csv2oerp Documentation

Release

Stéphane Mangin

August 30, 2012

CONTENTS

1 Introduction 1

2 Quick start 3

3 Download and install 5

4 Documentation 7

5 Supported Desktop versions 9

6 Supported Python versions 11

7 License 13

8 Bugs or suggestions 15

CHAPTER

ONE

INTRODUCTION

csv2oerp is a simple tool to perform a data migration, through an *OpenERP* mapping bounded to a *CSV* file. You can easily order your data as the manner of *OpenERP* (*models*, *fields*). The structure of the mapping is simple and intuitive, you can also perform a pre-process on data being imported, omit it or just modify it.

CHAPTER

TWO

QUICK START

Import csv2oerp and some callbacks:

```
>>> from csv2oerp import Import_session
>>> from csv2oerp.fields import Column, Custom, Relation
>>> from csv2oerp.callbacks import get_id, get_ids
```

Configure OpenERP connection:

```
>>> host = '198.168.0.1'
>>> port = 8069
>>> dbname = 'database'
>>> user = 'admin'
>>> pwd = 'admin'
>>> csv2oerp.connect(host, port, user, pwd, dbname)
```

Create a new importation instance:

```
>>> example_file = Import_session()
```

Set the CSV file to use:

```
>>> example_file.set_input_file('example_file.csv', ',', '"', 'utf-8')
```

Define a custom callback(Field's value pre-treatment):

```
>>> def country_code(self, model, field, value, line):
...     """Return the first two uppered characters from current column value
...     """
...     return value[:2].upper()
```

Define your mapping to link both csv and OpenERP:

```
>>> mapping = {
...     # Define a base model (Always in list form)
...     'res.partner.address': [
...         {
...
...         # Custom field's value
...         'type':         Custom('default', search=True),
...
...         # Concatenated columns fields
...         'street':       Column([7, 6]),
...         'street2':      Column([8, 5]),
...
...         # Simple fields
```

```
...         'zip':          Column(9),
...         'city':         Column(10),
...         'cedex':        Column(68),
...         'phone':        Column(14),
...         'fax':          Column(15),
...         'email':        Column(17),
...
...         # Model's relation with dynamic insertion from OpenERP database
...         # Not native object from OpenERP framework
...         'region_id':    Custom(11, get_id('res.region', ['name']))
...         'dep_id':       Custom(12, get_id('res.dep', ['name']))
...
...         # Model's relations not updated if exists
...         'country_id':   Relation('REL_res_partner_address,NO_UPDATE::res.country'),
...
...         # Model's relations with unique value between objects
...         'partner_id':   Relation('REL_res_partner_address::res.partner', search=True),
...
...         },
...     ],
...
...     # Define relations between models
...     'REL_res_partner_address::res.partner': {
...         'name':         Column(1),
...         'siren':        Column(2),
...         'website':      Column(16),
...         'comment':      Column(56),
...         },
...
...     # Relation without write if exists
...     'REL_res_partner_address,NO_UPDATE::res.country':  {
...         'code':         Column(13, callback=_country_code, search=True),
...         'name':         Column(13, search=True),
...         },
...     })
>>> example_file.set_mapping(mapping)
```

Finally start the import process:

```
>>> example_file.start()
```

And show statistics of objects's activities during the importation process:

```
>>> csv2oerp.show_stats()
```

CHAPTER

THREE

DOWNLOAD AND INSTALL

See *download-install* section.

CHAPTER

FOUR

DOCUMENTATION

See *content* section.

CHAPTER

FIVE

SUPPORTED DESKTOP VERSIONS

All architectures.

CHAPTER

SIX

SUPPORTED PYTHON VERSIONS

csv2oerp support Python versions 2.6 and 2.7.

CHAPTER

SEVEN

LICENSE

This software is made available under the LGPLv3 license.

CHAPTER

EIGHT

BUGS OR SUGGESTIONS

Please, feel free to report bugs or suggestions in the Bug Tracker!