Mathematica Assembly File

Actions

- **Assemble All**

```
AcademicPensieveAssemble["all"];
```

```
Exporting C:\drorbn\AcademicPensieve\Assemble.nb
  into C:\drorbn\AcademicPensieve\nb\Assemble.pdf ...
```

```
Exporting C:\drorbn\AcademicPensieve\PublicNotebook\2009-01.one
  into C:\drorbn\AcademicPensieve\2009-01\one\\ ...
```

```
Writing C:\drorbn\AcademicPensieve\2009-01\one\Random.pdf ...
```

```
C:\drorbn\AcademicPensieve\2009-01\index.html
```

```
C:\drorbn\AcademicPensieve\2009-01\one\index.html
```

```
C:\drorbn\AcademicPensieve\index.html
```

- **Convert all .nb Files**

```
AcademicPensieveAssemble["nb"]
```

```
Exporting C:\drorbn\AcademicPensieve\Assemble.nb
  into C:\drorbn\AcademicPensieve\nb\Assemble.pdf ...
```

- **Convert all .one Files**

```
AcademicPensieveAssemble["one"];
```

```
Exporting C:\drorbn\AcademicPensieve\PublicNotebook\2008-12.one
  into C:\drorbn\AcademicPensieve\2008-12\one\\ ...
```

```
Writing C:\drorbn\AcademicPensieve\2008-12\one\Reading_Furusho.pdf ...
```

```
Exporting C:\drorbn\AcademicPensieve\PublicNotebook\Annotations\2008.one
  into C:\drorbn\AcademicPensieve\Annotations\2008\one\\ ...
```

- **Make thumbnails for all images**

```
AcademicPensieveAssemble["thumbs"]
```

- **Create Index Pages**

```
AcademicPensieveAssemble["indexes"]
```

AcademicPensieveAssembleRandom.php

```
AcademicPensieveAssemble["random.php"]
```

```
C:\drorbn\AcademicPensieve\random.php
```

- **Delete all TOC files**

```
DeleteFile /@ FileNames["TOC.m", {AcademicPensieveDirectory}, Infinity];
```

- **Delete all index files**

```
DeleteFile /@ FileNames["index.html", {AcademicPensieveDirectory}, Infinity];
```

Programs

- **Assemble All**

```
BeginPackage["AcademicPensieve`"];
AcademicPensieveDirectory = "C:\\drorbn\\AcademicPensieve";
AcademicPensieveAssemble["all"] :=
  AcademicPensieveAssemble /@ {"nb", "one", "thumbs", "indexes", "random.php"};
EndPackage[]
```

- **Utilities / General**

```
BeginPackage["AcademicPensieve`"]; Begin["`Private`"];
RootDir = AcademicPensieveDirectory;
StripRootDir[s_] := StripDir[s, RootDir];
StripDir[s_, sdir_] := Module[{dir},
   dir = StringReplace[s, sdir → ""];
   While[dir =!= "" && (StringTake[dir, 1] == "\\" || StringTake[dir, 1] == "/"),
    dir = StringDrop[dir, 1]
   ];
   dir
  ];
KosherFilename[s_String] := StringReplace[s, {
    " " → "_", ":" → "-", "/" → "-", "?" → "Q", "\"" → "'"
   }];
End[]; EndPackage[]
```

- **For .nb files**

```
BeginPackage["AcademicPensieve`"]; Begin["`Private`"];
OpenNotebooks = Notebooks[];
NB2PDF[NotebookFilename_String] := Module[
   {SplitName, PDFFilename, PDFDir, nb},
   SplitName = StringSplit[NotebookFilename, {"\\", "/"}];
   PDFFilename  = ToFileName[
     PDFDir = ToFileName[Append[Drop[SplitName, -1], "nb"]],
     StringDrop[Last@SplitName, -2] <> "pdf"
    ];
   If[
    Or[
     FileType[PDFFilename] === None,
     AbsoluteTime[FileDate[PDFFilename]] < AbsoluteTime[FileDate[NotebookFilename]]
    ],
    Print["Exporting ", NotebookFilename, " into ", PDFFilename, " ..."];
    If[FileType[PDFDir] === None, CreateDirectory[PDFDir]];
    NotebookPrint[nb = NotebookOpen[NotebookFilename], PDFFilename];
    If[! MemberQ[OpenNotebooks, nb], NotebookClose[nb]];,
    (* Print["Skipping ", NotebookFilename,"."] *)
   ];
   PDFFilename
  ];
AcademicPensieveAssemble["nb"] := Module[
   {legits, orphans},
   legits = NB2PDF /@ FileNames["*.nb", {RootDir}, Infinity];
   orphans = Complement[
     Flatten[
      If[FileType[#] === Directory,
          FileNames["*", {#}],
          {}
         ] & /@ FileNames["nb", {RootDir}, Infinity]
     ],
     legits,
     FileNames["index.html", {RootDir}, Infinity]
    ];
   DeleteFile[orphans]
  ];
End[]; EndPackage[]
```

- **For .one files**

More on the Mathematica / .NET interface is at http://reference.wolfram.com/mathematica/NETLink/tutorial/CallingNETFromMathematica.html

```
BeginPackage["AcademicPensieve`"]; Begin["`Private`"];
Needs["NETLink`"];
If[! NETObjectQ[OneNoteLink],
  InstallNET[];
  OneNoteLink = CreateCOMObject["OneNote.Application"]
 ];
One2PDF[OneNoteFilename_String] := Module[
   {
    SplitName, PDFDirectory, OneNoteDocument,
    XMLString, XML, PageDescriptors, legits = {}, tocfile
   },
   SplitName = StringSplit[OneNoteFilename, {"\\", "/"}];
   PDFDirectory = ToFileName[Flatten[{
       DeleteCases[Drop[SplitName, -1], "PublicNotebook" | "old"],
       {StringDrop[Last@SplitName, -4], "one"}
      }]];
   If[
    ! Or[
      FileType[PDFDirectory] === None,
      FileType[ToFileName[PDFDirectory, "TOC.m"]] === None,
      AbsoluteTime[FileDate[PDFDirectory]] < AbsoluteTime[FileDate[OneNoteFilename]]
     ],
    (* Print["Skipping ", OneNoteFilename,"."] *),
    Print["Exporting ", OneNoteFilename, " into ", PDFDirectory, "\\ ..."];
    If[FileType[PDFDirectory] === None, CreateDirectory[PDFDirectory]];
    OneNoteLink@OpenHierarchy[OneNoteFilename, "", OneNoteDocument];
    OneNoteLink@GetHierarchy[OneNoteDocument, 4, XMLString];
    XML = ImportString[XMLString, "XML"];
    PageDescriptors = Cases[XML, XMLElement[{_, "Page"}, page_, {}] :> page, Infinity];
    legits = OnePage2PDF[#, PDFDirectory] & /@ PageDescriptors;
    Put[XML, tocfile = ToFileName[PDFDirectory, "TOC.m"]];
    AppendTo[legits, tocfile];
    DeleteFile[Complement[FileNames["*", PDFDirectory], legits]]
   ];
  ];
OnePage2PDF[desc_List, dir_String] := Module[
   {ID, name, dateTime, lastModifiedTime, pdffilename},
   {ID, name, dateTime, lastModifiedTime} =
    {"ID", "name", "dateTime", "lastModifiedTime"} /. desc;
   pdffilename = ToFileName[dir,
     KosherFilename[name] <> ".pdf"
    ];
```

```
    If[
     Or[
      FileType[pdffilename] === None,
      AbsoluteTime[DatePlus[FileDate[pdffilename], {-$TimeZone, "Hour"}]] <
       AbsoluteTime[lastModifiedTime]
     ],
     Print["Writing ", pdffilename, " ..."];
     If[FileType[pdffilename] =!= None, DeleteFile[pdffilename]]
      OneNoteLink@Publish[ID, pdffilename, 3, ""]
    ];
    pdffilename
   ];
AcademicPensieveAssemble["one"] := If[!NETObjectQ[OneNoteLink],
   $Failed, One2PDF /@ FileNames["*.one", {RootDir}, Infinity];];
AcademicPensieveAssemble["one"] := If[!NETObjectQ[OneNoteLink], $Failed,
   One2PDF /@ FileNames["*.one", {RootDir}, Infinity]
  ];
End[]; EndPackage[]
```

- **Make Thumbnails**

```
BeginPackage["AcademicPensieve`"]; Begin["`Private`"];
ImageTypes = ("*.jpg" | "*.gif" | "*.png");
MakeThumb[ImageFilename_String] := Module[
   {SplitName, ThumbFilename, ThumbDir},
   SplitName = StringSplit[ImageFilename, {"\\", "/"}];
   ThumbFilename  = ToFileName[
     ThumbDir = ToFileName[Append[Drop[SplitName, -1], "thumbs"]],
     Last@SplitName
    ];
   If[
    And[
     SplitName[[-2]] =!= "thumbs",
     Or[
      FileType[ThumbFilename] === None,
      AbsoluteTime[FileDate[ThumbFilename]] < AbsoluteTime[FileDate[ImageFilename]]
     ]
    ],
    Print["Thumbnailing ", ImageFilename, " into ", ThumbFilename, " ..."];
    If[FileType[ThumbDir] === None, CreateDirectory[ThumbDir]];
    Export[ThumbFilename, ImageResize[Import[ImageFilename], {120}]];
   ];
   ThumbFilename
  ];
AcademicPensieveAssemble["thumbs"] := Module[
   {legits, orphans},
   legits = MakeThumb /@ FileNames[ImageTypes, {RootDir}, Infinity];
   orphans = Complement[
     Flatten[
      If[FileType[#] === Directory,
          FileNames["*", {#}],
          {}
         ] & /@ FileNames["thumbs", {RootDir}, Infinity]
     ],
     legits,
     FileNames["index.html", {RootDir}, Infinity]
    ];
   DeleteFile[orphans]
  ];
End[]; EndPackage[]
```

Cell Extraction

```
BeginPackage["AcademicPensieve`"]; Begin["`Private`"];
Clear[ExtractDataCell];
ExtractDataCell[cn_String] := ExtractDataCell[cn] = First[Cases[
     NotebookGet[EvaluationNotebook[]],
     Cell[
       cc_String?(StringMatchQ[#, ("Data Cell " <> cn <> ":\n") ~~ ___] &),
       "Text", ___
      ] :> Last[StringSplit[cc, "\n", 2]],
     Infinity
    ]];
ExtractDataCell /@ {"Index Template", "random.php"};
End[]; EndPackage[]
```

Assemble Index Pages

```
BeginPackage["AcademicPensieve`"]; Begin["`Private`"];
AssembleIndexPage[s_String] := Module[
   {
    dir, fulldir, Customizations, parentdir, next, previous, siblings, p,
    fname, t, d, SplitPath, l, FullTitle, Title, Navigator, i, j, OneNotePages,
    OneNoteData, NBDir, MathematicaNotebooks, NBFileNames, ThumbsDir,
    Images, ImageFileNames, DirectoryListing, FileListing, last, htmlfile
   },
   dir = StripRootDir[s];
   fulldir = ToFileName[RootDir, dir];
   Customizations = ToFileName[fulldir, "index.m"];
   Customizations = If[FileType[Customizations] === File, Get[Customizations], {}];
   If[dir === "", next = previous = "AcademicPensieve",
    parentdir =
     StringReplace[fulldir, par__ ~~ Shortest[{"\\", "/"}] ~~ __ ~~ EndOfString] :> par];
    siblings = Select[FileNames["*", parentdir], FileType[#] === Directory &];
    l = Length[siblings];
    {{p}} = Position[siblings, fulldir];
    next = StripDir[siblings[[1 + Mod[p, l]]], parentdir];
    previous = StripDir[siblings[[1 + Mod[p - 2, l]]], parentdir];
   ];
   fname = ToFileName[fulldir, "index.html"];
   If[
    ! Or[
      FileType[fname] === None,
      (t = AbsoluteTime[FileDate[fname]]) < AbsoluteTime[FileDate[fulldir]],
      FileType[d = ToFileName[fulldir, "nb"]] === Directory && t < Max[
          AbsoluteTime[FileDate[#]] & /@ FileNames["*.pdf", d]
        ],
      FileType[d = ToFileName[fulldir, "one"]] === Directory &&
       t < AbsoluteTime[FileDate[d]]
     ],
    (* Print["Skipping ", fulldir,"."] *),
    l = Length[SplitPath = Prepend[StringSplit[dir, {"\\", "/"}], "AcademicPensieve"]];
    FullTitle = StringJoin[(# <> ": ") & /@ SplitPath];
    Title = Last[SplitPath];
    Navigator = StringJoin[Table[
       StringJoin[
        "<a href=\"",
```

```
     StringJoin[Table["../", {l - i}]],
     "index.html\">" <> SplitPath[[i]] <> "</a>: "
    ],
    {i, l - 1}
   ]];
OneNotePages = If[FileType[ToFileName[{fulldir, "one"}, "TOC.m"]] === None, "",
  OneNoteData = Get[ToFileName[{fulldir, "one"}, "TOC.m"]][[2, 3, All, 2]];
  StringJoin[
   "<h2>Notebook Pages</h2>\n",
   "NotebookPagesNotes" /. Customizations /. "NotebookPagesNotes" → "",
   "<table border=1 cellspacing=0>\n",
   "<tr><th>Page</th><th>Created</th><th>Last Modified</th></tr>\n",
   StringJoin[StringJoin[
       "<tr align=left>\n",
       StringJoin[
        "  <td>",
        If[("isSubPage" /. #) == "true", "--- ", ""],
        "<a href=\"one/",
        KosherFilename["name" /. #],
        ".pdf\">",
        "name" /. #,
        "</a></td>\n"
       ],
       "  <td>" <>
        StringReplace["dateTime" /. #, {"T" → " ", ".000Z" → ""}] <> " UT</td>\n",
       "  <td>" <> StringReplace["lastModifiedTime" /. #,
         {"T" → " ", ".000Z" → ""}] <> " UT</td>\n",
       "</tr>\n"
      ] & /@ OneNoteData],
   "</table>"
  ]
 ];
NBDir = ToFileName[fulldir, "nb"];
MathematicaNotebooks = If[FileType[NBDir] =!= Directory, "",
  l = Length[
    NBFileNames =
     StringTake[StringReplace[#, NBDir → ""], {2, -5}] & /@ FileNames["*.pdf", NBDir]
   ];
  StringJoin[
   "<h2>Mathematica Notebooks</h2>\n",
   "MathematicaNotebooksNotes" /. Customizations /.
    "MathematicaNotebooksNotes" → "",
   "<table border=0 cellspacing=0 width=100%><tr>\n",
```

```
    "<td width=50% align=left valign=top><ul>\n",
    StringJoin[
     i = 0;
     StringJoin[
        If[(i++) == Ceiling[l / 2],
         "</ul></td><td width=50% align=left valign=top>\n", ""],
        "  <li> ",
        "<a href=\"", #, ".nb\">", #, ".nb</a>",
        " (<a href=\"nb/", #, ".pdf\">pdf</a>)\n"
       ] & /@ NBFileNames
    ],
    "</ul></td></tr></table>\n"
   ]
  ];
ThumbsDir = ToFileName[fulldir, "thumbs"];
Images = If[FileType[ThumbsDir] =!= Directory, "",
   ImageFileNames = StringDrop[StringReplace[#, ThumbsDir → ""], 1] & /@
     FileNames[ImageTypes, ThumbsDir];
   StringJoin[
    "<h2>Images</h2>\n",
    "<table border=0 cellspacing=5 width=100%>\n",
    StringJoin[
     i = 0;
     StringJoin[
        ++i;
        If[OddQ[i], "  <tr>", ""],
        "<td width=125 align=center><a href=\"",
        #, "\"><img src=\"thumbs/", #, "\"></a></td>",
        "<td align=left><a href=\"", #, "\">", #, "</a></td>",
        If[EvenQ[i], "</tr>\n", ""]
       ] & /@ ImageFileNames
    ],
    If[OddQ[i], "</tr>\n", ""],
    "</table>\n"
   ]
  ];
FileListing = FileNames["*", {fulldir}];
DirectoryListing = Select[FileListing, FileType[#] == Directory &];
FileListing = StringJoin[(
      last = Last[StringSplit[#, {"\\", "/"}]];
      " <a href=\"" <> last <> "\">" <> last <> "</a>  \n"
     ) & /@ Complement[FileListing, DirectoryListing]];
DirectoryListing = StringJoin[(
```

```
          last = Last[StringSplit[#, {"\\", "/"}]];
          " <a href=\"" <> last <> "/index.html\">" <> last <> "</a>  \n"
         ) & /@ DirectoryListing];
     htmlfile = OpenWrite[fname];
     WriteString[htmlfile,
      StringReplace[ExtractDataCell["Index Template"], {
        "<#FullTitle#>" → FullTitle,
        "<#Title#>" → Title,
        "<#next#>" → next,
        "<#previous#>" → previous,
        "<#Navigator#>" → Navigator,
        "<#OneNotePages#>" → OneNotePages,
        "<#MathematicaNotebooks#>" → MathematicaNotebooks,
        "<#Images#>" → Images,
        "<#FileListing#>" → FileListing,
        "<#DirectoryListing#>" → DirectoryListing
       }]
     ];
     Print[Close[htmlfile]]
    ]
   ];
AcademicPensieveAssemble["indexes"] := (
    AssembleIndexPage /@
     Select[FileNames["*", RootDir, Infinity], FileType[#] == Directory &];
    AssembleIndexPage[""];
   );
End[]; EndPackage[]
```

- **Assemble Random.php**

```
BeginPackage["AcademicPensieve`"]; Begin["`Private`"];
AcademicPensieveAssemble["random.php"] := Module[
  {phpfile},
  phpfile = OpenWrite[ToFileName[RootDir, "random.php"]];
  WriteString[phpfile,
   StringReplace[ExtractDataCell["random.php"], {
     "<#DB#>" → StringDrop[StringJoin[
        StringJoin[
           "\"",
           StringDrop[StringReplace[#, {RootDir → "", "\\" → "/"}], 1],
           "\",\n"
          ] & /@ FileNames["*.html" | "*.pdf", {RootDir}, Infinity]
       ], -2]
    }]
  ];
  Close[phpfile]
 ]
End[]; EndPackage[]
```

Templates

The Index Template

Data Cell Index Template:
<! Script generated - do not edit! >

```
<html>
<head>
  <link rel="stylesheet" type="text/css" href="http://www.math.toronto.edu/~drorbn/global.css">
  <title>Dror Bar-Natan: Odds, Ends, Unfinished: <#FullTitle#></title>
  </head>

<body>

<table width=100%><tr>
<td align=left>
<a href="http://www.math.toronto.edu/~drorbn/Copyleft/index.html">&copy;</a>|
<a href="http://www.math.toronto.edu/~drorbn/">Dror Bar-Natan:</a>
<a href="http://www.math.toronto.edu/~drorbn/Misc/">Odds, Ends, Unfinished:</a>
<#Navigator#>
</td>
<td align=right>
  <a href="http://katlas.math.toronto.edu/drorbn/AcademicPensieve/About.html">About</a>
    
  <a href="http://katlas.math.toronto.edu/drorbn/AcademicPensieve/random.php">Random</a>
</td>
</tr></table>

<h1><a href="../<#previous#>/index.html">&lt;&lt;</a>  <#Title#>  <a href="../<#next#>/index.html-
">&gt;&gt;</a></h1>

<h2>All Subdirectories</h2>
<#DirectoryListing#>

<#OneNotePages#>
<#MathematicaNotebooks#>
<#Images#>

<h2>All Files</h2>
<#FileListing#>

</body></html>
```

Random.php

Data Cell random.php:
<! Script generated - do not edit! >
<?php
$db = array (
<#DB#>
);
$rand= $db[array_rand($db)];
echo "<meta http-equiv=refresh content=\"0; url=$rand\">";
?>

Experiments

Do I need and image browser?

```
FileNames["*.jpg" | "*.png" | "*.gif", {AcademicPensieveDirectoty}, Infinity]
```

```
{C:\drorbn\AcademicPensieve\2008-02\Notes Project.png,
 C:\drorbn\AcademicPensieve\2008-12\AnIdealKnot.jpg, C:\drorbn\AcademicPensieve\Icon.png}
```