NTS: Programming Languages and Paradigms

Jiří Zlatuška
A new, more powerful successor to TEx-the-typesetter.
A new, more powerful successor to TEX-the-typesetter.

An effort to re-program TEX-the-program.
A procedural language based on a formal syntax and well-defined semantics.

Objectives

Pascal
A procedural language based on a formal syntax and well-defined semantics.

Structured programming as a methodology.

Objectives

Pascal
A procedural language based on a formal syntax and well-defined semantics.

Structured programming as a methodology.

Building large programs from manageable smaller components.

Objectives

Pascal
A procedural language based on a formal syntax and well-defined semantics.

Structured programming as a methodology.

Building large programs from manageable smaller components.

Abstract data structures instead of just the data types provided by architectures.
A procedural language based on a formal syntax and well-defined semantics.

Structured programming as a methodology.

Building large programs from manageable smaller components.

Abstract data structures instead of just the data types provided by architectures.

Portability to different platforms.
Introducing the world to literate programming.

Objectives

Pascal

Knuth’s choices
Introducing the world to literate programming.

Extending Pascal by pre-processing the source code.

Objectives

Pascal

Knuth’s choices
Introducing the world to literate programming.

Extending Pascal by preprocessing the source code.

Tangling a WEB and weaving the documentation.

Objectives

Pascal

Knuth’s choices
Introducing the world to literate programming.

Extending Pascal by pre-processing the source code.

Tangling a WEB and weaving the documentation.

Arithmetic calculations and memory management built in to guarantee portability.

Objectives

Pascal

Knuth’s choices
Extending \TeX{} is not that easy.

Objectives

Pascal

Knuth’s choices

20 years later
Extending \TeX{} is not that easy.

Successors of Pascal didn’t make it.
Extending \TeX{} is not that easy.

Successors of Pascal didn’t make it.

Even Knuth has moved to C.
Extending \TeX{} is not that easy.

Successors of Pascal didn’t make it.

Even Knuth has moved to C.

Hardware is no longer a constraint.

Objectives

Pascal

Knuth’s choices

20 years later
Extending TÉX is not that easy.

Successors of Pascal didn’t make it.

Even Knuth has moved to C.

Hardware is no longer a constraint.

TÉX data structures have become time bombs.

Objectives

Pascal

Knuth’s choices

20 years later
Extending \TeX{} is not that easy.

Successors of Pascal didn’t make it.

Even Knuth has moved to C.

Hardware is no longer a constraint.

\TeX{} data structures have become time bombs.

\TeX{}’s monolithic, interwoven character has become a burden.

Objectives

Pascal

Knuth’s choices

20 years later
Extending \TeX is not that easy.

Successors of Pascal didn’t make it.

Even Knuth has moved to C.

Hardware is no longer a constraint.

\TeX data structures have become time bombs.

\TeX’s monolithic, interwoven character has become a burden.

We need to re-create \TeX using modern methodologies and technologies.
Removing all constraints and most complexity.

Objectives

Pascal

Knuth’s choices

20 years later

Options
Removing all constraints and most complexity.

Adding modularity and clear interfaces.

Objectives
Pascal
Knuth’s choices
20 years later
Options
Removing all constraints and most complexity.

Adding modularity and clear interfaces.

Functional programming would suit \TeX's basic manipulations: Lisp?

Objectives

Pascal

Knuth's choices

20 years later

Options
Removing all constraints and most complexity.

Adding modularity and clear interfaces.

Functional programming would suit TeX’s basic manipulations: Lisp?

Logic programming would help us specify and solve typographic problems: Prolog?

Objectives

Pascal

Knuth’s choices

20 years later

Options
Removing all constraints and most complexity.

Adding modularity and clear interfaces.

Functional programming would suit \TeX’s basic manipulations: Lisp?

Logic programming would help us specify and solve typographic problems: Prolog?

Procedural programming along the lines \TeX is coded now: C and C++?

Objectives
Pascal
Knuth’s choices
20 years later
Options
Removing all constraints and most complexity.

Adding modularity and clear interfaces.

Functional programming would suit \TeX’s basic manipulations: Lisp?

Logic programming would help us specify and solve typographic problems: Prolog?

Procedural programming along the lines \TeX is coded now: C and C++?

The portable compromise, procedural programming while using objects: Java!

Objectives

Options

Pascal

Knuth’s choices

20 years later
Provide a high level of structure using objects and methods.

Objectives
Pascal
Knuth’s choices
20 years later
Options
NTS in Java
Provide a high level of structure using objects and methods.

Anchor NTS in the World Wide Web.

Objectives
Pascal
Knuth’s choices
20 years later
Options
NTS in Java
Provide a high level of structure using objects and methods.

Anchor NTS in the World Wide Web.

Support as many platforms as possible.

Objectives

Pascal

Knuth’s choices

20 years later

Options

NTS in Java
Provide a high level of structure using objects and methods.

Anchor NTS in the World Wide Web.

Support as many platforms as possible.

Use standardized interfaces.

Objectives

Pascal

Knuth’s choices

20 years later

Options

NTS in Java
Objectives

Pascal

Knuth’s choices

20 years later

Options

NTS in Java

Provide a high level of structure using objects and methods.

Anchor NTS in the World Wide Web.

Support as many platforms as possible.

Use standardized interfaces.

Move from WEB to WWWeb.