

= Thunderbird Filterregeln für Apple Mail aufbereiten

Thunderbird Regeln finden:

```
$ find ~/.thunderbird -name msgFilterRules.dat
```

Apple Mail Regeln finden:

```
$ find ~/Library/Mail -name MessageRules.plist
```

Ein sehr, sehr unfertiges Fragment:

```
#!/usr/bin/ruby -w

require 'rexml/document'
# $ sudo gem install uuid -v 2.0.0
require 'uuid'

module Mail
  module Filter
    class Rule
    end
  end
end
module Apple
  # Add a key-value pair to a plist-dictionary
  def Apple.add parent, key, type, value
    k = REXML::Element.new 'key'
    k.text = key
    parent << k
    k = REXML::Element.new type.to_s
    k.text = "#{value}" if !value.nil?
    parent << k
    return k
  end
  def Apple.create_rules factory_rules = true
    dst = REXML::Document.new()
    dst << REXML::XMLDecl.new('1.0', 'utf-8')
    dst << REXML::DocType.new('plist', 'PUBLIC "-//Apple//DTD PLIST
1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd"')
    plist = REXML::Element.new 'plist'
    plist.add_attribute 'version', '1.0'
    dst << plist
    dict = REXML::Element.new 'dict'
    plist << dict
    array = Apple.add(dict, :rules, :array, nil)
    Apple.add dict, :version, :integer, 2
    if factory_rules
    end
    return dst
  end
end
```

```
end
module ThunderBird
  def ThunderBird.each_rule src
    kv_pat = /([^(=]+)="([^(=]+)"/
    tmp={}
    src.each_line do |l|
      m = kv_pat.match l
      raise "Couldn't match line [#{l}]" if m.nil?
      if m[1] == 'name'
        yield tmp if !tmp[name].nil?
        tmp.clear
      end
      tmp[m[1].to_sym] = m[2]
    end
    yield tmp if !tmp[name].nil?
  end

  def ThunderBird.rules_to_apple_mail src, dst=nil
    array = nil
    if dst.nil?
      dst = Apple.create_rules
    else
      raise 'TODO find and assign \'array\''
    end

    ThunderBird.each_rule(src) do |rule|
      puts rule[:rule]
    end
    dst
  end
end
end

src=File.open('/home/m/.thunderbird/....default/Mail/.../msgFilterRules.dat'
)
puts Mail::ThunderBird.rules_to_apple_mail(src)

# file:///usr/lib64/ruby/gems/1.8/doc/uuid-2.0.0/rdoc/index.html
#uuid = UUID.new
# 10.times do
#   p uuid.generate
# end
```

From:

<http://wiki.mro.name/> - □

Permanent link:

http://wiki.mro.name/playground/mail_filter_rulesLast update: **2008/12/21 19:49**

