

LENGUAJE DE PROGRAMACIÓN ORIENTADO A OBJETOS PARA LA COMPOSICIÓN TIPOGRÁFICA

David Lilue

Universidad Simón Bolívar

2016

```
use inputenc { encode: utf8 }

foo :: Article {
  font: 10pt
  paper: letter
}

titulo :: Text {
  align: center
  size: LARGE
  text: '\vspace{4em}\OhTeX\vspace{4em}'
}

texto :: Text {
  text: '\OhTeX{} es un lenguaje de programación orientado a
  objetos para la composición tipográfica, haciendo uso de
  \LaTeX{} como base para ello. Ofrece la facilidad de manejar
  elementos tipográficos de una manera abstracta y expresiva,
  facilitando la comprensión y aprendizaje del mismo.'
}

foo << titulo << texto

foo >> 'foo'
```

O_HT_EX

O_HT_EX es un lenguaje de programación orientado a objetos para la composición tipográfica, haciendo uso de L^AT_EX como base para ello. Ofrece la facilidad de manejar elementos tipográficos de una manera abstracta y expresiva, facilitando la comprensión y aprendizaje del mismo.

```
use inputenc { encode: utf8 }

foo :: Article {
  font: 10pt
  paper: letter
}

titulo :: Text {
  align: center
  size: LARGE
  text: '\vspace{4em}\OhTeX\vspace{4em}'
}

texto :: Text {
  text: '\OhTeX{} es un lenguaje de programación orientado a
  objetos para la composición tipográfica, haciendo uso de
  \LaTeX{} como base para ello. Ofrece la facilidad de manejar
  elementos tipográficos de una manera abstracta y expresiva,
  facilitando la comprensión y aprendizaje del mismo.'
}

foo << titulo << texto

foo >> 'foo'
```

» Paquetes

```
use inputenc { encode: utf8 }
```

» Paquetes

```
use inputenc { encode: utf8 }
```

- Uso de paquetes

» Paquetes

```
use inputenc { encode: utf8 }
```

- Uso de paquetes
- Opciones de paquetes

» Paquetes

```
use inputenc { encode: utf8 }
```

- Uso de paquetes
- Opciones de paquetes
- Incorporación de inicialización por nombre

» Paquetes

```
use inputenc { encode: utf8 }
```

- Uso de paquetes
- Opciones de paquetes
- Incorporación de inicialización por nombre

```
use hyperref, graphicx, listings
```

» Paquetes

```
use inputenc { encode: utf8 }
```

- Uso de paquetes
- Opciones de paquetes
- Incorporación de inicialización por nombre

```
use hyperref, graphicx, listings
```

Similar a como se hace en \LaTeX

» Contrucción de instancia

Instancias de documentos

```
foo :: Article {  
    font: 10pt  
    paper: letter  
}
```

» Contrucción de instancia

Instancias de documentos

```
foo :: Article {  
    font: 10pt  
    paper: letter  
}
```

- Clases de documentos

» Contrucción de instancia

Instancias de documentos

```
foo :: Article {
  font: 10pt
  paper: letter
}
```

- Clases de documentos
- Simbolos '::' y ':'

» Contrucción de instancia

Instancias de documentos

```
foo :: Article {
  font: 10pt
  paper: letter
}
```

- Clases de documentos
- Simbolos '::' y ':'
- Atributo-Valor

» Contrucción de instancia

Instancias de documentos

```
foo :: Article {
  font: 10pt
  paper: letter
}
```

- Clases de documentos
- Simbolos '::' y ':'
- Atributo-Valor
- Unidades y variables definidas

» Contrucción de instancia

Instancias de documentos

```
foo :: Article {
  font: 10pt
  paper: letter
}
```

- Clases de documentos
- Simbolos '::' y ':'
- Atributo-Valor
- Unidades y variables definidas

El uso del simbolo ':' fue influenciado por CSS

» Formato de texto

Instancias de texto

```
titulo :: Text {  
    align: center  
    size: LARGE  
    text: '\vspace{4em}\OHTEX\vspace{4em}'  
}  
  
texto :: Text {  
    text: '\OHTEX{} es un lenguaje de programación orientado a  
    objetos para la composición tipográfica, haciendo uso de  
    \LaTeX{} como base para ello. Ofrece la facilidad de manejar  
    elementos tipográficos de una manera abstracta y expresiva,  
    facilitando la comprensión y aprendizaje del mismo.'  
}
```

» Formato de texto

Instancias de texto

```
titulo :: Text {  
    align: center  
    size: LARGE  
    text: '\vspace{4em}\OhTeX\vspace{4em}'  
}  
  
texto :: Text {  
    text: '\OhTeX{} es un lenguaje de programación orientado a  
    objetos para la composición tipográfica, haciendo uso de  
    \LaTeX{} como base para ello. Ofrece la facilidad de manejar  
    elementos tipográficos de una manera abstracta y expresiva,  
    facilitando la comprensión y aprendizaje del mismo.'  
}
```

- Formato de texto como un atributo

» Formato de texto

Instancias de texto

```
titulo :: Text {
    align: center
    size: LARGE
    text: '\vspace{4em}\OhTeX\vspace{4em}'
}

texto :: Text {
    text: '\OhTeX{} es un lenguaje de programación orientado a
    objetos para la composición tipográfica, haciendo uso de
    \LaTeX{} como base para ello. Ofrece la facilidad de manejar
    elementos tipográficos de una manera abstracta y expresiva,
    facilitando la comprensión y aprendizaje del mismo.'
```

- Formato de texto como un atributo
- Usa de *verbatim* L_AT_EX en O_HT_EX

» Formato de texto

Instancias de texto

```
titulo :: Text {  
    align: center  
    size: LARGE  
    text: '\vspace{4em}\OhTEX\vspace{4em}'  
}  
  
texto :: Text {  
    text: '\OhTEX{} es un lenguaje de programación orientado a  
    objetos para la composición tipográfica, haciendo uso de  
    \LaTEX{} como base para ello. Ofrece la facilidad de manejar  
    elementos tipográficos de una manera abstracta y expresiva,  
    facilitando la comprensión y aprendizaje del mismo.'}
```

- Formato de texto como un atributo
- Usa de *verbatim* L_AT_EX en O_HT_EX

Integración, manteniendo similitudes

» Incorporación de elementos tipográficos y salida

```
foo << titulo << texto
```

```
foo >> 'foo'
```

» Incorporación de elementos tipográficos y salida

```
foo << titulo << texto
```

```
foo >> 'foo'
```

- Operador '<<' y '>>'

» Incorporación de elementos tipográficos y salida

```
foo << titulo << texto
```

```
foo >> 'foo'
```

- Operador '<<' y '>>'
- Estructuración de un documento

» Incorporación de elementos tipográficos y salida

```
foo << titulo << texto
```

```
foo >> 'foo'
```

- Operador '<<' y '>>'
- Estructuración de un documento
- Composición referencial

» Incorporación de elementos tipográficos y salida

```
foo << titulo << texto
```

```
foo >> 'foo'
```

- Operador '<<' y '>>'
- Estructuración de un documento
- Composición referencial

Crear un elemento tipográfico no lo incluye en un documento

» Proceso de traducción

1 O_HT_EX

2 *Ruby*

3 L^AT_EX

4 PDF

» Proceso de traducción

L^AT_EX generado1 O_HTEX2 *Ruby*3 L^AT_EX

4 PDF

```
\documentclass[10pt,letterpaper]{article}
\usepackage[utf8]{inputenc}

\begin{document}
\begin{center}
{\LARGE \vspace{4em}\OhTeX\vspace{4em}}
\end{center}

{\normalsize \OhTeX{}} es un lenguaje de programación
orientado a objetos para la composición
tipográfica, haciendo uso de \LATEX{} como base
para ello. Ofrece la facilidad de manejar elementos
tipográficos de una manera abstracta y expresiva,
facilitando la comprensión y aprendizaje del mismo.
}
\end{document}
```

» Proceso de traducción

L^AT_EX generado

- 1 OH_HTEX
- 2 *Ruby*
- 3 L^AT_EX
- 4 PDF

```
\documentclass[10pt,letterpaper]{article}
\usepackage[utf8]{inputenc}

\begin{document}
\begin{center}
{\LARGE \vspace{4em}\0hTeX\vspace{4em}}
\end{center}

{\normalsize \0hTeX{} es un lenguaje de programación
orientado a objetos para la composición
tipográfica, haciendo uso de \LaTeX{} como base
para ello. Ofrece la facilidad de manejar elementos
tipográficos de una manera abstracta y expresiva,
facilitando la comprensión y aprendizaje del mismo.
}
\end{document}
```

Puede que sean “menos” caracteres. Eso no significa mayor expresividad

```
use inputenc { encode: utf8 }

foo :: Article {
  font: 10pt
  paper: letter
}

titulo :: Text {
  align: center
  size: LARGE
  text: '\vspace{4em}\OhTeX\vspace{4em}'
}

texto :: Text {
  text: '\OhTeX{} es un lenguaje de programación orientado a
objetos para la composición tipográfica, haciendo uso de
\LaTeX{} como base para ello. Ofrece la facilidad de manejar
elementos tipográficos de una manera abstracta y expresiva,
facilitando la comprensión y aprendizaje del mismo.'
}

foo << titulo << texto

foo >> 'foo'
```

- Origen
- Influencias y Allegados
- Implementación
- Manejo de objetos tipográficos
- Estructuración de un documento
- Conclusión

- Bi Sheng. China. Mediados del siglo 11
 - *Types* movibles de cerámica

- Bi Sheng. China. Mediados del siglo 11
 - *Types* movibles de cerámica
- Corea. Siglo 13-14
 - *Types* movibles metálicos

- Bi Sheng. China. Mediados del siglo 11
 - *Types* movibles de cerámica
- Corea. Siglo 13-14
 - *Types* movibles metálicos
- Johannes Gutenberg. Magnuncia, Alemania. Mediados del siglo 15
 - Prensa

- Bi Sheng. China. Mediados del siglo 11
 - *Types* movibles de cerámica
- Corea. Siglo 13-14
 - *Types* movibles metálicos
- Johannes Gutenberg. Magnuncia, Alemania. Mediados del siglo 15
 - Prensa
- Ottmar Mergenthaler. Finales del siglo 19
 - Maquina Linotapia

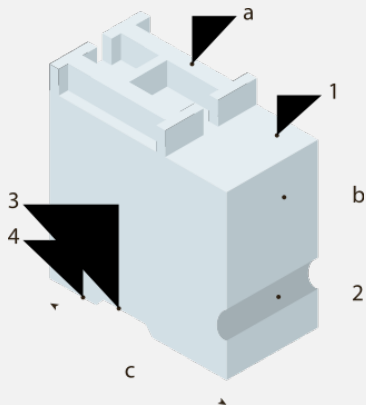
- Bi Sheng. China. Mediados del siglo 11
 - *Types* movibles de cerámica
- Corea. Siglo 13-14
 - *Types* movibles metálicos
- Johannes Gutenberg. Magnuncia, Alemania. Mediados del siglo 15
 - Prensa
- Ottmar Mergenthaler. Finales del siglo 19
 - Maquina Linotapia
- Foto-composición tipográfica. 1970-1980.

- Bi Sheng. China. Mediados del siglo 11
 - *Types* movibles de cerámica
- Corea. Siglo 13-14
 - *Types* movibles metálicos
- Johannes Gutenberg. Magnuncia, Alemania. Mediados del siglo 15
 - Prensa
- Ottmar Mergenthaler. Finales del siglo 19
 - Maquina Linotapia
- Foto-composición tipográfica. 1970-1980.
- Composición tipográfica digital.
 - T_EX. Donald Knuth. 1978
 - L^AT_EX. Leslie Lamport. 1985

- Bi Sheng. China. Mediados del siglo 11
 - *Types* movibles de cerámica
- Corea. Siglo 13-14
 - *Types* movibles metálicos
- Johannes Gutenberg. Magnuncia, Alemania. Mediados del siglo 15
 - Prensa
- Ottmar Mergenthaler. Finales del siglo 19
 - Maquina Linotapia
- Foto-composición tipográfica. 1970-1980.
- Composición tipográfica digital.
 - T_EX. Donald Knuth. 1978
 - L^AT_EX. Leslie Lamport. 1985

El libro más viejo impreso con *types* movibles es el *Jikji*

Origen



- Origen
- Influencias y Allegados
- Implementación
- Manejo de objetos tipográficos
- Estructuración de un documento
- Conclusión

Influencias y Allegados

- Herramientas de composición tipográfica
 - Lenguajes de marcado
 - T_EX/L^AT_EX

- Herramientas de composición tipográfica
 - Lenguajes de marcado
 - T_EX/L^AT_EX
- Lenguaje de programación orientados a objetos
 - Ruby

- Herramientas de composición tipográfica
 - Lenguajes de marcado
 - T_EX/L^AT_EX
- Lenguaje de programación orientados a objetos
 - Ruby
- Lenguaje de estilo y formato
 - CSS

- Herramientas de composición tipográfica
 - Lenguajes de marcado
 - T_EX/L^AT_EX
- Lenguaje de programación orientados a objetos
 - Ruby
- Lenguaje de estilo y formato
 - CSS
- Notación legible por humano
 - JSON

- Herramientas de composición tipográfica
 - Lenguajes de marcado
 - T_EX/L^AT_EX
- Lenguaje de programación orientados a objetos
 - Ruby
- Lenguaje de estilo y formato
 - CSS
- Notación legible por humano
 - JSON
- Hermano?
 - Curl

Influencias y Allegados

- Aplicaciones integradas con \LaTeX
 - LyX, TeXmacs (no \TeX)
 - Texmaker
 - Overleaf, ShareLaTeX

- Aplicaciones integradas con \LaTeX
 - LyX, TeXmacs (no \TeX)
 - Texmaker
 - Overleaf, ShareLaTeX
- \O\TeX Lenguaje de dominio específico
 - Que tan específico?
 - Ventajas y Desventajas

- Aplicaciones integradas con \LaTeX
 - LyX, TeXmacs (no \TeX)
 - Texmaker
 - Overleaf, ShareLaTeX
- \OHTeX Lenguaje de dominio específico
 - Que tan específico?
 - Ventajas y Desventajas
- \OHTeX Lenguaje embebido
 - Ruby
 - \LaTeX
 - Tiene que ser embebido?

Influencias y Allegados

- Aplicaciones integradas con \LaTeX
 - LyX, TeXmacs (no \TeX)
 - Texmaker
 - Overleaf, ShareLaTeX
- \OHTeX Lenguaje de dominio específico
 - Que tan específico?
 - Ventajas y Desventajas
- \OHTeX Lenguaje embebido
 - Ruby
 - \LaTeX
 - Tiene que ser embebido?

WYSIWYM: *what-you-see-is-you-mean*. WYSIWYG: *what-you-see-is-you-get*

- Origen
- Influencias y Allegados
- Implementación
- Manejo de objetos tipográficos
- Estructuración de un documento
- Conclusión

- *Lexer y Parser*
 - Por qué Ruby?

- *Lexer y Parser*
 - Por qué Ruby?
- De librería a DSL

DSL: *Domain specific language*

- *Lexer y Parser*
 - Por qué Ruby?
- De librería a DSL
- Consecuencias de usar dos intermediarios
 - Cuales son esenciales?
 - Por qué dejar archivos intermedios?

DSL: *Domain specific language*

- *Lexer y Parser*
 - Por qué Ruby?
- De librería a DSL
- Consecuencias de usar dos intermediarios
 - Cuales son esenciales?
 - Por qué dejar archivos intermedios?
- Consideración de opciones

DSL: *Domain specific language*

- Origen
- Influencias y Allegados
- Implementación
- Manejo de objetos tipográficos
- Estructuración de un documento
- Conclusión

Manejo de objetos tipográficos

» Estructuras de control

```
tabla1 :: Table {  
    header : ['Número', 'Cuadrado']  
}  
  
valores = [1,2,3,4,5,6,7,8,9]  
  
for i in values {  
    tabla1.add_row([i, i*i])  
}
```

Manejo de objetos tipográficos

» Estructuras de control

```
nombre :: Text {
    size  : Large
    align : left
}

.. Suponiendo que tenemos un objeto membrete y un texto
foo << membrete << nombre << lipsum1

nombres = [
    'Sr. Doe',
    'Sra. Jane',
    'Dr. John',
    'Prof. Jenny'
]

for r in nombres {
    nombre.text = r
    foo >> r
}
```

Manejo de objetos tipográficos

» Estructuras de control

```
nombre :: Text {
    size  : Large
    align : left
}

.. Suponiendo que tenemos un objeto membrete y un texto
foo << membrete << nombre << lipsum1

nombres = [
    'Sr. Doe',
    'Sra. Jane',
    'Dr. John',
    'Prof. Jenny'
]

for r in nombres {
    nombre.text = r
    foo >> r
}
```

Generar documentos similares a través de estructuras de control

Manejo de objetos tipográficos

» Referencia a un mismo objeto

```
..{ Suponiendo que tenemos un objeto 'encabezado' usado  
    comunmente en documentos administrativos y/o repetitivos }..  
  
carta :: Letter  
acta  :: Article  
memo  :: Memo  
  
carta << encabezado << nombre1 << texto1  
  
acta  << encabezado << contenido  
  
memo  << encabezado << titulo << texto2  
  
...
```

Manejo de objetos tipográficos

» Referencia a un mismo objeto

```
..{ Suponiendo que tenemos un objeto 'encabezado' usado  
    comunmente en documentos administrativos y/o repetitivos }..  
  
carta :: Letter  
acta  :: Article  
memo  :: Memo  
  
carta << encabezado << nombre1 << texto1  
  
acta  << encabezado << contenido  
  
memo  << encabezado << titulo << texto2  
  
...
```

El uso de un identificador para un objeto tipográfico evita reescribirlo

- Origen
- Influencias y Allegados
- Implementación
- Manejo de objetos tipográficos
- Estructuración de un documento
- Conclusión

Estructuración de un documento

- \LaTeX . Secuencia de elementos tipográficos

```
\documentclass[10pt,letterpaper]{book}
\usepackage[spanish]{babel}

\begin{document}
% elementos tipográficos

\section{Sección 1}

\begin{itemize}
% Items
\end{itemize}

\section{Sección 2}

% texto lipsum

% elementos tipográficos
\end{document}
```

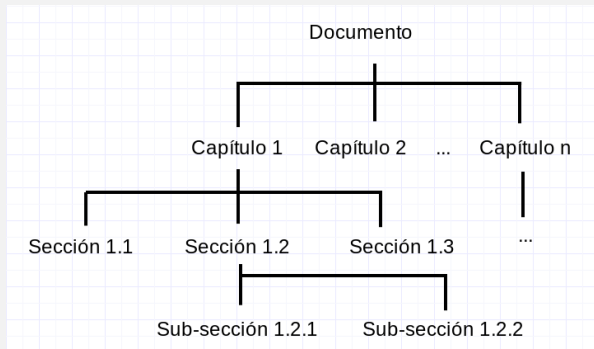
Estructuración de un documento

- O^HT_EX Árbol de objetos tipográficos

```
main :: Book {  
    font : 10pt  
    paper : letter  
}  
  
seccion1 :: Section { name : 'Sección 1' }  
seccion2 :: Section { name : 'Sección 2' }  
  
lista :: List {  
    items : [...]  
}  
  
seccion1 << lista  
  
texto :: Text {  
    text : .. lipsum  
}  
  
seccion2 << texto  
  
main << seccion1 << seccion2
```

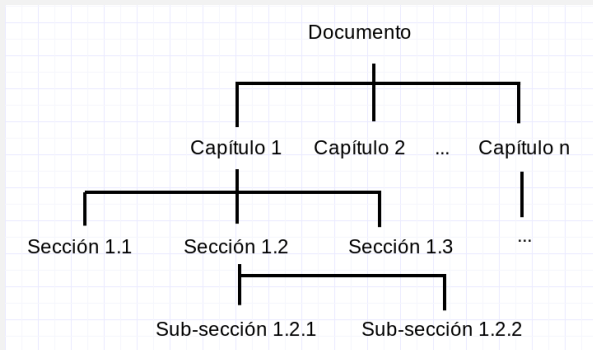

Estructuración de un documento

- \LaTeX Árbol de objetos tipográficos



Estructuración de un documento

- \LaTeX Árbol de objetos tipográficos



La estructuración se asemeja a un tabla de contenido

- Origen
- Influencias y Allegados
- Implementación
- Manejo de objetos tipográficos
- Estructuración de un documento
- Conclusión

Conclusión

- Expresividad
- Dificultad de aprendizaje
- Usabilidad
- Lenguajes anfitriones
- Implementación