV řádku
- `\nextrow` – přechod na další řádek tabulky (makra `\cA`, `\cB` ... nebo `\Firstname`, `\Lastname`, ... jsou naplněna novými hodnotami)

Hlavní T_EXová makra pro používání knihovny

- `\setfiletoscan{CSVFile}` – nastavení jména souboru
- `\setheader` – nastavení příznaku existence hlavičky
- `\resetheader` – vrácení nastavení na defaultní hodnotu
- `\setsep{,}, \setld{*}, \setrd{!}` – nastavení oddělovače sloupců a vymezovačů sloupců na vlastní hodnotu (nedefaultní)
- `\resetsep, \resetld, \resetrd` – reset na default hodnotu
- `\opencsvfile{CSVFile}` - otevření CSV souboru
- `\nextrow` – skok na další řádek CSV tabulky
- `\printline` – vypsání celého kompletního řádku
- `\printall` – vypsání celé kompletní tabulky

- `\filelineaction`
`\filelineaction{CSVfile}`
`\filelineaction{CSVfile}{do}`
`\filelineaction{CSVfile}{od}{do}`

tato makra zpracují uživatelem definované makro `\lineaction` (jméno je vyžadováno) v cyklu pro všechny řádky CSV souboru.

L^AT_EXoví uživatelé mohou použít pouze příkaz:

`\filelineaction{CSVfile}` (s jedním parametrem).

ScanCSV.lua rozlišuje různá prostředí (ConT_EXt nebo LuaL^AT_EX), v závislosti na tom definuje příslušná makra (kvůli kompatibilitě).

Příklad 1

```
% Zpracovávaný soubor - example1.csv:
% 1;Petr;Novák;19.5.1989;m;Nymburk;U Brány 7
% 2;Jan;Novotný;5.7.1991;m;Praha;Uhlířská 178
% ...

\directlua{dofile("scancsv.lua")} % Nahrát knihovnu

% vzor pro zpracování v cyklu
\def\lineaction{\item \cB\ {\bf \cC}\ (\cD)}

\starttext
List of participants:
  \startitemize[n]
    \filelineaction{example1.csv} % generování v cyklu
  \stopitemize
\stoptext
```

Výsledek příkladu 1

List of participants:

1. Petr **Novák** (19.5.1989)
2. Jan **Novotný** (5.7.1991)
3. Zuzana **Vašíčková** (13.9.1984)
4. Pavel **Brožkan** (14.8.1992)
5. Lenka **Rábelová** (8.11.1992)

Příklad 2

```
% example2.csv:
% Surname,Firstname,Birthdate,Sex,City,Zipcode,Street
% Novák,Jan,14.10.1997,m,Zbečno,27024,Farní 21
% Pospíšilová,Hana,4.1.1996,ž,Zábřeh,78901,Studénky 420

\directlua{dofile("scancsv.lua")}

% vzor pro zpracování v cyklu
\def\lineaction{\item \Birthdate: {\bf \Surname}\ \Firstname}

\starttext
\setheader % nastavení příznaku hlavičky
\setsep{,} % nastavení oddělovače polí (delimiter)

Seznam účastníků:

\startitemize[n]
  %\filelineaction{example2.csv} % Generování pro všechny řádky tabulky
  \filelineaction{example2.csv}{3} % pro první 3 řádky
  % \filelineaction{example2.csv}{3}{5} % od 3. do 5. řádku
\stopitemize

\stoptext
```


Výsledek příkladu 2

List of participants:

1. 14.10.1997: **Novák** Jan
2. 4.1.1996: **Pospíšilová** Hana
3. 15.8.1997: **Žáková** Eliška

Informační výpis zhotovený makrem `\csvreport`

Current CSV file report

Input CSV file: : **example2.csv**

Separator (delimiter) and "quoters" see Lua variables `Sep`, `Ld` a `Rd`

Current settings of delimiters and quoters: `pole1`, `pole2`, `pole3`, ...

Number of columns in a table: **7**

Number of rows in the table: **5**

Macros supplying columns data in each row of table:

`\cA=\Surname`, `\cB=Firstname`, `\cC=Birthdate`, `\cD=Sex`, `\cE=City`,

`\cF=Zipcode`, `\cG=Street`,

Additional predefined macros:

`\csvfilename` – name of open CSV file (**example2.csv**)

`\numcols` – number of table columns (**7**)

`\numrows` – number of currently processed lines (**5**)

`\numline` – number of the currently loaded row (for use in print reports)

`\csvreport` – prints the report on file open

`\printline` – lists the current CSV row table in a condensed form

`\printall` – CSV output table in a condensed form

`\setfiletoscan{filename}` – setting of name of CSV file

`\opencsvfile{filename}` – open CSV table

`\openheadercsvfile{filename}` – open CSV table (and set header flag)

`\setheader` – set a header flag

`\resetheader` – unset a header flag

`\readrow` – next row of CSV table

`\nextrow` – next row of CSV table (with test of EOF)

`\setsep{separator}` – set separator of columns

`\resetsep` – unset to default values

`\setld{delimiter}` – set left quoter

`\resetld` – unset left quoter to default values

`\setrd{delimiter}` – set right quoter

`\resetrd` – unset right quoter to default values

`\blinehook` – begin line hook macro (process before first column value of each row)

`\elinehook` – end line hook macro (process after last column value of each row)

`\bfilehook` – begin file hook macro (process before whole file processing)

`\efilehook` – end file hook macro (process after whole file processing)

Příklad 3 – generátor tabulky

```
%  
% example2.csv:  
% Surname,Firstname,Birthdate,Sex,City,Zipcode,Street  
% Novák,Jan,14.10.1997,m,Zbečno,27024,Farní 21  
% Pospíšilová,Hana,4.1.1996,ž,Zábřeh,78901,Studénky 420  
% .....  
  
\directlua{dofile("scancsv.lua")}  
  
% vzor pro zpracování v cyklu  
\def\tableaction{\expanded{  
  \bTR  
  \bTD \numline. \eTD  
  \bTD \Firstname\ \Surname \eTD  
  \bTD \Street, \Zipcode\ \City \eTD  
  \eTR}  
}  
  
\starttext  
\setheader % nastavení hlavičky  
\setsep{,} % nastavení nedefaultního oddělovače (delimiter)  
\opencsvfile{example2.csv} % otevřít CSV soubor
```

List of participants:

```
\blank[big]
```

% Příklad užití standardního ConTeXtového cyklu

```
\bTABLE
```

```
  %\dorecurse{4}{\tableaction\nextrow} % pro první 4 řádky
```

```
  \doloop{\ifEOF\exitloop\else\tableaction\nextrow\fi} % všechny řádky
```

```
\eTABLE
```

```
\stoptext
```

Výsledek příkladu 3

List of participants:

1.	Jan Novák	Farní 21, 27024 Zbečno
2.	Hana Pospíšilová	Studénky 420, 78901 Zábřeh
3.	Eliška Žáková	Radniční 38, 78901 Zábřeh
4.	Jan Novotný	Uhlířská 178, 11150 Praha
5.	Přemysl Oráč	Rybniční 25, 75421 Mošnov

Příklad 4 – užití "hooks" pro výpis tabulky

```
% example2.csv:
% Surname,Firstname,Birthdate,Sex,City,Zipcode,Street
% Novák,Jan,14.10.1997,m,Zbečno,27024,Farní 21
% Pospíšilová,Hana,4.1.1996,ž,Zábřeh,78901,Studénky 420

\directlua{dofile("scancsv.lua")}

% definice "hooks":
\def\bfilehook{\bTABLE}
\def\efilehook{\eTABLE}
\def\blinehook{\bTR}
\def\elinehook{\eTR}

% vzor pro použití v cyklu
\def\lineaction{\expanded{
  \bTD \numline. \eTD
  \bTD \Firstname\ \Surname \eTD
  \bTD \Street, \Zipcode\ \City \eTD}
}
```

```
\starttext
\setheader % nastavení hlavičky
\setsep{,} % oddělovač polí (delimiter)

List of participants:\blank[big]

% pouze od 3. do 5. řádku tabulky example2.csv
\filelineaction{example2.csv}{3}{5}

\stoptext
```

Výsledek příkladu 4

List of participants:

1.	Eliška Žáková	Radniční 38, 78901 Zábřeh
2.	Jan Novotný	Uhlířská 178, 11150 Praha
3.	Přemysl Oráč	Rybniční 25, 75421 Mošnov

Příklad 5 – použití Lua^LA^TE_Xu

```
% database.csv:
% Id;Surname;Firstname;City;ZIP;Street;Country
% 1;Surname1;Firstname1;City1;ZIP1;Street1;Country1
% 2;Surname2;Firstname2;City2;ZIP2;Street2;Country2
%
% Kompilace: lualatex scancsv-envelope.tex

\documentclass[business]{envelope}
\usepackage[utf8]{luainputenc}
\usepackage{luatextra}

\def\lineaction{\Addressee{
    \Surname\ \Firstname \
    \Street \
    \City\ \ \ZIP \ \ }
    \makeEnvelope\newpage}

\begin{document}
  \directlua{dofile("scancsv.lua")}
  \setheader
  \filelineaction{database.csv}
\end{document}
```

Výsledek příkladu 5

Waroc Informatik
Endeavour House
11 Kingsgate PI
Bolton ON L7E 5Z5

Surname1 Firstname1
Street1
City1 ZIP1

SCANCSV.LUA a "bláznivé CSV tabulky"

Ukázka "bláznivé" CSV tabulky:

```
Id;Align;Color;LineMacro;lineaction;Prijmeni;Jmeno;DatumNarozeni;Pohlavi;Mesto;PSC;Ulice
1;left;\red;{\framed[width=5cm,height=3cm]{\numline: {\Color\Jmeno\ \Prijmeni}} };{\numline -
\Prijmeni\ {\Color\Jmeno}\par\LineMacro};Novák;Jan;14.10.1997;m;Zbečno;27024;Farní 21
2;right;\blue;{\framed[width=5cm,height=3cm]{\numline: {\Color\Prijmeni\ \Jmeno}} };{\numline
- \Jmeno\ {\Color\Prijmeni}\par\LineMacro};Pospíšilová;Hana;4.1.1996;ž;Zábřeh;78901;Studénky
420
3;middle;\orange;{\framed[width=3cm,height=5cm]{\numline: {\Color\Jmeno\ \Prijmeni}} };{\numline
- Jméno = {\Color\Jmeno}\par\LineMacro};Žáková;Eliška;15.8.1997;ž;Zábřeh;78901;Radniční 38
4;right;\magenta;{\framed[width=2cm,height=4cm]{\numline: {\Color\Jmeno\ \Prijmeni}} };{\numline
- Přijmení={\Color\Prijmeni\LineMacro}\par};Novotný;Jan;5.7.1991;m;Praha;11150;Uhlířská 178
5;left;\green;{\framed[width=6cm,height=3cm]{\numline: {\Color\Jmeno\ \Prijmeni}} };{\numline
- XXL \LineMacro};Oráč;Přemysl;11.5.1989;m;Mošnov;75421;Rybniční 25
...
```

a tabulky použité pro ukázkovou PDF animaci:

```
Number;Pointcolor;Linecolor;Text;Radius
1;\red;\blue;Zkušební text \the\numexpr\Number+\numcols\relax;30
2;\blue;\green;Lorem ipsum \the\numexpr\Number+\numline+55\relax;25
3;\cyan;\blue;Lorem \the\numexpr\Number-\numline+107\relax;20
4;\yellow;\green;Lorem sit \the\numexpr\Number+26+\numline\relax;18
5;\green;\blue;Dim = \the\dimexpr(\vsize+\numline\hsize-2730pt);16
...
```

Příklad "crazy" CSV tabulky

```
% File: makravcsvsouboru.csv
% Id; Barva; Align; Color; LineMacro; lineaction; Prijmeni; Jmeno;

% 1; orange; left; \darkgreen; \dc{\framed[width=5cm, height=3cm, background=color, backgroundcolor = FrColor]{\numline: {\Color\Jmeno\ \Prijmeni}}} }; {\numline - \Prijmeni\ {\Color \Jmeno}\par \LineMacro}; Novák; Jan

% 2; magenta; right; \blue; \dc{\framed[width=5cm, height=3cm,background=color, backgroundcolor = FrColor]{\numline: {\Color\Prijmeni\ \Jmeno}}} }; {\numline - \Jmeno\ {\Color\Prijmeni}\par \LineMacro}; Pospíšilová; Hana

% 3; green; middle; \orange; \dc{\framed[width=3cm, height=5cm, background=color, backgroundcolor = FrColor]{\numline: {\Color\Jmeno\ \Prijmeni}}} }; {\numline - Jméno = {\Color \Jmeno}\par \LineMacro}; Žáková; Eliška

% 4; yellow; right; \magenta; \dc{\framed[width=2cm, height=4cm, background=color, backgroundcolor = FrColor]{\numline: {\Color\Jmeno\ \Prijmeni}}} }; {\numline - Příjmení= {\Color \Prijmeni \LineMacro}\par}; Novotný; Jan

% 5; gray; left; \green; \dc{\framed[width=6cm, height=3cm, background=color, backgroundcolor= FrColor]{\numline: {\Color\Jmeno\ \Prijmeni}}} }; {\numline - XXL \LineMacro}; Oráč; Přemysl

\directlua{dofile("scancsv.lua")} % Nahrát knihovnu

\let\LineMacro\relax
\let\lineaction\relax
\def\dc{\definecolor[FrColor][\Barva]}

\starttext
\setheader
\filelineaction{makravcsvsouboru.csv}
\stoptext
```

Výsledek zpracování "bláznivých dat"

1- Novák Jan

1: Jan Novák

2- Hana Pospíšilová

2: Pospíšilová Hana

3- Jméno = Eliška

3: Eliška
Záková

4: Jan
Novotný

4- Příjmení= Novotný

5: Přemysl Oráč

5- XXL

Makra pro zpřístupnění údajů ve sloupcích tabulky

CSV soubor bez hlavičky (default volba - \resetheader)

\cA **\cB** **\cC** **\cD** ... \resetheader

↑ ↑ ↑ ↑

1;Petr;Novák;19.5.1989;m;Nymburk;U Brány 7
2;Jan;Novotný;5.7.1991;m;Praha;Uhlířská 178
3;Zuzana;Vašíčková;13.9.1984;ž;Ostrava;Jánská 14
...

není hlavička
datové řádky

CSV soubor s hlavičkou (zapnutí - \setheader)

\cA = \Surname **\cB = \Firstname** **\cC = \Birthdate** ... \setheader

↑ ↑ ↑

Surname;Firstname;Birthdate;Sex;City;Zipcode;Street ← hlavička
Novák;Jan;14.10.1997;m;Zbečno;27024;Farní 21
Pospíšilová;Hana;4.1.1996;ž;Zábřeh;78901;Studénky 420
...

datové řádky

Možnost nastavení římského číslování:

\cI, \cII, \cIII, \cIV, ... (default UserColumnNumbering='XLS')

Makra pro získání „systémových“ informací

- `\csvfilename` – název aktuálně otevřeného CSV souboru
- `\numcols` – počet sloupců aktuálně zpracovávané CSV tabulky
- `\numrows` – počet řádků CSV tabulky
- `\numline` – pořadové číslo aktuálně zpracovávaného řádku
- `\csvreport` – Informační report - zevrubné informace o otevřeném CSV souboru

Hooks - háčky pro zpracování

- `\blinehook`, `\elinehook` – begin line hook, end line hook - tato makra se vykonají před a po zpracováním daného řádku makrem `\lineaction`.
- `\bfilehook`, `\efilehook` – vykonají se před a po zpracování celé CSV tabulky.
- `\bch`, `\ech` – begin column hook, end c.h. - jejich použití musí být realizováno změnou ve zdrojovém kódu knihovny (z důvodu nemožnosti testování je tato volba defaultně vypnuta).

Standardní hodnoty všech hooks jsou `\relax`.

Modifikace základních funkcí knihovny

- Výchozí nastavení lze změnit úpravou zdrojového textu souboru `scancsv.lua` v úvodní části kódu.
- Při zpracování ConT_EXt em MkIV (LuaL_AT_EX em) je možné průběžně měnit nastavení oddělovačů (delimiter), vymezovačů, hlavičky atd. pomocí T_EX ových maker
- Možnost zpracování několika různých CSV souborů v jednom dokumentu (s různými oddělovači a separátory sloupců atd.).
- Používání hooks – defaultně jsou `\relax`.

TeXové IF pro testování konce (EOF) CSV souboru

- `\ifEOF` – TRUE, pokud se dostaneme na konec CSV souboru.
- `\ifnotEOF` – opak `\ifEOF`.

Testování and cykly

Podmínky AND a OR (viz Olšák TBN).

```
% Condition A AND B
\doloop{
\ifnum\Id>2
    \ifnum\Id<10\lineaction
    \fi
\fi
\ifEOF\exitloop\else\nextrow\ifEOF\exitloop\fi\fi
}
```

```
-----
% Condition A OR B
\def\AorB{\lineaction}
\doloop{
\ifnum\Id=1\AorB%
\else\ifnum\Id>3\AorB\fi
\fi
\ifEOF\exitloop\else\nextrow\ifEOF\exitloop\fi\fi
}
```

SCANCSV.LUA a cykly

Příklady ConT_EXt ových cyklů:

- `\dorecurse{5}{\lineaction\nextrow}`
- `\doloop{\lineaction\nextrow\ifnum\numline>7\exitloop\fi}`
- `\doloop{\ifEOF\exitloop\else\lineaction\nextrow\fi}`
- `\doloop{\lineaction\nextrow \if\ld3 \exitloop \fi}`

Příklady knihovních cyklů SCANCSV.LUA:

Makra jsou odvozena z makra `\doloop`.

- `\doloopwhile{\Class}{3.A}{\tableaction}`
- `\doloopuntil{\Class}{3.A}{\tableaction}`
- `\doloopforall{\lineaction}`
- `\doloopfromto{3}{7}{\lineaction}`
- `\doloopaction`
- `\doloopaction{\useraction}`
- `\doloopaction{\useraction}{5}`
- `\doloopaction{\useraction}{5}{7}`

Omezení, kompatibilita, nedostatky

- SCANCSV.LUA nezpracovává obecné CSV soubory. Důvod: Parsovací algoritmus je velmi jednoduchý. Pokud je položka obsahuje sloupcový oddělovač (delimiter) “,” pak je správný CSV výstup realizován jako:
`1, Jan, Novotny, "The Gate 4, 111 50 Prague", ...`
V tuto chvíli vyřešeno zatím pouze v novém ConT_EXt ovém modulu. Ten je již v tomto ohledu zobecněn a umožňuje zpracovávat i takovéto CSV soubory. V případě zájmu je možné upravit i kód původního SCANCSV.LUA.
- Občasné problémy s expanzí. Např. se mi nepodařilo zprovoznit SCANCSV.LUA v databázovém modulu (`\usemodule [database]`) od Mojca Miklavec.
- Některé věci fungují pouze v ConT_EXt u. Lze řešit řádným testováním - nemám k dispozici.
- Uvědomujme si, že CSV soubory mohou obsahovat znaky `%`, `&`, `_`, atd. Je třeba před zpracováním změnit catcodes těchto znaků.

Možnosti vylepšení a rozšíření ...

- Nové cykly, funkce, ...
- Vyčištění kódu,
- Vytvoření dobré dokumentace a uživatelského manuálu
- Zlepšení a zobecnění algoritmů (aktuální algoritmy dostačují mým soukromým potřebám.
- Vytvoření samostatného MkIV modulu (tj. odstranění omezení plynoucích s přísným dodržováním kompatibility)
- Realizace návrhů vylepšení od uživatelů

Modul `t-scansv.lua` pro ConTEXT

Důvody vzniku `scansv.lua`

- seznámení se s jazykem Lua (před začátkem práce - nulové zkušenosti s Lua)
- v té době nefungovalo Olšákovo makro v MkIV (pův. pro osobní potřebu)
- podnět P. Stríže -> pro širší okruh uživatelů

Důvody vzniku modulu `t-scansv.lua`

- úprava zdrojového kódu na základě připomínek H.Hage-
na a W.Schustera
- mírně pokročilá zdatnost práce s Lua
- obtíže a omezení plynoucí ze striktního umělého udr-
žování kompatibility
- výhradní používání ConTEXT u -> vlastní potřeba

Použití modulu `t-scancsv.lua`, rozdíly

Původní kód byl kompletně a zásadně přepracován a optimalizován (připomínky a rady vývojářů ConT_EXt u)! Funkčnost modulu je navenek stejná jako u `scancsv.lua` - tj. modul zachovává jistou kompatibilitu! (kvůli mým starým zdrojovým kódům, používající původní kód)

Zavedení `scancsv.lua`:

```
\directlua{dofile("scancsv.lua")}
```

Zavedení ConT_EXt ového modulu `t-scancsv.lua`:

```
\usemodule[t-scancsv] - jediný rozdíl (ale lze zavést i původním způsobem)
```

Od tohoto okamžiku vše funguje navenek stejně jako u `scancsv.lua`!

Vylepšení - hlavně nový cyklus ...

```
\doloopif#1#2#3#4
```

```
\doloopif{value1}{[compare_operator]}{value2}{macro_for_doing}
```

```
[compare_operators]: <, >, ==(eq), ~=(neq), >=, <=, in, while, until
```

Příklady:

- `\doloopif{\Trida}{==}{3.A}{\tableaction}`
- `\doloopif{\Vek}{>=}{35}{\tableaction}`
- `\doloopif{\Jmeno}{~=}{Petr}{\tableaction}`
- `\doloopif{Ja}{in}{\Prijmeni}{\tableaction}`

Ukázky reálného nasazení a použití

Poděkování...

- Členům mailové konference `ntg-context@ntg.nl` za jejich rady týkající se ConT_EXtu and Lua. Knihovna by bez jejich asistence nevznikla. Speciálně díky Taco Hoekwaterovi, Hansu Hagenovi, Wolfgangu Schusterovi.
- Členům mailové konference `cstex@cs.felk.cvut.cz` za jejich rady týkající se T_EX u and L^AT_EX u. Speciálně díky Zdeňku Wagnerovi, Vitu Zýkovi, Pavlu Strížovi, Petru Olšákovi, ...
- Pavlu Strížovi za inspiraci, testování rady a to, že mně přiměl knihovnu dokončit do fáze, kdy ji může použít i někdo jiný než já.

Linky k příkladům

<http://jaroslav.hajtmr.com/storage/tex/ctm+te2010/presentation>

a zde je [direct link to file browser](#)