Entity Relationship Modeling Toolkit Documentation

Release 0.0.1

Fischbacher Berndt **Homolka Nicolas**
Passet Christian **Prinz Andreas** **Hantich Tobias**
Reichl Christoph **Stavarache Mario** **Häusler Paul**

Apr 03, 2019

CONTENTS

CHAPTER

ONE

MANUAL

1.1 Requirements

To make use of ermtk you have to install python3 and the following modules for python3:

- cmd2
- lxml
- BaseXClient
- cx_Oracle
- keyboard
- psycopg2
- mysql
- mysqlclient
- mysql-connector
- pyodbc
- networkx
- graphviz
- faker
- tkinter
- pygraphviz

This modules can be installed by running the install.sh skript(located in the Ermtk directory).

1.2 Installation

1. Clone the Repository to a Path your choice.
2. Now go to the Ermtk folder and copy the path to the ermtk.py file.
3. Next go to /home/bin (if you don't have this directory, you have to create it and add the path to the PATH variable)
4. Run this command: `ln -s [path to ermtk.py] ermtk`

- Info: Type `ermtk help` for help
- Example: `ln -s ~/Ermtk/ermtk.py ermtk`

CHAPTER

TWO

COMMANDS AND SUB-COMMANDS

```
usage: ermtk [-h] [-v]
             {erdgenerate,open,close,exit,shell,bye,list,erdfocus,blockdiagram,
↪ddlgenerate,dmlgenerate,dmlform,config}
             ...
```

2.1 Named Arguments

-v, --version show program's version number and exit

2.2 sub-commands

subparser Possible choices: erdgenerate, open, close, exit, shell, bye, list, erdfocus, block-diagram, ddlgenerate, dmlgenerate, dmlform, config

sub-command help

2.3 Sub-commands:

2.3.1 erdgenerate

Generate an ERD from XERML Modell

```
ermtk erdgenerate [-h] [-i INPUTFILE] [-o OUTPUT] [-n NOTATION] [-t TYP] [-a]
                  [-c] [-g] [-p] [-d] [-v] [-l LOC] [-s] [--auto]
```

Named Arguments

-i, --inputfile Inputfile

-o, --output Outputfile

-n, --notation Takes a value to define the notation Example: –notation crowfoot

-t, --typ Attributes with types are displayed in the ERD

-a, --attr The ERD displays Attributes

-c, --color	The ERD is colored
-g, --graphml	The Output-Type is a GraphML File
-p, --pic	The Output-Type is a Pic-File
-d, --draw	The Output-Type is a LibreOffice Draw File
-v, --viz	The Output-Type is a Graphviz File
-l, --loc	Define the output language
-s, --show	Shows generated Diagramm in Programm
--auto	ERD generated with default options

Examples

```
ermtk erdgenerate  -i sis.xerml.xml  -o sis.graphml -p
```

```
ermtk erdgenerate  -i sis.xerml.xml  -o sis.graphml -n crowfoot -g
```

2.3.2 open

Opens a XERML-File to use it in the Shell

```
ermtk open [-h] [-i INPUTFILE] [-l LANGUAGE] [-t TYPE]
```

Named Arguments

-i, --inputfile	Inputfile
-l, --language	Laguagefile
-t, --type	Typefile

Examples

```
ermtk open  -i sf.bug.xerml.xml
```

```
ermtk open -l sf.bug.xerml.loc.xml
```

2.3.3 close

Closes a XERML-File if it is not needed anymore

```
ermtk close [-h] [-t] [-l]
```

Named Arguments

-t, --type	Typefile
-l, --language	Languagefile

Examples

```
ermtk close
```

```
ermtk close  -l
```

2.3.4 exit

Exit the shell

```
ermtk exit [-h]
```

2.3.5 shell

Enters the shell

```
ermtk shell [-h]
```

2.3.6 bye

Exit the shell

```
ermtk bye [-h]
```

2.3.7 list

Generate an ERD from XERML Modell

```
ermtk list [-h] [-all] [-ent] [-rel] [-attr]
```

Named Arguments

-all	List all Entity-Types and Relationship-Types
-ent	List Entity-Types
-rel	List Relationship-Types
-attr	List Attributes

Examples

```
ermtk list  -all
```

```
ermtk list -ent
```

2.3.8 erdfocus

Generate an ERD focusing on specified Entities

```
ermtk erdfocus [-h] [-i INPUTFILE] [-o OUTPUT] [-f FILTER] [-r RELCOUNTER]
               [-n NOTATION] [-t TYP] [-a] [-c] [-g] [-p] [-d] [-v] [-l LOC]
               [--auto]
```

Named Arguments

-i, --inputfile	Inputfile
-o, --output	Outputfile
-f, --filter	Takes a value to filter the ERD
-r, --relcounter	Takes a number to define how much you want to see.
-n, --notation	Takes a value to define the notation Example: –notation crowfoot
-t, --typ	Attributes with types are displayed in the ERD
-a, --attr	Attributes are displayed in the ERD
-c, --color	The ERD is colored
-g, --graphml	The Output-Type is a GraphML File
-p, --pic	The Output-Type is a Pic-File
-d, --draw	The Output-Type is a LibreOffice Draw File
-v, --viz	The Output-Type is a Graphviz File
-l, --loc	Define the output language
--auto	ERD generated with default options

Examples

```
ermtk erdfocus -i sis.xerml.xml -f "Klasse" -r 1
```

2.3.9 blockdiagram

Generate an blockdiagram of the relational model

```
ermtk blockdiagram [-h] [-i INPUTFILE] [-l LANGUAGE]
```

Named Arguments

-i, --inputfile	Inputfile
-l, --language	Define the output language

Examples

```
ermtk blockdiagram  -i aaa.xerml.xml
```

2.3.10 ddlgenerate

Convert an XERML Modell into DDL-Commands

```
ermtk ddlgenerate [-h] [-i INPUTFILE] [-o OUTPUT] [--amount AMOUNT] [-t TYP]
                  [-l LOC] [--auto] [--alltables] [--notnull] [--format]
                  [--constaft] [--consttabl] [--keyword]
                  databasetype
```

Positional Arguments

databasetype	Available Databasetype: basexml, basexschema, mysql, oracle, postgresql, rel, sqlserver, sqlite

Named Arguments

-i, --inputfile	Inputfile
-o, --output	Outputfile
--amount	Amount of XML data (only availible for BaseX)
-t, --typ	Typdescription
-l, --loc	Localisation
--auto	Automatically run against database
--alltables	All relation as tables
--notnull	Deactivates the not null
--format	Format for outputfile
--constaft	Constraints after create Tables
--consttabl	Constraints as Tableconstraints
--keyword	Keywords as uppercase

Examples

```
ermtk ddlgenerate -i aaa.xerml.xml -o aaa.basex.xml --amount 3 basexml
```

```
ermtk ddlgenerate -i tank.xerml.xml -o tank.mysql.sql -t tank.xerml.ty.xml --auto  --
↪notnull mysql
```

2.3.11 dmlgenerate

Generate example data in form of DML-Commands

```
ermtk dmlgenerate [-h] [-i INPUTFILE] [-o OUTPUT] [--alltables] [--auto]
                  [-t TYP] [-l LOC]
                  database amount
```

Positional Arguments

database	Available Databases: baseX, mysql, oracle, postgresql, rel, sqlserver
amount	Amount of Example Data

Named Arguments

-i, --inputfile	Inputfile
-o, --output	Outputfile
--alltables	All relations as tables
--auto	Automatically run against database
-t, --typ	Typdescription
-l, --loc	Localisation

Examples

```
ermtk dmlgenerate -i mondial.xerml.xml -o mondial.dml.sql oracle 4
```

```
ermtk dmlgenerate -i mondial.xerml.xml -o mondial.dml.sql sqlserver 4 --alltables
```

2.3.12 dmlform

Generate an entry form for typ in example data

```
ermtk dmlform [-h] [-i INPUTFILE] [-t TYP] [-l LOC] [-o OUTPUT] [--alltables]
              database
```

Positional Arguments

database	Database

Named Arguments

-i, --inputfile	Inputfile
-t, --typ	Typdescription
-l, --loc	Localisation

-o, --output	Outputfile
--alltables	All relations as tables

Examples

```
ermtk dmlform -i sf.xerml.xml Rel
```

2.3.13 config

Configure database connection attributes

```
ermtk config [-h] [-s]
             {basex,oracle,postgresql,sqlite,sqlserver,mysql,rel} ...
```

Named Arguments

-s, --save	Save current configuration

config-commands

subparser	Possible choices: basex, oracle, postgresql, sqlite, sqlserver, mysql, rel config-command help

Sub-commands:

basex

BaseX connection attributes

```
ermtk config basex [-h] [-a ADDRESS] [-por PORT] [-usr USER] [-pwd PASSWORD]
```

Named Arguments

-a, --address	Configure Address DEFAULT = Localhost
-por, --port	Configure Port DEFAULT = 1984
-usr, --user	Configure User DEFAULT = admin
-pwd, --password	Configure Password DEFAULT = admin

Examples

```
ermtk config basex  -a 192.168.0.3 -por 1133
```

oracle

Oracle connection attributes

```
ermtk config oracle [-h] [-a ADDRESS] [-s SERVICENAME] [-usr USER]
                    [-pwd PASSWORD]
```

Named Arguments

-a, --address	Configure Address DEFAULT = 127.0.0.1
-s, --servicename	Configure ServiceName DEFAULT = XE
-usr, --user	Configure User DEFAULT = i13075
-pwd, --password	Configure Password DEFAULT = topsecret

Examples

```
ermtk config oracle  -a 123.123.123.123 -s XE -usr admin -pwd topsecret
```

postgresql

PostgreSQL connection attributes

```
ermtk config postgresql [-h] [-a ADDRESS] [-d DATABASE] [-usr USER]
                        [-pwd PASSWORD]
```

Named Arguments

-a, --address	Configure Address DEFAULT = localhost
-d, --database	Configure Database DEFAULT = test
-usr, --user	Configure User DEFAULT = postgres
-pwd, --password	Configure Password DEFAULT = postgres

Examples

```
ermtk config postgresql -d postgres -usr root -pwd QWERasdf12
```

sqlite

SQLite connection attributes

```
ermtk config sqlite [-h] [-d DIRECTORY]
```

Named Arguments

-d, --directory Configure Directory DEFAULT = /home/christoph/GitCopy/Ermtk/SQLite_DB.db

Examples

```
ermtk config sqlite -d /home/SQLite_DB.db
```

sqlserver

SQLServer connection attributes

```
ermtk config sqlserver [-h] [-dri DRIVER] [-ser SERVER] [-da DATABASE]
                       [-usr USERID] [-pwd PASSWORD]
```

Named Arguments

-dri, --driver Configure Driver DEFAULT = ODBC Driver 17 for SQL Server

-ser, --server Configure Server DEFAULT = localhost

-da, --database Configure Database DEFAULT = mydatabase

-usr, --userid Configure UserID DEFAULT = sa

-pwd, --password Configure Password DEFAULT = xX94Hugo

Examples

```
ermtk config sqlserver -dri ODBC Driver 17 for SQL Server
```

mysql

MySQL connection attributes

```
ermtk config mysql [-h] [-a ADDRESS] [-usr USER] [-pwd PASSWORD]
                   [-da DATABASE]
```

Named Arguments

-a, --address Configure Address DEFAULT = Localhost

-usr, --user Configure User DEFAULT = root

-pwd, --password Configure Password DEFAULT = rootpasswordgiven

-da, --database Configure Database DEFAULT = mydatabase

rel

REL connection attributes

```
ermtk config rel [-h]
```