The ltx3obj package
An Object-Oriented System for LaTeX3*

Sean Allred†

Released 2013/08/15

This is not an official LaTeX3 package.

One of the major struggles facing the contributing developers of LaTeX packages is the *absolutely foreign* language syntax and behaviour. I can't help most of that, but LaTeX3 will hopefully be making it a fair bit less mind-bendy. Still, a language constructed entirely around the idea of a *control sequence* is foreign to most developers (if not all of those who are not in-the-know). Thus, I present ltx3obj, a package that aspires to implement object-orientism in LaTeX both at the programming level.

1 Introduction

I realize many aspiring LaTeX package developers may not be 'in-the-know' about what object-orientism really is. Thus, I will attempt to explain it concisely for the attentive reader, and defer you to many online resources for deeper explanations.

Object-orientism is that paradigm in computer programming and systems design where every element of the program is perceived to have state and function. It is a system by which you can look at things like l3seq and create *new* data structures with associated abilities. It is a means by which you can create and maintain a `circle` object, give it properties, give it abilities, and give it uses.[1]

2 Document-Level Use

2.1 Data Structures

We will work through this package as most early languages worked through their own time. Through TeX we have data (via macros) and through LaTeX3 we have a certain set of *data structures*.[2] Wouldn't it be nice to have your *own* data structure?

*This file describes v0, last revised August 15th, 2013.

†E-mail: seallred@smcm.edu

[1] That is, a `circle` could be a property of another class.

[2] Briefly, a *data structure* is what we call a collection of different pieces of data that have cohesive meaning or value when grouped together and taken as a whole.

\NewDataStructureSpecification

Updated: 2013/08/16

`\NewDataStructureSpecification [⟨superstructure name⟩] {⟨structure name⟩} {⟨property specification⟩}`

Declares a new data structure. ⟨*superstructure name*⟩ is another *defined* data structure name (as was given in its own ⟨*structure name*⟩).

This creates a l3prop property list of the form `\g_@@_ds_⟨class name⟩__properties_prop` to hold something like the following:

```
               color = default,
  some_base_property = values,
some_object_property = here
```

Note that this continues the LaTeX3 trend of 'it all depends on interpretation.' There is only one real type for any given property, and that's the token list. It is the methods of your object which will do the interpreting.

\NewClass

Updated: 2013/08/15

`\NewClass [⟨superclass name⟩] {⟨data structure name⟩} {⟨class name⟩} {⟨functions⟩}`

```
\NewClass { shape } { Shape } {
  print:N = {
    The object is
    ##1 % is always the object
   },

  print_qualities:N = {
    % \shape_get/set family from data structure
    The object has size \shape_get_size:N { ##1 }
   }
}
```

\NewObject

Updated: 2013/08/11

`\NewObject {⟨class name⟩} {⟨handler⟩} [⟨property specification⟩]`

Creates an object of type ⟨*class name*⟩ with the handler ⟨*handler*⟩ and optionally its properties.

\SetObject

Updated: 2013/08/11

`\SetObjectProperties {⟨handler⟩} {⟨property specification⟩}`

Sets specific properties for ⟨*handler*⟩. It is a type error if those properties do not exist.

3 propbox implementation

```
⟨*initex | package⟩
⟨@@=propbox⟩
⟨*package⟩
\ProvidesExplPackage
  {\ExplFileName}{\ExplFileDate}{\ExplFileVersion}{\ExplFileDescription}
```

```
\__expl_package_check:
⟨/package⟩
```

Believe it or not, this is on my computer at work. I forgot to turn Dropbox on. (This is before I made the `git` repository, of course.)

```
{ % Create a new class called Shape
  \NewClass{Shape}

  \prop_new:c \g__propbox_Class__Shape
}

\def\__propbox_add_class_prop:w #1.#2 #3..{\prop_add:Nnn { #1 } { #2 } { #3 }}

{ % Set its properties
  \SetClassProperties{Shape}{
    .color tl,
    .parent \ReferenceClass{Shape}
  }

  \clist_set:Nn \l_tmpa_clist { #2 }
  \clist_map:Nn \l_tmpa_clist {
    \__propbox_add_class_prop:w { g__propbox_Class__Shape } ##1..
  }
}

{ % Create a new subclass Ball(Shape) with properties
  \NewClass[Shape]{Ball}[
  .radius int
  ]

  \prop_new:c { g__propbox_Class__Ball }
  \prop_set_eq:NN \g__propbox_Class__Ball \g__propbox_Class.Shape

  \clist_set:Nn \l_tmpa_clist { #2 }
  \clist_map:Nn \l_tmpa_clist {
    \__propbox_add_class_prop:w { g__propbox_Class.Ball } ##1..
  }
}

{ % Create a new Shape called myShape
  \NewObject{Shape}{myShape}
  % and set its properties; note the undefined \ClassReference should have a -NoValue- marker
  \SetObjectProperties{myShape}{
    color = blue,
  }

  \prop_new:N \g__propbox_Object__myShape

}

```

```
{ % Create a new Ball called myBall and set its properties
  \NewObject{Ball}{myBall}[
  color = purple with a hint of orange,
  radius = 5
  parent = \ReferenceObject{myShape}
  ]
}

\UseObjectProperty{myBall.color}
\SetObjectProperty{myBall.parent}{\ReferenceObject{myShape}}

```