# The propbox package
# Boxes with Properties[*]

Sean Allred[†]

Released ⟨*not released*⟩

This is some cool stuff. The basic idea is this: have a sequence of boxes that you can use (all with generated names using `\newsavebox`) and then have their properties stored as an additional token list of nearly the same name. Then, when it comes time to retrieve the boxes, just iterate through the sequence and pick some that satisfy a given filter.

## 1 Properties

### 1.1 Defining New Properties

`\DeclareBoxProperties`
Updated: 2013/08/11

`\DeclareBoxProperties {⟨properties⟩}`

Declares the set of box properties as a comma-separated list.

`\NewBoxProperty`
Updated: 2013/08/11

`\NewBoxProperty {⟨property⟩}`

Adds a *single* key to the property list.

`\RemoveBoxProperty`
Updated: 2013/08/11

`\RemoveBoxProperty {⟨property⟩}`

Removes a *single* key from the property list.

`\ClearBoxProperties`
Updated: 2013/08/11

`\ClearBoxProperties`

Removes *all* keys from the property list.

[*] This file describes v0, last revised 2013/08/13.

[†] E-mail: seallred@smcm.edu

# 2 propbox implementation

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

## 2.1 Debugging

\g__propbox_debug_mode_bool

This controls almost all log output, and will certainly clutter your log file if set. When set, propbox will show the contents of internal sequences and keys throughout the compile. It is probably hard to follow, so the usefulness of macros dependendent on this remains dubious.

```
\bool_new:N \g__propbox_debug_mode_bool
```

(*End definition for* \g__propbox_debug_mode_bool *This variable is documented on page* ??.)

## 2.2 Boxes

\g__propbox_boxes_seq

Stores all of the boxes by name. Remember that \newsavebox needs two parameters; it needs a single ⟨*cs name*⟩ and its contents. It is this ⟨*cs name*⟩ that we are storing into this sequence.

```
\seq_new:N \g__propbox_boxes_seq
```

(*End definition for* \g__propbox_boxes_seq *This variable is documented on page* ??.)

## 2.3 Properties

\g__propbox_properties_tl

This list contains all of the properties that are used for boxes along with their default values. The expected value of this variable is *at all times expected* to be a valid l3keys specification. It should only be altered by \DeclareBoxProperties and friends.

```
\tl_new:N \g__propbox_properties_tl
```

(*End definition for* \g__propbox_properties_tl *This variable is documented on page* ??.)

\g__propbox_new_box:nn

When creating a new box, care must be taken so that the new box and its properties are somehow associated with each other. This function does so by generating a name for the box (whose content is given as #1) and then using this generated name (say, \gn) for the box handle (as \gn_box) and property list, #2, as \gn_prop (using l3prop). Thus, when retrieving the properties for a box, one only needs to append _prop to the control sequence to retrive that box's properties.

```
\cs_new_protected:Npn { \g__propbox_new_box:nn } #1 #2
 {
```

Store the generated name into a local temporary token list. (It gets very tedious, not to mention inefficient (because of \seq_count:N), to type out all the time.)

```
   \tl_set:Nn \l_tmpa_tl { propbox_box_number_  \seq_count:N \__propbox_boxes_seq  }
```

Store the properties for this box in its own property list. This way, the keys can be set easily and simply to determine whether or not to select the box. The properties are stored as a property list that is checked with `\prop_get:NnN` during the filter stage.

```
  \tl_new:cn { g_ \l_tmpa_tl _properties }
  \tl_set:Nn \l_tmpa_tl { #1 }
  \clist_map_inline:Nn \l_tmpa_clist
   {
    \prop_put:Nnn \__propbox_properties_prop { ##1 } { ##2 }
   }
```

Now the moment we've all been waiting for. Create a new save box of the name `\propbox_box_number_`$N$ where $N$ is the current number of items in the collection. ($N \in \{0\} \cup \mathbf{N}$.) The `minipage` environment is used here to force page-wise sequential output only.[1]

```
  \newsavebox { \l_tmpa_tl _box }
  \savebox { \l_tmpa_tl _box } {
    \begin{minipage}{\linewidth}
      #2
    \end{minipage}
  }
```

After the box is saved, put its handle into the collection of boxes. (Remember, the name of this handle is stored in `\l_tmpa_tl`, and the `v` argspec constructs a cs name from the expanded value of `\l_tmpa_tl`, which holds the stem of our name.

```
  \seq_push:Nv \__propbox_boxes_seq
               { \l_tmpa_tl _box }

 }
```

(*End definition for* `\g__propbox_new_box:nn` *This function is documented on page* **??**.)

## 2.4 Debugging Utilities

`\__propbox_debug_mode_bool` Create a toggle to turn on/off debug mode.

```
\bool_new:c { propbox_debug_mode_bool }
```

(*End definition for* `\__propbox_debug_mode_bool` *This variable is documented on page* **??**.)

```
⟨/initex | package⟩
```

[1] This package grew out of a question on the TeX Stack Exchange site, and this seemed appropriate at the time.