



Ł^AT_EX3 Project

Linia rozwojowa Ł^AT_EX_A

Zalety Ł^AT_EX_A:

Potrzeby ulepszeń w wersji 3

Interfejs projektanta typograficznego

Szablony typograficzne (*templates*)

Mechanizm pobierania argumentów poleceń

Nowa procedura wyjścia (*output routine*)

Podsumowanie

Marcin Woliński
L^AT_EX 3 nadchodzi



BachoT_EX 2001



L^AT_EX3 Project

Cel: Opracowanie nowej wersji L^AT_EXa

Leslie Lamport
Frank Mittelbach
Chris Rowley
Rainer Schöpf

<http://www.latex-project.org>

Linia rozwojowa L^AT_EXa

L^AT_EX 2.09 1985

L^AT_EX 2_ε 1994

L^AT_EX 3 200?

Linia rozwojowa L^AT_EX_α

Fragmenty L^AT_EX_α 3 dostępne są w postaci pakietów dla L^AT_EX_α 2_ε
pod adresem:

<http://www.latex-project.org/code/experimental/>

Ta hybryda nazywana jest L^AT_EX 2_ε*

Linia rozwojowa L^AT_EXa

L ^A T _E X 2.09	1985
L ^A T _E X 2 _ε	1994
L ^A T _E X 2 _ε *	2000
L ^A T _E X 3	200?

Zalety \LaTeX a:

- ugruntowany standard adiustacji tekstu, formuł matematycznych i tabel,
- rozdzielenie abstrakcyjnej adiustacji od formatowania,
- łatwość użycia (dla autorów),
- przenośność dokumentów,
- przystosowywalność do różnych języków,
- dostępność, niski koszt :-),
- wsparcie i utrzymywanie systemu przez *$\text{\LaTeX}3$ Project Team*.

Potrzeby ulepszeń w wersji 3

- możliwość ścisłego wiązania tekstu z elementami graficznymi (np. stosowanie apli we fragmentach akapitu lub tzw. *changebars*),
- ułatwienie formatowania dokumentów generowanych z SGML (poprzez uwzględnienie niektórych pojęć SGML),
- integracja obsługi hipertekstu: generowania dowiązań i innych elementów potrzebnych przy tworzeniu dokumentów prezentowanych *on-line*

Potrzeby ulepszeń w wersji 3

- Zostanie ulepszona odporność na błędy, w szczególności zostaną usunięte sztuczne ograniczenia zagnieżdżania poleceń.
- Zostanie wprowadzony ulepszony interaktywny system pomocy.
- Nowe interfejsy zostaną szczegółowo udokumentowane. W dystrybucji systemu znajdzie się obszerny zbiór przykładów.
- Kod nowego systemu będzie modularny i dogłębnie udokumentowany. Pozwoli to łatwo utrzymywać i ulepszać system.

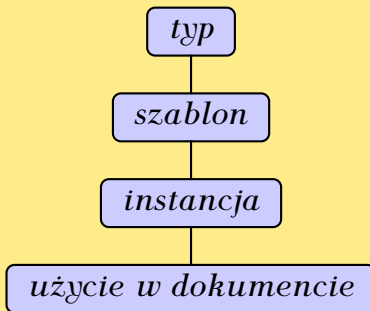
Interfejs projektanta typograficznego

W L^AT_EXu 3 ma pojawić się nowy mechanizm pozwalający na projektowanie formatowania poprzez zestawianie ze sobą sparametryzowanych „klocków” (szablonów). Z założenia takie działanie ma nie wymagać umiejętności programistycznych.

Interfejs projektanta typograficznego

Można się spodziewać, że z czasem pojawi się (graficzne) narzędzie pozwalające zestawiać ze sobą elementy formatujące i określać ich parametry.

Szablony typograficzne (*templates*)



Typ heading

Szablony tego typu służą do formatowania nagłówków (pod)rozdziałów.

Argumenty:

- #1 (boolowski) czy nadać numer?
- #2 tekst nagłówka
- #3 wariant tekstu do żywej paginy (lub \NoValue)
- #4 wariant tekstu do spisu treści (lub \NoValue)

Szablon 2e-like typu heading

Parametry:

heading-counter	nazwa
heading-justification	instancja typu justification
number-format	makro z argumentem
before-skip	klej
after-skip	klej
indent-first	wartość boolowska

Szablon 2e-like-run-in typu heading

Parametry:

- | | |
|-----------------|--------------------|
| heading-counter | nazwa |
| number-format | makro z argumentem |
| before-skip | klej |
| after-space | klej |

Definicja instancji mysection szablonu 2e-like

```
\DeclareInstance{heading}{mysection}{2e-like}
  { heading-counter = section,
    heading-justification = raggedright-nohyph,
    number-format = #1\quad,
    before-skip = 3.5ex plus 1ex minus .2ex,
    after-skip = 2.3ex plus .2ex,
    indent-first = false
  }
```


Użycie w dokumencie:

```
\UseInstance{heading}{mysection}{\BooleanTrue}  
  {0 blabalii fetorycznej}  
  {0 blabalii\dots}{\NoValue}
```

Mechanizm pobierania argumentów poleceń

W \LaTeX u 2_ε polecenie `\newcommand` pozwala definiować polecenia z argumentami wymaganymi i ewentualnie jednym (pierwszym) opcjonalnym. Nie wystarcza to do zdefiniowania niektórych poleceń \LaTeX a.

Ograniczenia te pokonuje nowy mechanizm pobierania argumentów poleceń (eksperymentalny pakiet `xparse`).

```
\DeclareDocumentCommand{\section}{soom}  
  {\UseInstance{heading}{mysection}  
   {#1}{#4}{#2}{#3}}
```

```
\DeclareDocumentCommand{\section}{som}
  {\UseInstance{heading}{mysection}
   {#1}{#3}{#2}{#2}}
```

Oznaczenia rodzajów argumentów:

m	wymagany
o	opcjonalny
<code>O{dfлт.}</code>	opcjonalny z wartością domyślną
s	opcjonalna gwiazdka
c	para w nawiasie, np. (1,23)
<code>C{{x}{y}}</code>	para z wartością domyślną

Brakujący argument opcjonalny bez wartości domyślnej przyjmuje wartość `\NoValue`.

Przy przetwarzaniu argumentów fakt ten można wykryć przy pomocy makra `\IfNoValueTF`:

```
\IfNoValueTF{#1}  
  {Brak parametru}  
  {Parametr ma wartość #1}
```

Wynikiem użycia argumentu oznaczonego literą s, czyli opcjonalnej gwiazdki, jest przekazanie na odpowiedniej pozycji wartości `\BooleanTrue` (gwiazdka obecna) lub `\BooleanFalse` (brak gwiazdki). Do sprawdzania tych przypadków służy następujące makro `\IfBooleanTF`:

```
\IfBooleanTF{#2}  
  {Jest gwiazdka}  
  {Brak gwiazdki}
```

Wywołanie polecenia `\section`

```
\section[0 blabalii\dots][Fetoblabyka]
  {0 blabalii fetorycznej}
```

rozwija się do:

```
\UseInstance{heading}{mysection}{\BooleanFalse}
  {0 blabalii fetorycznej}
  {0 blabalii\dots}{Fetoblabyka}
```


Wywołanie polecenia `\section`

```
\section*[0 blabalii\dots]{0 blabalii fetorycznej}
```

rozwija się do:

```
\UseInstance{heading}{mysection}{\BooleanTrue}  
  {0 blabalii fetorycznej}  
  {0 blabalii\dots}{\NoValue}
```

Nowa procedura wyjścia (*output routine*)

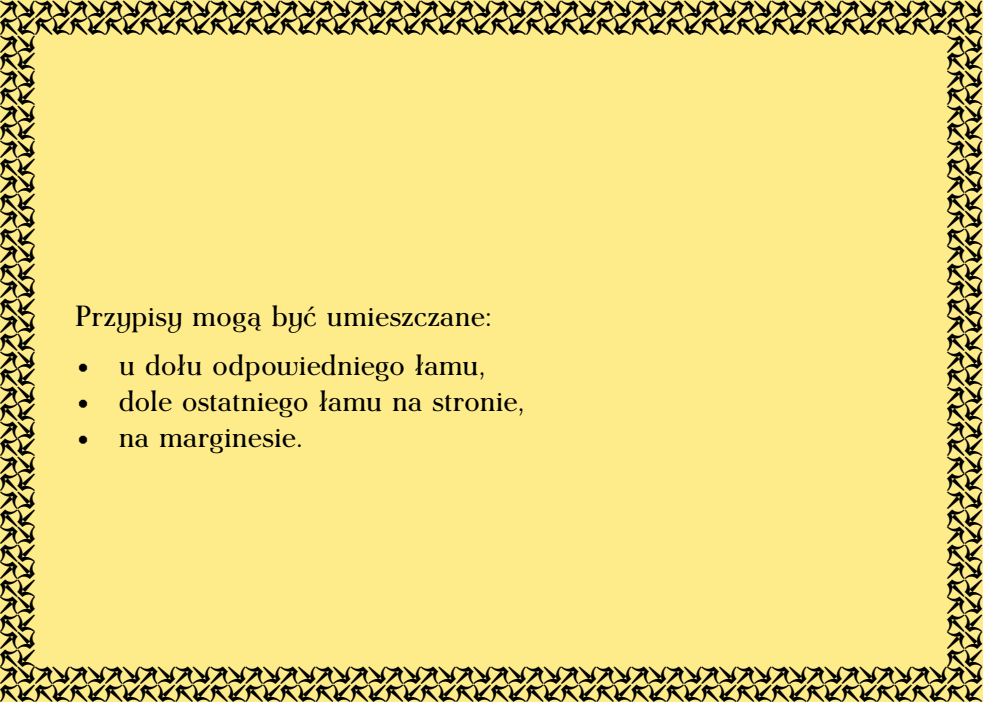
Nowa wersja L^AT_EXowej procedury wyjścia umożliwia wielołamowy skład dokumentów z rozmieszczaniem środowisk ruchomych (*floats*).

Nowy algorytm dopuszcza dowolną liczbę łamów równej szerokości. Łamy na ostatniej kolumnie mają być wyrównywane (jeszcze nie zaimplementowane).

Zmiana liczby łamów może następować jedynie od nowej strony.

Algorytm może operować pojęciem rozwarcia.

Środowiska ruchome umieszczane są w prostokątnych obszarach rozciągających się przez dowolną liczbę łamów, na górze i na dole kolumny.



Przypisy mogą być umieszczane:

- u dołu odpowiedniego łamu,
- dole ostatniego łamu na stronie,
- na marginesie.

W główkach i stopkach stron można odwoływać się do informacji zachowanej poleceniami `\mark` dla każdego z łamów z osobna.

Możliwa zatem jest np. niezależna żywa pagina dla każdego z łamów z osobna.

Podpisy uzyskiwane poleceniem `\caption`, są składane dopiero w momencie umieszczania składu środowiska na kolumnie.

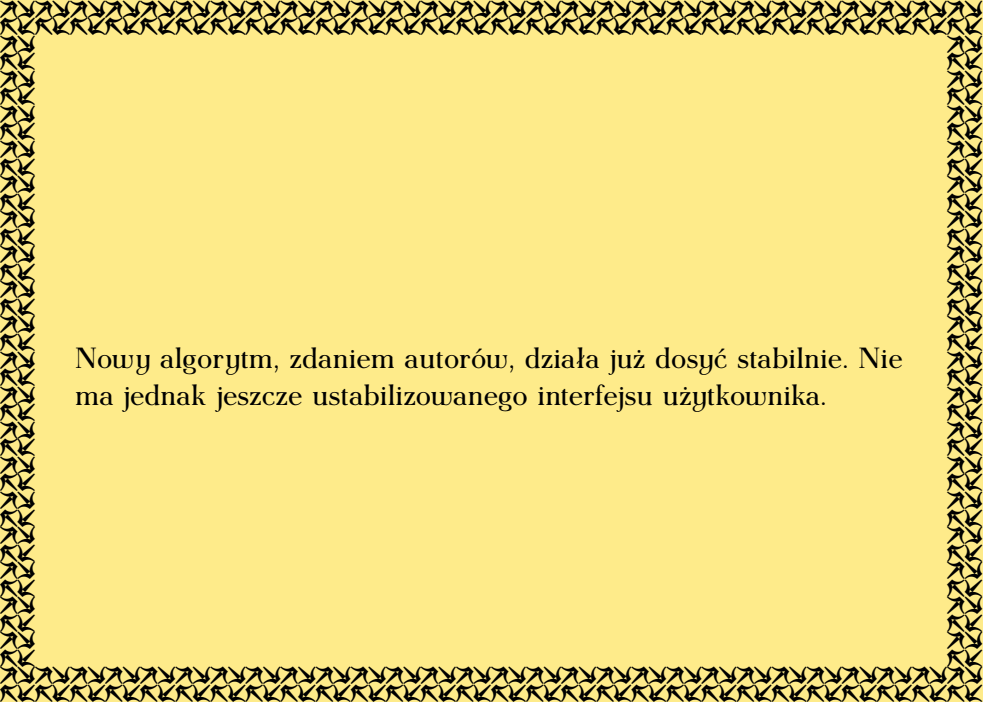
Zatem położenie podpisu względem rysunku jest częścią projektu dokumentu i nie zależy od miejsca umieszczenia polecenia `\caption`.

Środowiska ruchome można rozmieszczać ręcznie.

Algorytm może zapisać swoje decyzje rozmieszczenia środowisk ruchomych w pliku z rozszerzeniem `.fpl`. Plik ten można skopiować z rozszerzeniem `.fpc` i wyedytować zmieniając niektóre lub wszystkie decyzje programu. Miejsca umieszczenia obiektów wskazuje się, podając numery stron i nazwy obszarów.

W nowej procedurze wyjścia uwzględniono potrzebę składu na siatce, czyli zachowania rejestru (pakiet xo-grid).

Głównym narzędziem jest polecenie powodujące „przyciągnięcie” bieżącego wiersza dokumentu do najbliższej linii siatki o zadanym odstępie. Wszystkie polecenia i środowiska generujące zmienne odstępy zostają wzbogacone o wywołanie tego polecenia tak, aby zwykłe akapity tekstu miały linie bazowe na siatce.



Nowy algorytm, zdaniem autorów, działa już dosyć stabilnie. Nie ma jednak jeszcze ustabilizowanego interfejsu użytkownika.

Podsumowanie

Data ukazania się nowej wersji L^AT_EXa pozostaje zagadką.