

# The `acrosort` Package

D. P. Story  
Email: [dpstory@acrotex.net](mailto:dpstory@acrotex.net)

processed June 16, 2020

## Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Main Code</b>	<b>2</b>
<b>3</b>	<b>Some simple controls</b>	<b>2</b>
<b>4</b>	<b>Embedding the tiles</b>	<b>3</b>
<b>5</b>	<b>Inserting the tiles to be sorted</b>	<b>4</b>
<b>6</b>	<b>Document JavaScript for <code>acrosort</code></b>	<b>5</b>
<b>7</b>	<b>Index</b>	<b>9</b>
<b>8</b>	<b>Change History</b>	<b>10</b>

## 1 Introduction

`acrosort` is a novelty L<sup>A</sup>T<sub>E</sub>X package for embedding a series of tiled images of a picture. The tiled images are randomly arranged, then resorted before the user's eyes using a bubble sort.

This new version of `acrosort`, dated 2020/06/02 or later, supports all common workflows: `pdflatex`, `lualatex`, `xelatex`, and `dvips -> distiller`.

The `graphicx`, `eforms`, and `icon-appr` packages are automatically input by `acrosort`. When the workflow `dvips -> distiller` is used, the package `aeb-pro` is required.

For the first time, *multiple tiled bubble sorts* can appear in the same document, though only one can be sorted at a time.

## 2 Main Code

```
1 <*package>
```

### Required Packages.

```
2 \@ifpackageloaded{eforms}{\let\execjs=y}
3   {\RequirePackage[execJS]{eforms}}
4 \ifxetex\makeXasPDoff\fi
```

(2020/06/02) We require the icon-appr package.

```
5 \RequirePackage{icon-appr}
6 \RequirePackage{multido}
7 \RequirePackage{graphicx}
```

## 3 Some simple controls

Some buttons to control the starting, stopping and clearing of the image. You can change the appearance of these buttons by using the optional parameter, see the eforms documentation.

`\StartSort[<options>]{<name>}{<wd>}{<ht>}` Starts the sort for the pictures having the associated <name>.

```
8 \newcommand{\StartSort}[4] []
9   {\pushButton[\CA{Press Me}#1\A{\JS{\sortCustomStartJS;\r
10     startSortAS("#2");
11     \@appendStartSortJS}]{btStartSort}{#3}{#4}%
12 }
```

`\ClearSort[<options>]{<name>}{<wd>}{<ht>}` Clears the sort for the pictures having the associated <name>, it does this simply by making all pictures hidden.

```
13 \newcommand{\ClearSort}[4] []%
14   {\pushButton[\CA{Clear}#1\A{\JS{%
15     var f=this.getField("btn#2pic");\r
16     if(f!=null)f.display=display.hidden;\r
17     asOk2Continue = false;
18     \@appendClearSortJS
19   }]{btnClearSort}{#3}{#4}%
20 }
```

`\StopSort[<options>]{<wd>}{<ht>}` Starts the sort for the pictures having the associated <name>.

```
21 \newcommand{\StopSort}[3] []
22   {\pushButton[\CA{Stop}#1\A{\JS{%
23     asOk2Continue = false;
24     \@appendStopSortJS
25   }]{btnStopSort}{#2}{#3}%
26 }
```

`\customStartJS{script}` Inserts *script* into the beginning of the `\StartSort` command.

```
27 \newcommand{\customStartJS}[1]{\def\sortCustomStartJS{#1}}
28 \let\sortCustomStartJS@gobbletwo
```

`\appendStartSortJS{script}` Appends *script* to the `\StartSort` control.

```
29 \def\appendStartSortJS#1{\def\@rgi{#1}\ifx\@rgi\@empty
30   \let\@ppendStartSortJS\@empty\else
31   \def\@ppendStartSortJS{\r #1}\fi}
32 \let\@ppendStartSortJS\@empty
```

`\appendClearSortJS{script}` Appends *script* to the `\ClearSort` control.

```
33 \def\appendClearSortJS#1{\def\@rgi{#1}\ifx\@rgi\@empty
34   \let\@ppendClearSortJS\@empty\else
35   \def\@ppendClearSortJS{\r #1}\fi}
36 \let\@ppendClearSortJS\@empty
```

`\appendStopSortJS{script}` Appends *script* to the `\StopSort` control.

```
37 \def\appendStopSortJS#1{\def\@rgi{#1}\ifx\@rgi\@empty
38   \let\@ppendStopSortJS\@empty\else
39   \def\@ppendStopSortJS{\r #1}\fi}
40 \let\@ppendStopSortJS\@empty
```

`\asIconPic[opts]{fieldname}{wd}{ht}` There is allowance for displaying additional button images. The

```
41 \newcommand{\asIconPic}[4][{}]{% \I{\csOff{name}} required
42   \pushButton[\Ff{\FfReadOnly}\BG{}\S{S}#1\TP{1}\F{\FHhidden}
43     \PA{.5 1}{#2}{#3}{#4}}
```

## 4 Embedding the tiles

The first step is to embed the tiles using the embedding environment of icon-app.

`\asEmbedTiles[ext]{name}{n-pics}{path}` We take a graphic and explode it into rows and columns, *n-pics* is the total number of tiled pictures. We assume the tiles are created row-wise, possible by the tile-graphic package. We assume also a naming convention for the tiles if mypic is the basename of the picture or graphic, then the tiles are named mypic\_01, mypic\_02, mypic\_03, .... It is assumed a single digit

`\isPackage` index has a leading 0. Use the command `\isPackage` prior to `\asEmbedTiles`.

```
44 \newcount\as@nCnt
45 \newif\if@isPackaged\@isPackagedfalse
46 \def\isPackage{\@isPackagedtrue}
47 \let\asIconObjs@gobble
48 \newcommand{\asEmbedTiles}[4][{}]{\begingroup
49   \gdef\asNumSideShowPics{#3}%
50   \csarg\gdef\asGraphicPath#2{#4}%
51   \def\@Ext{#1}\ifx\@Ext\@empty\def\@Ext{.pdf}\else\def\@Ext{.#1}\fi
52   \@tempcnta\z@
53   \let\@embedList\@empty
```

```

54 \let\DPSIndxList\@gobble
55 \edef\z{\noexpand\g@addto@macro\noexpand
56   \asIconObjs{"#2":\asNumSideShowPics}}\z
57 \@whilenum \@tempcnta < \asNumSideShowPics \do{%
58   \as@nCnt\@tempcnta\advance\as@nCnt\@ne
59   \ifnum\as@nCnt<10\relax\edef\x{0\the\as@nCnt}\else
60     \edef\x{\the\as@nCnt}\fi
61   \edef\z{\noexpand\g@addto@macro\noexpand\DPSIndxList{"\x"}}\z
62   \ifxetex\if@isPackaged
63     \PackageWarning{acrosort}
64     {There is no support for embedding packaged\MessageBreak
65     PDFs with xelatex. Ignoring the \string\isPackage\MessageBreak
66     command}%
67     \@isPackagedfalse
68   \fi\fi
69   \if@isPackaged
70     \ifpdf
71       \edef\y{\noexpand
72         \embedIcon[name=#2pic\x,%
73         hyopts={page=\x}]{#4_package.pdf}}%
74     \else
75       \edef\y{\noexpand
76         \embedIcon[name=#2pic\x,placement=btn#2pic.\x,%
77         page=\x-1]{#4_package.pdf}}%
78     \fi
79   \else
80     \edef\y{\noexpand
81       \embedIcon[name=#2pic\x,placement=btn#2pic.\x]{#4_x\@Ext}}%
82     \fi
83     \expandafter\g@addto@macro\expandafter\@embedList\expandafter{\y}%
84     \@tempcnta\as@nCnt
85   }% do
86   \toks@=\expandafter{\@embedList}\the\toks@
87   \endgroup
88   \global\@isPackagedfalse
89 }

```

## 5 Inserting the tiles to be sorted

The next step is to insert the tiles into the body of the document, and sort them.

`\insertTiles{<width>}{<rows>}{<cols>}` Command for placing the tiles of a picture. We assume that the pictures are numbered consecutively across rows.

- `<name>` The name of the graphic (a JavaScript identifier)
- `<width>` The width of the image, the height is scaled proportionally
- `<rows>` The number of rows
- `<cols>` The number of columns

```

90 \newcommand\insertTiles[4]{\begingroup
91   \setbox\z@\hbox{%
92     \includegraphics[draft,width=#2]{\@nameuse{asGraphicPath#1}}}%
93   \edef\asGrphWd{\the\wd\z@}%
94   \@tempdima\wd\z@
95   \divide\@tempdima #4\relax
96   \edef\asTileWd{\the\@tempdima}%
97   \setlength\@tempdima{\ht\z@+\dp\z@}%
98   \setbox\z@\box\voidb@x
99   \edef\asTtlGrphHt{\the\@tempdima}%
100  \@tempdima\asTtlGrphHt\relax
101  \divide\@tempdima #3\relax
102  \edef\asTileHt{\the\@tempdima}%
103  \begin{minipage}{#2}%
104    \offinterlineskip\@tempcnta\z@
105    \multido{\iR=1+1}{#3}{\hbox{%
106      \multido{\iC=1+1}{#4}{%
107        \global\advance\@tempcnta\@ne
108        \ifnum\@tempcnta<10\relax
109          \edef\x{0\the\@tempcnta}\else
110          \edef\x{\the\@tempcnta}\fi
111          \edef\iconPresets{\noexpand\I{\noexpand\csOf{#1pic\x}}}%
112          \asIconPic[BC]{FB{true}}\presets{\iconPresets}%
113          \presets{\astile@KVs}}{btn#1pic.\x}{\asTileWd}{\asTileHt}%
114        }% inner multido
115      }% hbox, outer multido
116    \end{minipage}%
117  \endgroup
118 }

```

`\astileKVs`(*KV-pairs*) A way to pass eform key-values to the optional argument of the underlying push button.

```

119 \def\astileKVs#1{\def\astile@KVs{#1}}
120 \astileKVs{}

```

## 6 Document JavaScript for acrosort

Most of the work of this package is done with document JavaScript, and here it is.

`\customFinishJS`(*script*) Inserts *script* at the end of the bubble sort.

```

121 \newcommand{\customFinishJS}[1]{\gdef\sortCustomFinishJS{#1}}
122 \def\sortCustomFinishJS{}

123 \begin{insDLJS}{asSort}{AcroSort: Bubble Sort}
124 // Global Data:
125 var btnbase=""; // btn<name>pic.01, btn<name>pic.02, etc
126 var iconbase=""; // <name>pic01, <name>pic02, etc.
127 var iconObjs={\asIconObjs};

```

```

128 var sortName="";
129 var randomAS = new Array();
130 var timeout = 10;
131 var shutdown;
132 var asOk2Continue = false;
133 function initRandomDPS(name) {
134   var nTotalTiles = iconObjs[name];
135   for (i=1; i<=nTotalTiles; i++) randomAS[i]=i;
136   return nTotalTiles;
137 }

```

startSortAS(*name*) is the function that starts the sorting.

```

138 function startSortAS(name) {
139   if (!asOk2Continue) {
140     sortName=name;
141     randomAS = new Array();
142     var f=this.getField("btn"+name+"pic");
143     if(f!=null)f.display=display.visible;
144     asOk2Continue = true;
145     mixupAS();
146     showAS();
147     sortoutAS();
148   }
149 }
150 function mixupAS()
151 {
152   var i, rand, temp;
153   var nTotalTiles=initRandomDPS(sortName);
154   var ldps = randomAS.length;
155   for (i=1; i<= nTotalTiles; i++)
156   {
157     var rand = Math.random();
158     rand *= ldps*ldps;
159     rand = Math.ceil(rand);
160     rand = rand \% ldps;
161     if (rand == 0 ) rand = 1;
162     temp = randomAS[i];
163     randomAS[i]=randomAS[rand];
164     randomAS[rand]=temp;
165   }
166 }
167 function showAS()
168 {
169   var I,J;
170   var nTotalTiles = iconObjs[sortName];
171   btnbase="btn"+sortName+"pic.";
172   iconbase=sortName+"pic";
173   for ( var i=1; i<=nTotalTiles; i++ )
174   {
175     I=((i<10)?"0":"")+i;

```

```

176         J=((randomAS[i]<10)?"0":"")+randomAS[i];
177         var oIcon = this.getIcon(iconbase+J);
178         var f = this.getField(btnbase+I);
179         f.buttonSetIcon(oIcon);
180     }
181 }
182 function sortoutAS()
183 {
184     outerLoop(randomAS.length-1);
185 }
186 function outerLoop(i)
187 {
188     if ( asOk2Continue && (i >= 0) )
189         shutdown = app.setTimeout("app.clearTimeout(shutdown); %
190 innerLoop("+i+",1);", timeout);
191     else {
192         asOk2Continue=false;
193         \sortCustomFinishJS
194     }
195 }
196 function innerLoop(i,j)
197 {
198     var I, J;
199     if ( j <= i )
200     {
201         if (randomAS[j-1] > randomAS[j])
202         {
203             var temp = randomAS[j-1];
204             randomAS[j-1] = randomAS[j];
205             randomAS[j] = temp;
206             J=((randomAS[j-1]<10)?"0":"")+randomAS[j-1];
207             I=((j-1 < 10)?"0":"")+(j-1);
208             var oIcon = this.getIcon(iconbase+J);
209             var f = this.getField(btnbase+I);
210             f.buttonSetIcon(oIcon);
211             J=((randomAS[j]<10)?"0":"")+randomAS[j];
212             I=((j < 10)?"0":"")+j;
213             var oIcon = this.getIcon(iconbase+J);
214             var f = this.getField(btnbase+I);
215             f.buttonSetIcon(oIcon);
216         }
217         j++
218         if ( asOk2Continue ) %
219 shutdown = app.setTimeout("app.clearTimeout(shutdown); %
220 innerLoop("+i+","+j+"");", timeout);
221     }
222     else
223     {
224         i--;
225         outerLoop(i);

```

```
226     }  
227 }  
228 \end{insDLJS}  
229 \end{package}
```



## 7 Index

Numbers written in *italic* refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; numbers in *roman* refer to the code lines where the entry is used.

Symbols		E
\%	160	\embedIcon . . . . . 72, 76, 81
\@Ext	51, 81	\execjs . . . . . 2
\@embedList	53, 83, 86	
\@isPackagedfalse	45, 67, 88	<b>F</b>
\@isPackagedtrue	46	\F . . . . . 42
\@appendClearSortJS	18, 34–36	\FB . . . . . 112
\@appendStartSortJS	11, 30–32	\Ff . . . . . 42
\@appendStopSortJS	24, 38–40	\FfReadOnly . . . . . 42
\@rgi	29, 33, 37	\FHidden . . . . . 42
		<b>G</b>
<b>A</b>		\g@addto@macro . . . . . 55, 61, 83
\A	9, 14, 22	
\appendClearSortJS	3, 33	<b>I</b>
\appendStartSortJS	3, 29	\I . . . . . 41, 111
\appendStopSortJS	3, 37	\iC . . . . . 106
\as@nCnT	44, 58–60, 84	\iconPresets . . . . . 111, 112
\asEmbedTiles	3, 48	\if@isPackaged . . . . . 45, 62, 69
\asGrphWd	93	\ifpdf . . . . . 70
\asIconObjs	47, 56, 127	\ifxetex . . . . . 4, 62
\asIconPic	3, 41, 112	\includegraphics . . . . . 92
\asNumSideShowPics	49, 56, 57	\insertTiles . . . . . 4, 90
\astile@KV s	113, 119	\iR . . . . . 105
\asTileHt	102, 113	\isPackage . . . . . 3, 46, 65
\astileKV s	5, 119, 120	
\asTileWd	96, 113	<b>J</b>
\asTtlGrphHt	99, 100	\JS . . . . . 9, 14, 22
		JS functions:
<b>B</b>		startSortAS . . . . . 6
\BC	112	<b>M</b>
\BG	42	\makeXasPDOff . . . . . 4
\box	98	\multido . . . . . 105, 106
		<b>O</b>
<b>C</b>		\offinterlineskip . . . . . 104
\CA	9, 14, 22	
\ClearSort	<u>13</u>	<b>P</b>
\csarg	50	\PA . . . . . 43
\csOf	41, 111	\PackageWarning . . . . . 63
\customFinishJS	<u>121</u>	\presets . . . . . 112, 113
\customStartJS	<u>27</u>	\pushButton . . . . . 9, 14, 22, 42
		<b>R</b>
<b>D</b>		\r . . . . . 9, 15, 16, 31, 35, 39
\divide	95, 101	\RequirePackage . . . . . 3, 5–7
\DPSIndxList	54, 61	

<b>S</b>		<b>V</b>	
\S	42	\voidb@x	98
\setbox	91, 98	<b>X</b>	
\sortCustomFinishJS	121, 122, 193	\x	59–61, 72, 73, 76, 77, 81, 109–111, 113
\sortCustomStartJS	9, 27, 28	<b>Y</b>	
\StartSort	<u>8</u>	\y	71, 75, 80, 83
startSortAS (JS function)	6	<b>Z</b>	
\StopSort	<u>21</u>	\z	55, 56, 61
<b>T</b>			
\TP	42		

## 8 Change History

v1.5 (2020/08/06)

General: Embed graphics and support for  
packaged graphics . . . . . 3

v1.6 (2020/06/02)

General: Rewrite whole package to support  
pdf<sub>l</sub>at<sub>e</sub>x, lua<sub>l</sub>at<sub>e</sub>x, and x<sub>e</sub>lat<sub>e</sub>x workflows. . . . . 1