

# Пакет churchslavonic — верстка церковнославянских текстов в системе L<sup>A</sup>T<sub>E</sub>X

Александр Андреев и Михаил Крутиков  
версия 0.1

3 мая 2016 г.

## Аннотация

Пакет churchslavonic позволяет верстать церковнославянские документы. В пакет включены шаблоны переносов, шрифты и набор необходимых макрокоманд.

## Содержание

<b>1</b>	<b>Как пользоваться пакетом</b>	<b>2</b>
1.1	Опции color, gray и bw . . . . .	2
<b>2</b>	<b>Числа</b>	<b>3</b>
2.1	\cuNum . . . . .	3
<b>3</b>	<b>Даты</b>	<b>4</b>
3.1	\cuDate . . . . .	4
3.2	\cuDefineDateFormat . . . . .	4
3.3	\cuUseDateFormat . . . . .	5
3.4	\cuJulianDate . . . . .	5
3.5	\cuMonthName . . . . .	6
3.6	\cuDayName . . . . .	6
3.7	\cuDayNameAccusative . . . . .	6
3.8	\cuToday . . . . .	6
3.9	\cuJulianToday . . . . .	6
3.10	\cuAsJulian . . . . .	6
3.11	\cuAsGregorian . . . . .	6

<b>4 Киноварь</b>	<b>7</b>
4.1 <code>\cuKinovar</code> . . . . .	7
4.2 <code>\cuKinovarColor</code> . . . . .	7
<b>5 Утилиты</b>	<b>7</b>

## Введение

Церковнославянский язык (ISO 639-2 код `cu`) – древний литературный язык славянских народов, который теперь используется в основном как богослужебный язык в Русской Православной Церкви, других поместных православных церквях, а также в грекокатолических и старообрядческих общинах. Пакет `churchslavonic` позволяет верстать церковнославянские тексты в системе  $\TeX$ . В пакет включены шаблоны переносов, шрифты и набор необходимых макрокоманд.

Данный пакет поддерживает только набор в стандарте Юникод. Тексты, набранные в устаревших кодировках HP и UCS, можно конвертировать в Юникод (точнее – в UTF-8) с помощью отдельных утилит, которые можно найти на сайте Инициативной группы славянской информатики. Для того, чтобы использовать возможности этого пакета Вам понадобится система верстки  $\TeX$ , поддерживающая Юникод, например,  $\XeTeX$  или  $\text{Lua}\TeX$ .

## 1. Как пользоваться пакетом

Добавьте следующие определения в преамбулу вашего документа:

```
\usepackage{polyglossia}
\setmainlanguage{churchslavonic}
\usepackage{churchslavonic}
```

После чего переключайтесь между языками стандартными средствами пакета `polyglossia`.

### 1.1. Опции `color`, `gray` и `bw`

Опции позволяют отключать цвет раскраски текста в командах типа `\cuKinovar`. Полезны для создания версий для печати на устройствах которые не поддерживают цветности.

- `color` эта опция используется по умолчанию и означает что используется нормальный красный цвет.

- `gray` заменяет красный цвет серым — используйте эту опцию если ваше устройство печати не поддерживает цветности, но может качественно передавать градации серого цвета.
- `bw` отключает всякую раскраску — используйте для создания черно-белой версии документа.

Пример:

```
\usepackage[gray]{churchslavonic}
```

## 2. Числа

Способ записи чисел в церковнославянском языке основывается на древнегреческом и в качестве цифр использует буквы. За подробностями отсылаем интересующегося читателя к соответствующей главе [1].

### 2.1. `\cuNum`

Команда печатает число в церковнославянской нотации. Она принимает единственный аргумент. Аргументом может быть текст, командная последовательность, или имя целочисленного регистра. Единственное условие — аргумент должен раскрыться в число.

<code>\cuNum{1}</code>	ⲁ
<code>\cuNum{12}</code>	Ⲃⲓ
<code>\cuNum{123}</code>	ⲠⲒⲒ
<code>\cuNum{1234}</code>	ⲂⲒⲂⲂ
<code>\cuNum{10345}</code>	ⲂⲒⲂⲒϷ
<code>\cuNum{12345}</code>	ⲂⲓⲒⲂⲒϷ
<code>\cuNum{123456}</code>	ⲠⲒⲒⲒϷⲒⲑ
<code>\cuNum{800456}</code>	ⲂⲒⲒⲒϷⲒⲑ
<code>\cuNum{1234567}</code>	ⲂⲒⲒⲒϷⲒⲑϷϷ
<code>\cuNum{1500567}</code>	ⲂⲒⲒⲒϷϷⲒⲑϷϷ
<code>\cuNum{12345678}</code>	ⲂⲓⲒⲂⲒϷⲒⲑⲒⲐ
<code>\cuNum{123456789}</code>	ⲠⲒⲒⲒϷⲒⲑⲒⲑⲒⲐ

### 3. Даты

<code>\cuDate{2016-4-21}</code>	ἡ̅κα ἀπρίλλια λ᾿ἔττα ,κ᾿ῶι
<code>\cuJulianDate{2016-4-21}</code>	ἡ̅ ἀπρίλλια λ᾿ἔττα ,κ᾿ῶι
<code>\cuDate{\cuToday}</code>	ἡ̅ μάϊα λ᾿ἔττα ,κ᾿ῶι

#### 3.1. `\cuDate`

Команда форматирует дату (в соответствии с текущим форматом). Аргумент должен иметь вид YYYY-MM-DD. Результат может выглядеть примерно так: ἡ̅κα ἀπρίλλια λ᾿ἔττα ,κ᾿ῶι.

Отметим что значение даты никоим образом не интерпретируется и не нормализуется. Поэтому можно вызывать макрокоманду и с невозможными датами, например 32-е апреля — такая дата будет отформатирована как 32-е апреля. Так что команду `\cuDate` можно использовать для набора фраз вроде “дата `\cuDate{2016-04-22}` не существует ни в одном календаре”.

Однако, если вы используете свой формат даты и в этом формате задействованы символические переменные `\cuDOW` (день недели) или `\cuYEARAM` (год от сотворения мира), то значения этих переменных будут вычислены исходя из даты — и дата интерпретируется как дата грегорианского календаря. Кроме того, невозможные даты будут нормализованы методом экстраполяции. Например, 32 апреля будет интерпретировано как 2 мая для целей определения дня недели и года от сотворения мира.

Если ваш формат использует день недели (`\cuDOW`) или год от сотворения мира (`\cuYEARAM`) и вы указываете дату юлианского календаря, вы должны вместо `\cuDate` пользоваться макрокомандой `\cuJulianDate` для правильной обработки дня недели и года от сотворения мира.

Лучше всего всегда использовать `\cuDate` для форматирования грегорианских дат и всегда использовать `\cuJulianDate` для форматирования юлианских дат — вне зависимости от значения текущего формата. Таким образом вы можете впоследствии изменить формат даты (например, добавив в него день недели) и результат останется правильным.

#### 3.2. `\cuDefineDateFormat`

Команда определяет формат даты. Она никак не влияет на то как `\cuDate` форматирует свой вывод (для этого используется `\cuUseDateFormat`). Пример:

```

\cuDefineDateFormat{long}{%
  \cuDayName{\cuDOW},
  \cuNum{\cuDAY}_гв~%
  \cuMonthName{\cuMONTH}~%
  лѣта ѿ сотворѣніа міра~%
  \cuNum{\cuYEARAM}%
}

```

определяет новый формат с именем `long`. Если мы напечатаем дату этим форматом, то получим: `пачбкя, кв_гв апріліа лѣта ѿ сотворѣніа міра ,зфкџ`.

При определении формата можно пользоваться следующими символическими именами:

- `\cuYEAR` — год (число, например 2016)
- `\cuYEARAM`<sup>1</sup> — год от сотворения мира, Anno Mundi (число, например 7525).
- `\cuMONTH` — месяц (число от 1 до 12)
- `\cuDAY` — день месяца
- `\cuDOW`<sup>1</sup> — день недели (число от 0 to 6, где 0 означает “воскресение”)
- `\cuINDICTION` — индикт (число от 1 до 15)

### 3.3. `\cuUseDateFormat`

Макро устанавливает имя текущего формата даты. Этот формат будет использоваться при последующих вызовах макрокоманд `\cuDate` и `\cuJulianDate`.

### 3.4. `\cuJulianDate`

Форматирует дату в соответствии с текущим форматом подобно команде `\cuDate`. Отличие заключается в том, что аргумент интерпретируется как дата юлианского а не грегорианского календаря. Это имеет значение только если текущий формат даты использует символические имена `\cuDOW` (день недели) или `\cuYEARAM` (год от сотворения мира).

<sup>1</sup>Если ваш формат пользуется этим значением, вы должны форматировать дату правильной макрокомандой: `\cuDate` если вы передаете дату грегорианского календаря, или `\cuJulianDate` если используется дата юлианского календаря

### 3.5. `\cuMonthName`

Команда раскрывает числовой аргумент в текстовое представление месяца. Команда обычно используется при задании формата даты. Например, формат с именем `default` определен так:

```
\cuDefineDateFormat{default}{%
  \cuNum{\cuDAY}~\cuMonthName{\cuMONTH}%
  ~л'б'га~\cuNum{\cuYEAR}%
}%
```

### 3.6. `\cuDayName`

Команда раскрывает числовой аргумент в текстовое представление дня недели в именительном падеже.

### 3.7. `\cuDayNameAccusative`

Команда раскрывает числовой аргумент в текстовое представление дня недели в винительном падеже.

### 3.8. `\cuToday`

Эта макрокоманда раскрывается в текущую дату в виде триплета `YYYY-MM-DD`. Используется грегорианский календарь.

### 3.9. `\cuJulianToday`

Эта макрокоманда раскрывается в текущую дату в виде триплета `YYYY-MM-DD`. Используется юлианский календарь.

Команда просто раскрывается в: `\cuAsJulian{\cuToday}`.

### 3.10. `\cuAsJulian`

Преобразует грегорианскую дату в юлианскую. Дата задается и выводится как числовой триплет `YYYY-MM-DD`.

Команда полезна когда одну и ту же дату надо представить и в грегорианском и в юлианском календаре.

### 3.11. `\cuAsGregorian`

Преобразует юлианскую дату в грегорианскую. Дата задается и выводится как числовой триплет `YYYY-MM-DD`.

## 4. Киноварь

Печатные и рукописные книги на церковнославянском часто пользуются красным цветом для выделения структурных элементов текста. Обычно заголовки, замечания и пометки на полях печатаются красным цветом. Кроме того, первая буква каждого параграфа тоже выделяется красным цветом.

### 4.1. `\cuKinovar`

Команда печатает свой аргумент красным цветом. Например, если мы явно зададим текст аргумента, то получим ожидаемый результат:

<code>\cuKinovar{лѣкѣ:}</code> гдѣ помѣлѣи.	лѣкѣ: гдѣ помѣлѣи.
---------------------------------------------	--------------------

Если же используется механизм Т<sub>Э</sub>X неявного задания аргумента, то команда будет применена к первой букве последующего текста. При этом команда “соберет” все необходимые выносные символы, так что все акценты первой буквы будут корректно раскрашены! Поэтому очень удобно использовать эту команду с неявным заданием аргумента для раскраски первой буквы параграфа.<sup>1</sup>

<code>\cuKinovar</code> Поимѣ гдѣи пѣснь нѣвѣю	Поимѣ гдѣи пѣснь нѣвѣю
<code>\cuKinovar</code> Ѡакѣ тѣча на трѣскоцѣ	Ѡакѣ тѣча на трѣскоцѣ

### 4.2. `\cuKinovarColor`

Команда переключает текущий цвет на красный. Должна использоваться внутри группы, которая ограничит ее действие (если вы не хотите чтобы весь последующий текст был напечатан красным цветом).

## 5. Утилиты

Механизм, используемый командой `\cuKinovar` для сбора выносных символов (при неявном задании аргумента), может оказаться полезен в некоторых

<sup>1</sup>Вы также можете поэкспериментировать со встроенной командой Т<sub>Э</sub>X `\everypar` с целью полностью автоматизировать раскраску первой буквы параграфа. Однако успех сильно зависит от того какой класс системы Л<sub>Т</sub><sub>Э</sub>X используется и от того какие пакеты подгружаются. Мы не нашли надежного способа заставить `\everypar` работать и поэтому пакет `churchslavonic` не предлагает никаких макрокоманд для дальнейшей автоматизации этого процесса. На практике, часто проще и надежнее воспользоваться функцией текстового редактора Найти/Заменить для вставки макрокоманды `\cuKinovar` перед каждым параграфом.

других случаях. Один из примеров — это “буквицы” в начале разделов. Для набора буквицы отлично работает стандартный пакет `lettrine`. Единственное неудобство — это необходимость задавать явно первую букву вместе во всеми ее выносными символами. Естественно желание упростить и автоматизировать эту работу задействовав тот же механизм что и в `\cuKinovar`.

Вот как можно добиться этого:

```
\def\cu@lettrine{\lettrine[lines=3,findent=0pt,nindent=0pt]}
\def\cuLettrine{\cu@tokenizeletter\cu@lettrine}
\renewcommand{\LettrineFontHook}{\cuKinovarColor}
```

Поместите это определение команды `\cuLettrine` в преамбулу вашего документа (и не забудьте окружить его командами `\makeatletter` и `\makeatother`). После этого можно набирать буквицы так:

```
\cuLettrine Њже дѣла сила въ немоци совершѣется...
```

**Њ**же дѣла сила въ немоци совершѣется, ѿкоже писано єсть, и вѣрѣмъ: въ немоци же не чѣлесè тóчнѹ, но оубо и слова, и премѣдрости на ѡзыцѣ лежѣца. И сè ѡвѣ ѿ многихъ оубо иныхъ, паче же ѿ ѡже ѡ великомъ вѣсломѣ, и вратѣ хрѣтѣ, благодѣнїю зрѣмѣмъ.

## Список литературы

- [1] Aleksandr Andreev, Yuri Shardt, and Nikita Simmons. *Church Slavonic Typography in Unicode*, Unicode Technical Note 41, <http://www.unicode.org/notes/tn41/>