

The background of the entire page is a dense, overlapping pattern of stylized document icons. Each icon consists of a gray rectangle with a white border and several horizontal gray lines representing text. The icons are arranged in a way that they appear to be floating or overlapping each other, creating a textured, layered effect.

ConT_EXt

title : ConT_EXt User Module
subtitle : Formatting of ini-files
author : Peter Münster
date : April 20, 2008


```

1  \writestatus{loading}{Formatting of ini-files}
   \doifnotmode{mkiv}{\writestatus{error}{needs luatex}\wait\end}

2  \unprotect

3  \startluacode
   -- namespace
   thirddata = thirddata or { }
   thirddata.inifile = {}

4  -- sort the table
   -- s1: primary key
   -- s2: secondary key
   -- s3: third key
   local function sort_table(s1, s2, s3)
     local function cmp(a, b)
       if s2 == "" and a[s1] == b[s1] then
         if s3 == "" and a[s2] == b[s2] then
           return a[s3] < b[s3]
         else
           return a[s2] < b[s2]
         end
       else
         return a[s1] < b[s1]
       end
     end
     if s1 == "" then
       table.sort(thirddata.inifile.t, cmp)
     end
   end

5  -- write default values to table entry
   -- d: the default values separated by commas
   -- i: the index of the entry
   local function write_defaults(d, i)
     for k, v in string.gmatch(d, "[,%s]*(.-)=[^,]*") do
       thirddata.inifile.t[i][k] = v
     end
   end

6  -- generate table from ini-file
   -- d: default values for all entries
   -- s1: primary sort-key
   -- s2: secondary sort-key
   -- s3: third sort-key
   function thirddata.inifile.make_table(d, s1, s2, s3)
     thirddata.inifile.t = {}
     local i = 0
     while true do
       local l = io.read()
       if not l then
         break
       end
       while true do

```

Formatting of ini-files

```

-- check for new entry:
key = string.match(l, "^%[(.+)%]$")
if key then
  i = i + 1
  thirddata.inifile.t[i] = {}
  thirddata.inifile.t[i]["key"] = key
  write_defaults(d, i)
  break -- continue !
end
-- treat continued lines:
while string.match(l, "\\\$") do
  local c = io.read()
  l = string.match(l, "^(.*)\\\$") .. c
end
-- get a new key value pair:
key, val = string.match(l, "^([%w_]+)%s*=%s(.*)$")
if key then
  thirddata.inifile.t[i][key] = val
end
break
end
end
sort_table(s1, s2, s3)
print(table.serialize(thirddata.inifile.t))
end

7 -- let ConTeXt print the sorted table with user defined formatting
-- c: the user supplied command to print one entry
function thirddata.inifile.print(c)
  for i = 1,#thirddata.inifile.t do
    tex.print(string.format("%s\\def\\IF@index{%d}%s\\%s",
      "\\unprotect", i, "\\protect", c))
  end
end

8 -- initialise the new entry, in general to be called in the beginning
-- of the user supplied formatting command
-- i: the index of the new entry
function thirddata.inifile.newentry(i)
  for k, v in pairs(thirddata.inifile.t[i]) do
    tex.print(string.format("\\def\\IF%s{%s}", k, v))
  end
end

9 -- filter applied to values of a key
-- k: the key
-- s: the search pattern
-- r: the replace string
function thirddata.inifile.filter(k, s, r)
  for i = 1,#thirddata.inifile.t do
    thirddata.inifile.t[i][k] =
      string.gsub(thirddata.inifile.t[i][k], s, r)
  end
end

```

```

end
\stopluacode

10 \getparameters[IF@] [defaults=,sortA=,sortB=,sortC=,file=/dev/null]
\def\setupIniFile[#1]{
  \getparameters[IF@] [#1]
  \ctxlua{io.input("\IF@file")}
  \ctxlua{thirddata.inifile.make_table("\IF@defaults",
    "\IF@sortA", "\IF@sortB", "\IF@sortC")}
}
\def\IniFilePrint{\ctxlua{thirddata.inifile.print("\IF@command")}}
\def\IniFileNewEntry{\ctxlua{thirddata.inifile.newentry(\IF@index)}}
\def\IniFileFilter[#1] [#2] [#3]{\ctxlua{thirddata.inifile.filter("#1",
  "#2", "#3")}}

11 \protect

12 \doifnotmode{demo}{\endinput}

Usage example:

13 \startbuffer[thewho]
  [p_t]
  givenname = Peter
  surname = Townshend
  birthyear = 1945
  comment = 100 % with nobreakspace

14 [r_d]
  givenname = Roger
  surname = Daltrey
  comment = 100% with thinspace

15 [j_e]
  givenname = John
  surname = Entwistle

16 [k_m]
  givenname = Keith
  surname = Moon
  birthyear = 1946

17 [k_j]
  givenname = Kenney
  surname = Jones
  birthyear = 1948
  \stopbuffer
  \savebuffer[thewho]

18 \usemodule[inifile]

19 \setupIniFile[defaults={birthyear=1944,comment=},
  sortA=birthyear,sortB=key,command=FormatMember,file=\jobname-thewho.tmp]
\IniFileFilter[comment] [\%\% ] [\letterpercent\space ]
\IniFileFilter[comment] [\%\%] [\letterpercent ]

```

Formatting of ini-files

```
20 \setupTABLE[frame=off,width=0.5\textwidth]
    \nonknuthmode

21 \def\IFbirthyear{}
    \def\FormatMember{
        \edef\LastBirthyear{\IFbirthyear}
        \IniFileNewEntry
        \doifnot\IFbirthyear\LastBirthyear{\section{\IFbirthyear}}
        \subsection{\WORD{\IFkey}}
        \bTABLE\bTR
        \bTD Given name: \IFgivenname\eTD\bTD Surname: \IFsurname\eTD
        \eTR\eTABLE
        \doifsomething\IFcomment{Comment to show the treatment of the percent
            sign: \IFcomment}}

22 \starttext
    \title{The Who}
    \IniFilePrint
    \stoptext
```