

УДК 681.322

Г.М.Губаль

Луцький національний технічний університет

## АНІМАЦІЯ В МАТЕМАТИЧНИХ ТЕКСТАХ НА МОВІ L<sup>A</sup>T<sub>E</sub>X

У статті досліджується можливість конвертації T<sub>E</sub>X-скриптів у html-формат. Розглядається створення анімації графічних об'єктів у математичних текстах на мові L<sup>A</sup>T<sub>E</sub>X.

**Ключові слова:** мова L<sup>A</sup>T<sub>E</sub>X, T<sub>E</sub>X-скрипти, конвертор, pdfL<sup>A</sup>T<sub>E</sub>X, анімація, dvips, ps2pdf, програма Adobe Reader.

**Лит. 8.**

L<sup>A</sup>T<sub>E</sub>X – видавнича система для створення математичних текстів і для програмування [1-8].

Для всіх світових наукових видань єдиним форматом є L<sup>A</sup>T<sub>E</sub>X. Проте для розміщення документів у Інтернеті часто використовують html-формат документів. У цьому випадку застосовують програми для перетворення формату L<sup>A</sup>T<sub>E</sub>X у html.

Умовно програми для конвертації T<sub>E</sub>X-скриптів у html-формат можна поділити на два види:

– програми, які модифікують поведінку T<sub>E</sub>X для виведення в html-формат (T<sub>E</sub>X4ht, hyperL<sup>A</sup>T<sub>E</sub>X);

– програми, які самі обробляють і перетворюють вихідний L<sup>A</sup>T<sub>E</sub>X/T<sub>E</sub>X код (L<sup>A</sup>T<sub>E</sub>X2html, HeVeA).

Програма T<sub>E</sub>X4ht визначна тим, що не вимагає яких-небудь втручань у результуючий html код, дозволяючи достатньо гнучко і майже зрозуміло конфігурувати процес перетворення і досягати необхідного результату:

– дозволяє задавати оточуючі html-теги для T<sub>E</sub>X заданих команд і оточень;

– редагувати css.

Конвертор T<sub>E</sub>X4ht розуміє всі команди визначені користувачем, бо є драйвером для T<sub>E</sub>X (як наприклад, pdfL<sup>A</sup>T<sub>E</sub>X). Крім того, програма T<sub>E</sub>X4ht вміє працювати з такими модулями, як makeind, multind, hyperref і т.д.

Запускається T<sub>E</sub>X4ht за допомогою такої команди:

```
htlatex filename "options1" "options2" "options3" "options4".
```

Параметр options1 містить перелік налаштувань через кому. Починається з команди html, xhtml або імені конфігураційного файлу. Після цього вказується поділ html-файлів відповідно до поділу частин, розділів документа. Параметри options2 і options3 містять перелік налаштувань через пробіл. Ці налаштування вказуються для постобробників T<sub>E</sub>X4ht.c і T4ht.c. Параметр options4 містить налаштування, які передаються компілятору L<sup>A</sup>T<sub>E</sub>X.

Для tex-файла процес запускається так:

```
htlatex clm.tex 'cltl,3,next,charset=utf-8' '-cunihtf -utf8' '' '- interaction=batchmode',
```

тобто компілюється файл clm.tex, використовується конфігураційний файл cltl.cfg, розділяються html-файли за трьома рівнями вкладення частин, розділів, встановлюється кодування в utf-8, після чого в постобробник вказується кодування utf8, а також параметр для L<sup>A</sup>T<sub>E</sub>X для ігнорування помилок і роботи в пакетному режимі.

Конфігураційний файл встановлюється в першій позиції параметра options1 без вказівки розширення файла (.cfg). Наведемо шаблон файла конфігурації

```
\Preamble{html}
```

```
\begin{document}
```

```
.....
```

```
команди налаштування
```

```
.....
```

```
\EndPreamble
```

Css для вихідного документа можна вказати так:

```
\Css{content}
```

```
\Css content \EndCss  
\CssFile[list-of-css-files] content\EndCssFile
```

Наприклад,

```
\Css{  
  table.tabbing{  
    border-width: 1px;  
    border-spacing: 1px;  
    border-style: none;  
    border-collapse: collapse;  
  }  
  .centerline {  
    text-align: center  
  }  
  tt {  
    font-family: monospace, monospace;  
  }  
  div.lisp {  
    color: blue;  
    padding: 2em  
  }  
}
```

Конфігурація html-виведення для L<sup>A</sup>T<sub>E</sub>X-оточень здійснюється за допомогою функції виведення html-тегів \Hcode і команди конфігурації оточення \ConfigureEnv{environment-name} {before-environment} {after-environment} {before-list} {after-list}. Наприклад,

```
\ConfigureEnv{lisp} {\Tg<div class=lisp>} {\Tg<div>} {} {}  
\ConfigureEnv{defun} {\Tg<div class=defun>} {\Tg<div>} {} {}
```

Щоб виконати T<sub>E</sub>X-код за умови використання T<sub>E</sub>X4ht, наприклад, для T<sub>E</sub>X4ht явно вказати розширення графічного файлу, необхідно визначити дві варіанти команди для включення ілюстрацій \includefigure

```
\ifx \Hcode\Undef  
\newcommand{\includefigure}[1]{\center{\includegraphics[width=1\linewidth]{#1}}}  
\else  
\newcommand{\includefigure}[1]{\center{\includegraphics[width=1\linewidth]{#1.png}}}  
\fi
```

Перший рядок перевіряє наявність T<sub>E</sub>X4ht команди \Hcode і виконує одну з двох дій.

Програма T<sub>E</sub>X4ht підтримує вставку ілюстрацій, бібліографії і посилань. Ілюстрації вставляються за допомогою команд пакету graphicx, причому розширення графічного файлу необхідно явно вказати.

Для створення предметного покажчика створюється \*.idx файл командою htlatex. \*.ind файл, який використовується у вихідному документі створює така команда:

```
tex '\def\filename{{FILENAME }}{idx}{4dx}{ind}} \input idxmake.4ht'  
makeindex -0 FILENAME.ind FILENAME.4dx
```

Досить часто можна зустрічати документи з рисунками, які не мають відповідного стилю. Рисунки, які містять слова і символи повинні відповідати тексту у документі. Спочатку не наголошувалось на використанні системи T<sub>E</sub>X для створення графічних об'єктів. Проте існує багато інструментів (PSTricks, pgf/TikZ, METAPOST), які генерують рисунки високої якості. Такі рисунки можна вставляти в документи, і вони відповідатимуть стилю тексту.

Для слайд шоу та для цілей викладання необхідно створювати анімаційні картини.

L<sup>A</sup>T<sub>E</sub>X пакет "animate" використовується для створення pdf-файлів з анімаційним вмістом графічних об'єктів, таких як L<sup>A</sup>T<sub>E</sub>X-картинки, PSTricks-картинки або PDF/TikZ-картинки. На

відміну від стандартних відео-файлів пакет "animate" створює анімацію векторної графіки. Результат подібний до swf (Flash) формату.

Пакет "animate" підтримує процес створення pdf-файлів: pdfL<sup>A</sup>T<sub>E</sub>X, L<sup>A</sup>T<sub>E</sub>X → dvips → ps2pdf/Distiller.

Остаточний pdf-файл можна переглянути у програмі Adobe Reader.

Пакет "animate" можна підключати за допомогою такої команди:

```
\usepackage[<package options>]{animate}
```

у преамбулі документа.

'animate' оголошує опції пакета:

```
dvipdfmx  
xetex  
autoplay  
autopause  
autoresume  
loop  
palindrome  
draft  
final  
controls  
buttonsize=<size>  
buttonbg=<colour>  
buttonfg=<colour>  
step  
useocg  
poster[=first | none | last]
```

Усі наведені опції, крім перших двох, доступні як командні опції. Командні опції здійснюють глобальні налаштування. Якщо наведені опції використовувати як опції пакета, то вони створюють ефекти на всіх анімаціях у документі.

Якщо pdf-файл створений через DVI і Postscript командною послідовністю latex → dvips → ps2pdf, то необхідний пакет 'graphicx'.

Якщо анімаційна послідовність графічних файлів має бути вбудованою багаторазово в документ, то більш компактне виведення pdf-файла одержується за допомогою опції '-dMaxInlineImageSize=0' to ps2pdf.

Іноді необхідно запустити L<sup>A</sup>T<sub>E</sub>X двічі для створення посилань на об'єкт у документі.

Пакет "animate" містить команду

```
\animategraphics[<options>]{<frame rate>}{<file basename>}{<first>}{<last>}
```

і оточення

```
\begin{animateinline}[<options>]{<frame rate>}  
.....надрукований матеріал.....  
\newframe[<frame rate>]  
.....надрукований матеріал.....  
\newframe*{<frame rate>}  
.....надрукований матеріал.....  
\newframe  
\multiframe{<number of frames>}{[<variables>]}{  
.....параметризований матеріал.....  
}  
\end{animateinline}
```

Коли \animategraphics збирає анімацію з графічних файлів або з багатосторінкового pdf-файла, оточення 'animateinline' створює анімацію з включеного в нього надрукованого матеріалу. Цей матеріал може бути створений у вигляді картинок з використанням можливостей PSTricks,

pgf/TikZ або L<sup>A</sup>T<sub>E</sub>X оточення 'picture'. Таким способом можна здійснити анімацію навіть із звичайного текстового матеріалу. Параметр <frame rate> задає число кадрів за секунду анімації.

Команда \newframe завершує кадр і починає наступний. Вона може використовуватись лише всередині оточення 'animateinline'. Команда \newframe\* створює паузу анімації на даному кадрі. Анімація продовжується після клацання на ній. Обидві команди \newframe мають аргумент, який змінює частоту кадрів всередині анімації.

Команда \multiframe створює повторення картинок. Перший аргумент цієї команди задає число кадрів. Другий аргумент задає список оголошених змінних через кому. Список може мати довільну довжину. У третьому аргументі змінні використовуються для параметризації картинок. Оголосимо одну змінну так:

```
<variable name>=<initial value>+<increment>
```

<variable name> – це послідовність однієї або більше літер без backslash. Перша літера імені змінної визначає тип змінної. Існує три різних типи: цілі ('i', 'I'), дійсні ('n', 'N', 'r', 'R') і L<sup>A</sup>T<sub>E</sub>X довжини ('d', 'D'). На першій ітерації змінна набуває значення <initial value>. Кожна наступна ітерація збільшує змінну на <increment>. Перед від'ємними зростаннями ставиться '-'. Змінні досягають їх поточного значення, якщо імені змінної передує backslash. Команда \multiframe повинна бути в рамках оточення 'animateinline' або в рамках команд \newframe. Дві послідовні команди \multiframe необхідно розділяти однією з команд \newframe.

Анімація будується покадрово за порядком включення матеріалу. Однак, контроль порядку появи, накладання і повторення матеріалу доступний через опцію 'timeline'.

Усі графічні файли послідовності повинні бути занумеровані. <file basename> – це крайня ліва частина імені файлу, яка є загальною для всіх членів послідовності. <first>, <last> – це, відповідно, номер першого і останнього файлу послідовності. Імена файлів можуть бути занумеровані так: 0...99. Якщо номери файлів починаються з нулів, то слід перевірити, чи всі вони мають однакову кількість цифр.

Команда \graphicspath{} з пакета 'graphicx' використовується для визначення директорій для графічних файлів.

Формати файлів залежать від драйвера виведення. З L<sup>A</sup>T<sub>E</sub>X+dvips пошук файлів здійснюється у такій послідовності: 'eps', 'mps', 'ps'; з pdfL<sup>A</sup>T<sub>E</sub>X маємо таку послідовність: 'pdf', 'mps', 'png', 'jpg', 'jpeg', 'jbig2', 'jb2', 'jp2', 'j2k', 'jpx'.

Файл '<file basename>.(pdf|jbig2|jb2)' сприймається як багатосторінковий документ, в якому кожна сторінка представляє один кадр анімації. Аргументи <first> і <last> визначають ряд сторінок, включених в анімацію. Якщо ці обидва аргументи пропущені, то маємо нумерацію від 0 до  $n-1$ , де  $n$  – загальне число сторінок. Аргументи, які знаходяться за межами даного ряду, автоматично коректуються. Якщо <first> більший від <last>, то сторінки вбудовуються у протилежному порядку. Наприклад, команда

```
\animategraphics{12}{pict}{}{}
```

створить анімацію з усіх сторінок файлу 'pict.pdf' в 12 кадрів за секунду.

Наведемо приклад, який демонструє експоненціальну функцію  $y = e^x$ , створену програмою PSTricks, і її наближення многочленами Тейлора різного порядку.

```
\documentclass{article}
\usepackage{animate}
\usepackage{graphicx}

\begin{filecontents}{timeline.txt}
::0x0 % система координат та функція  $y = e^x$ , яка повторюється до останнього кадру
::1 % одна синя лінія за кадр
::2
::3
::4
::5
::6
::7
::8
```

```
\end{filecontents}  
\begin{document}  
\begin{center}  
  \animategraphics[  
    controls, loop,  
    timeline=timeline.txt  
  ]{6}{e_}{0}{8}  
\end{center}  
\end{document}
```

1. Кнут Д. Е. Всё про Т<sub>E</sub>X / Пер. с англ. М. В. Лисиной. – Протвино: АО RDT<sub>E</sub>X, 1993.
2. Жуков М. Ю. Оформление математических текстов при помощи пакета L<sup>A</sup>T<sub>E</sub>X<sub>2 $\epsilon$</sub>  / М. Ю. Жуков, Е. В. Ширяева. – Ростов н/Д: Изд-во ООО «ЦВВР», 2003.
3. Жуков М. Ю. L<sup>A</sup>T<sub>E</sub>X<sub>2 $\epsilon$</sub> : искусство набора и вёрстки текстов с формулами / М. Ю. Жуков, Е. В. Ширяева. – Ростов н/Д: Изд-во ЮФУ, 2009.
4. Львовский С. М. Набор и вёрстка в системе L<sup>A</sup>T<sub>E</sub>X. – 3-е изд., испр. и доп. – М.: МЦНМО, 2003.
5. Goossens M. The L<sup>A</sup>T<sub>E</sub>X companion / M. Goossens, F. Mittelbach, A. Samarin. – Addison-Wesley, 1994. Русский перевод: Гуссенс М. Путеводитель по пакету L<sup>A</sup>T<sub>E</sub>X и его расширению L<sup>A</sup>T<sub>E</sub>X<sub>2 $\epsilon$</sub>  / М. Гуссенс, Ф. Миттельбах, А. Самарин. Перевод с английского Маховой О. А., Третьякова Н. В., Тюменцева Ю. В. и Чистякова В. В. под редакцией Маховой И. А. – М.: Мир, 1999.
6. Беляков Н. С. T<sub>E</sub>X для всех / Н. С. Беляков, В. Е. Палаш, П. А. Садовский. – Либроком, 2009.
7. Ширяева Е. В. Введение в T<sub>E</sub>X-программирование / Е. В. Ширяева, И. В. Ширяева. – Ростов н/Д: Изд-во ЮФУ, 2010.
8. Mittelbach F. The L<sup>A</sup>T<sub>E</sub>X<sub>3</sub> Project / F. Mittelbach, C. Rowley // TUGboat. – 1997. – Vol. 18, No. 3. – P. 195-198.