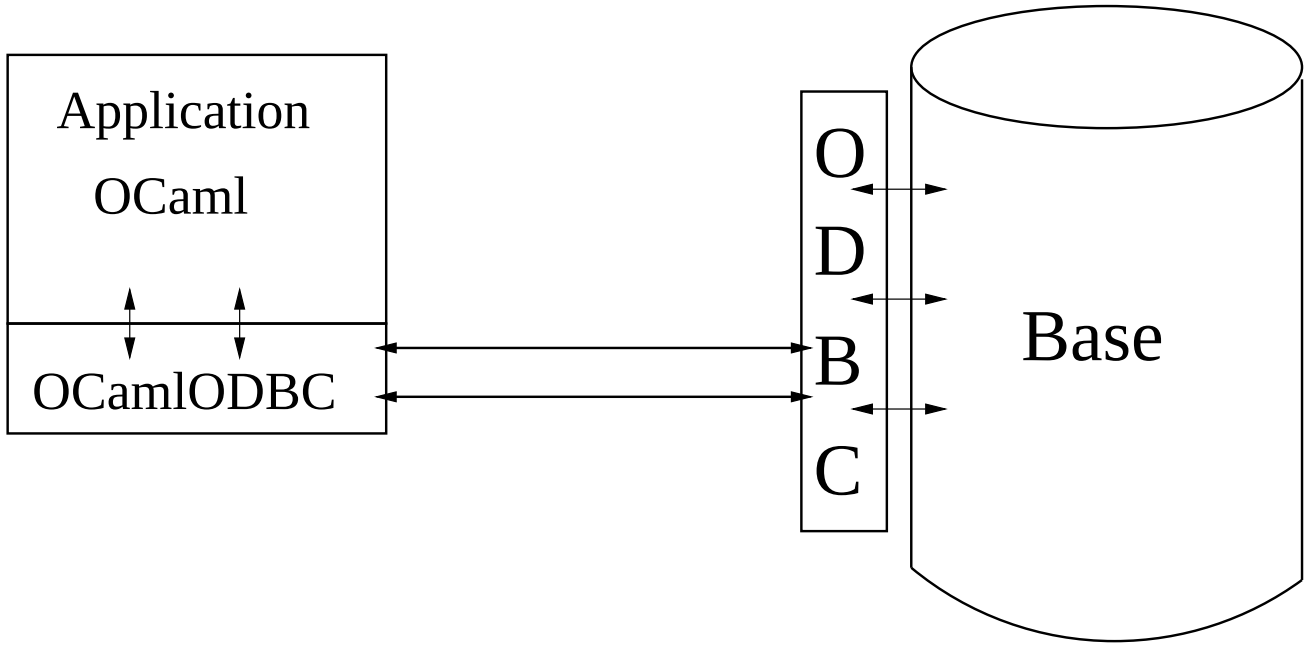
Développements dans le cadre de l'ODL OCaml

1. OCamlODBC
2. OCamldoc
3. Cameleon et les cliquodromes
4. Futurs développements
5. Démo de Cameleon sur l'outil `hump`

OCamlODBC



OCamlODBC - Bases supportées

- MySQL
- PostgreSQL
- Microsoft Access
- OpenIngres
- DB2

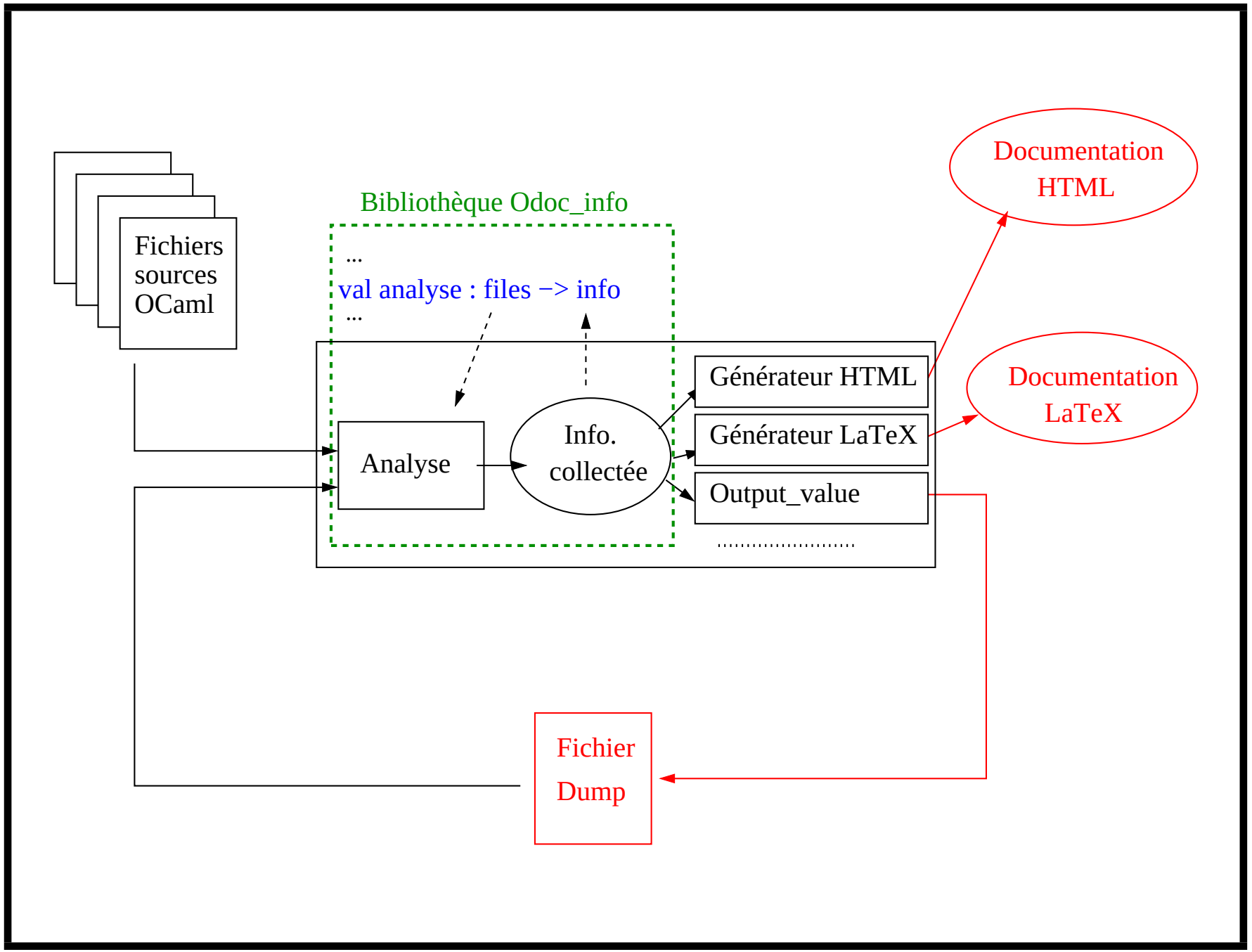
+ bases supportées par unixODBC.

OCamlODBC - Interface

```
type connection

val connect : string → string → string → connection

val disconnect : connection → unit

val execute : connection → string → int * string
list list

val execute_with_info : connection → string → int *
(string * Libocaml_odbc.sql_column_type) list * string
list list
```

Outil de documentation : OCamldoc

- architecture
- syntaxe des commentaires


Fichiers sources OCaml
Bibliothèque Odoc_info
...
val analyse : files −> info
...
Analyse
Info. collectée
Générateur HTML
Générateur LaTeX
Output_value
.....
Documentation HTML
Documentation LaTeX
Fichier Dump

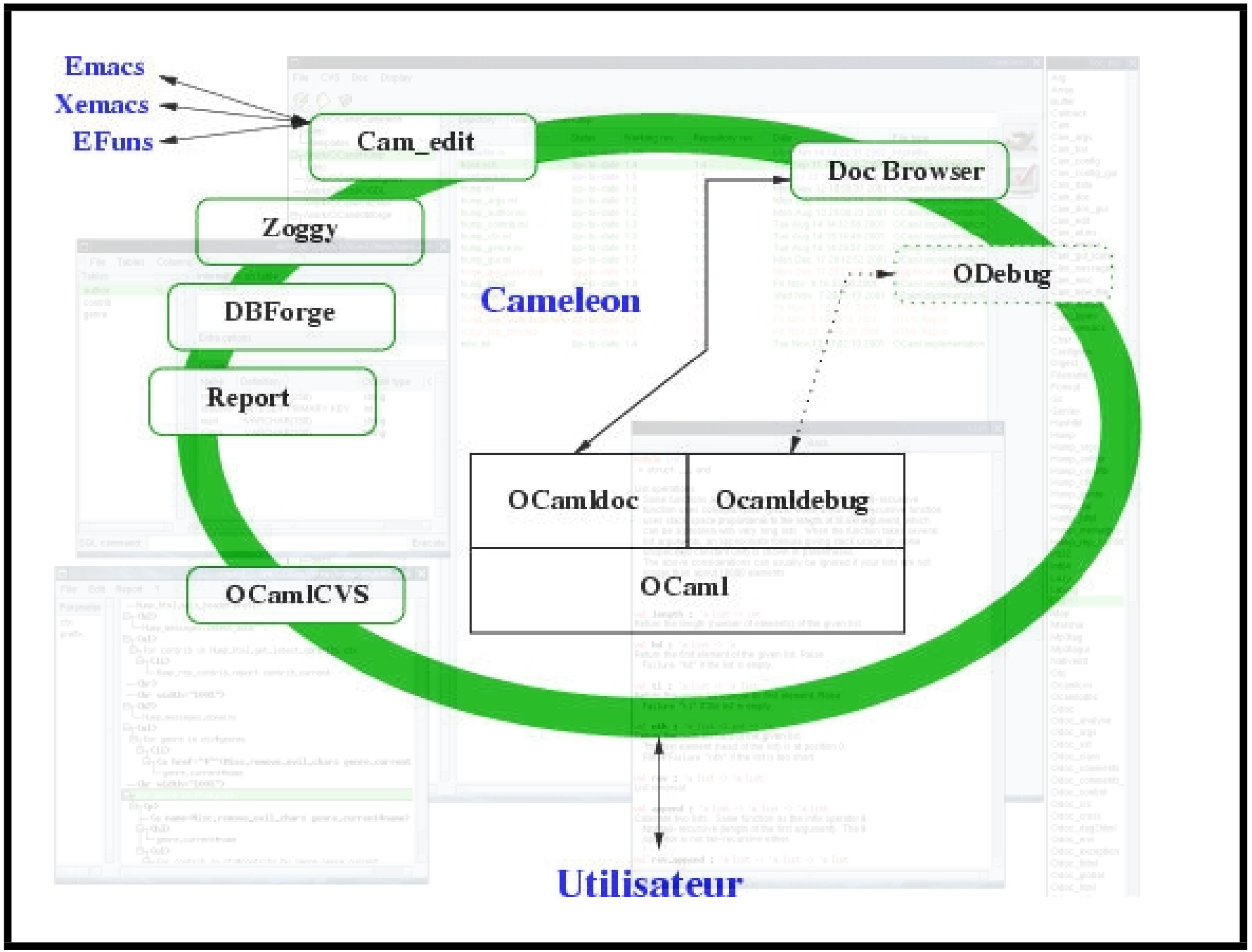
OCamldoc - Syntaxe des commentaires

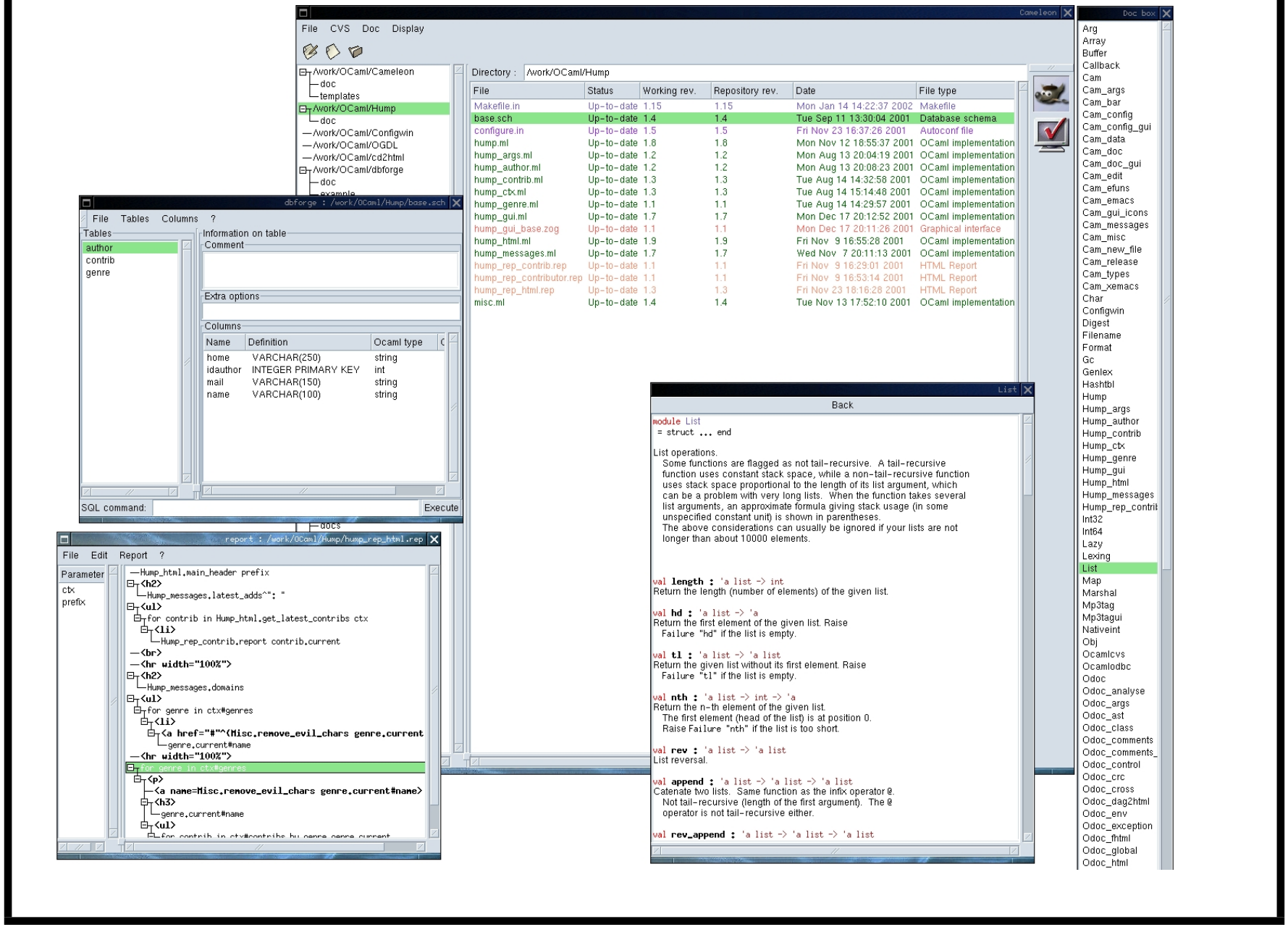
```
(** Un <I>commentaire</I> se compose d'une partie {i texte}
  et d'une partie ``a la Javadoc''.
@param x explication du parametre x.
@param y explication du parametre y.
@raise Failure quand on echoue.
@deprecated indique que l'element ne doit plus etre utilise
@author Lucky Luke
@author Jessie James
@since 1887
@version 1.2.3
@see <http://www.inria.fr> le site de l'inria
@see 'doc/spec.txt' pour plus d'information
@see ``OCaml reference manual'' pour plus d'information
@return l'age du capitaine
*)
```

Cameleon et les cliquodromes

Cameleon =

OCamlCVS
+ DBForge
+ Report
+ Zoggy
+ OCamldoc et butineur de documentation
+ interface avec Emacs, EFuns, XEmacs.


Emacs
Xemacs
EFuns
Cam_edit
Doc Browser
Zoggy
ODebug
Cameleon
DBForge
Report
OCamldoc
Ocamldebug
OCaml
OCamlCVS
Utilisateur


Cameleon
File CVS Doc Display
Directory : /work/OCaml/Hump
/work/OCaml/Cameleon
doc
templates
/work/OCaml/Hump
doc
/work/OCaml/Configwin
/work/OCaml/OGDL
/work/OCaml/cd2html
/work/OCaml/dbforge
doc
File Status Working rev. Repository rev. Date File type
Makefile.in Up-to-date 1.15 1.15 Mon Jan 14 14:22:37 2002 Makefile
base.sch Up-to-date 1.4 1.4 Tue Sep 11 13:30:04 2001 Database schema
configure.in Up-to-date 1.5 1.5 Fri Nov 23 16:37:26 2001 Autoconf file
hump.ml Up-to-date 1.8 1.8 Mon Nov 12 18:55:37 2001 OCaml implementation
hump_args.ml Up-to-date 1.2 1.2 Mon Aug 13 20:04:19 2001 OCaml implementation
hump_author.ml Up-to-date 1.2 1.2 Mon Aug 13 20:08:23 2001 OCaml implementation
hump_contrib.ml Up-to-date 1.3 1.3 Tue Aug 14 14:32:58 2001 OCaml implementation
hump_ctx.ml Up-to-date 1.3 1.3 Tue Aug 14 15:14:48 2001 OCaml implementation
hump_genre.ml Up-to-date 1.1 1.1 Tue Aug 14 14:29:57 2001 OCaml implementation
hump_gui.ml Up-to-date 1.7 1.7 Mon Dec 17 20:12:52 2001 OCaml implementation
hump_gui_base.zog Up-to-date 1.1 1.1 Mon Dec 17 20:11:26 2001 Graphical interface
hump_html.ml Up-to-date 1.9 1.9 Fri Nov 9 16:55:28 2001 OCaml implementation
hump_messages.ml Up-to-date 1.7 1.7 Wed Nov 7 20:11:13 2001 OCaml implementation
hump_rep_contrib.rep Up-to-date 1.1 1.1 Fri Nov 9 16:29:01 2001 HTML Report
hump_rep_contributor.rep Up-to-date 1.1 1.1 Fri Nov 9 16:53:14 2001 HTML Report
hump_rep_html.rep Up-to-date 1.3 1.3 Fri Nov 23 18:16:28 2001 HTML Report
misc.ml Up-to-date 1.4 1.4 Tue Nov 13 17:52:10 2001 OCaml implementation
Doc box
Arg
Array
Buffer
Callback
Cam
Cam_args
Cam_bar
Cam_config
Cam_config_gui
Cam_data
Cam_doc
Cam_doc_gui
Cam_edit
Cam_efuns
Cam_emacs
Cam_gui_icons
Cam_messages
Cam_misc
Cam_new_file
Cam_release
Cam_types
Cam_xemacs
Char
Configwin
Digest
Filename
Format
Gc
Genlex
Hashtbl
Hump
Hump_args
Hump_author
Hump_contrib
Hump_ctx
Hump_genre
Hump_gui
Hump_html
Hump_messages
Hump_rep_contrib
Int32
Int64
Lazy
Lexing
List
Map
Marshal
Mp3tag
Mp3tagui
Nativeint
Obj
Ocamlcvs
Ocamlodbc
Odoc
Odoc_analyse
Odoc_args
Odoc_ast
Odoc_class
Odoc_comments
Odoc_comments_
Odoc_control
Odoc_crc
Odoc_cross
Odoc_dag2html
Odoc_env
Odoc_exception
Odoc_fhtml
Odoc_global
Odoc_html
dbforge : /work/OCaml/Hump/base.sch
File Tables Columns ?
Tables
author
contrib
genre
Information on table
Comment
Extra options
Columns
Name Definition Ocaml type
home VARCHAR(250) string
idauthor INTEGER PRIMARY KEY int
mail VARCHAR(150) string
name VARCHAR(100) string
SQL command:
Execute
docs
report : /work/OCaml/Hump/hump_rep_html.rep
File Edit Report ?
Parameter
ctx
prefix
Hump_html.main_header prefix
<h2>
Hump_messages.latest_adds^": "

for contrib in Hump_html.get_latest_contribs ctx

Hump_rep_contrib.report contrib.current

<hr width="100%">
<h2>
Hump_messages.domains

for genre in ctx#genres

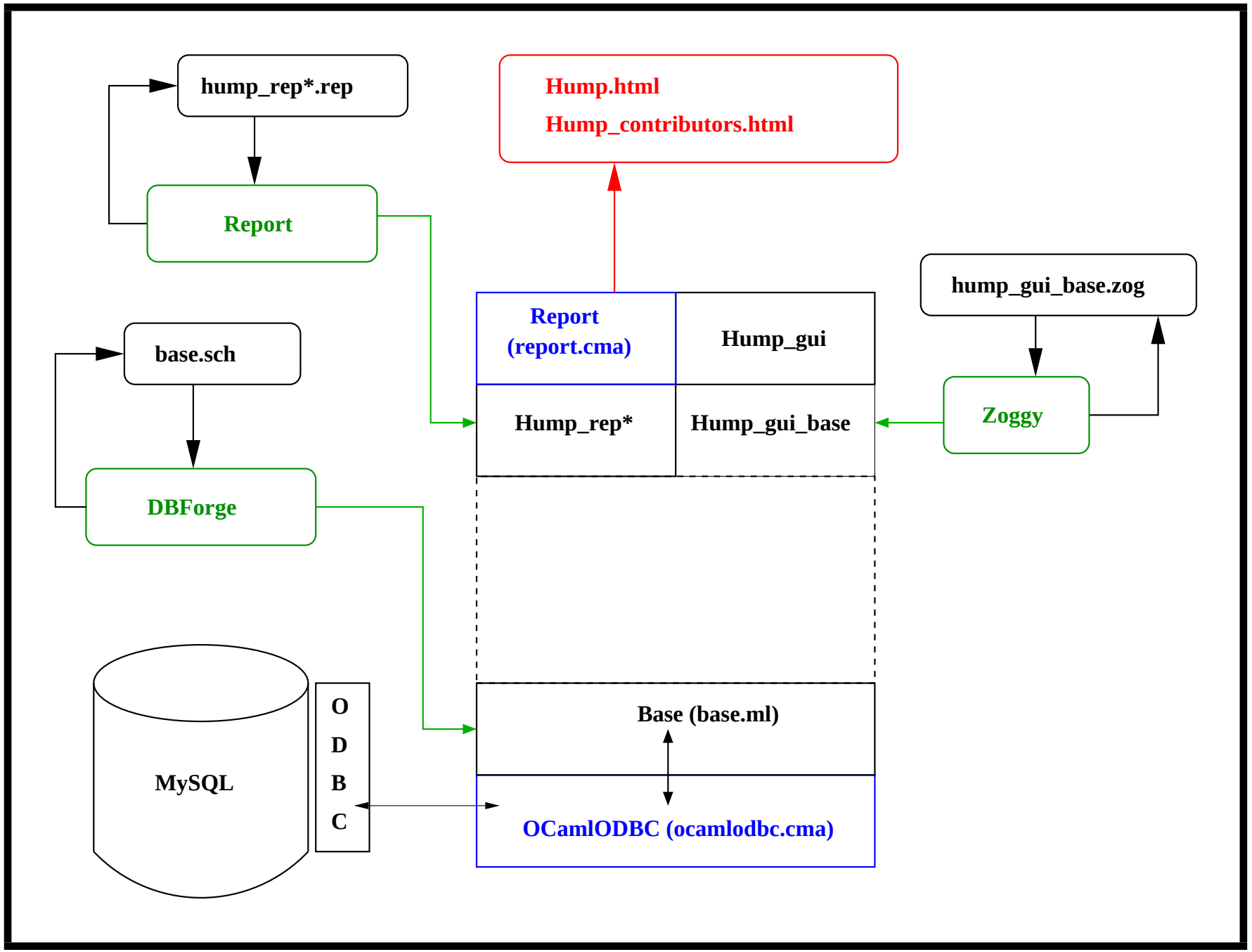
<a href="#"^(Misc.remove_evil_chars genre.current
genre.current#name
<hr width="100%">
for genre in ctx#genres
<p>
<a name=Misc.remove_evil_chars genre.current#name>
<h3>
genre.current#name

List
Back
module List
= struct ... end
List operations.
Some functions are flagged as not tail-recursive. A tail-recursive
function uses constant stack space, while a non-tail-recursive function
uses stack space proportional to the length of its list argument, which
can be a problem with very long lists. When the function takes several
list arguments, an approximate formula giving stack usage (in some
unspecified constant unit) is shown in parentheses.
The above considerations can usually be ignored if your lists are not
longer than about 10000 elements.
val length : 'a list -> int
Return the length (number of elements) of the given list.
val hd : 'a list -> 'a
Return the first element of the given list. Raise
Failure "hd" if the list is empty.
val tl : 'a list -> 'a list
Return the given list without its first element. Raise
Failure "tl" if the list is empty.
val nth : 'a list -> int -> 'a
Return the n-th element of the given list.
The first element (head of the list) is at position 0.
Raise Failure "nth" if the list is too short.
val rev : 'a list -> 'a list
List reversal.
val append : 'a list -> 'a list -> 'a list
Catenate two lists. Same function as the infix operator @.
Not tail-recursive (length of the first argument). The @
operator is not tail-recursive either.
val rev_append : 'a list -> 'a list -> 'a list

Futurs développements

- Débogueur graphique
- Outil de construction de site web ?
- Outils de test / interface avec Fort
- Support des Makefiles dans Cameleon
- Intégration d'OCamldoc dans OCaml
- Site de doclets pour OCamldoc

Utilisation de Cameleon pour `hump`


hump_rep*.rep
Report
Hump.html
Hump_contributors.html
hump_gui_base.zog
Report
(report.cma)
Hump_gui
base.sch
Hump_rep*
Hump_gui_base
Zoggy
DBForge
Base (base.ml)
MySQL
O
D
B
C
OCamlODBC (ocamlodbc.cma)