Redmineでも日本語全文検索

～　OSSでも最小限の労力で　～

町井　昌徳（まちいまさのり）@vegashrine
豊福　親信（とよふくちかのぶ）@nobu_toyofuku

2012年1月21日

ちょっと前置きを

- 私はRedmineをアジャイル開発専用のツールとは考えていない.
- チケット駆動型のサポートツールとして，機能仕様はピッタリこないが，その設計コンセプトは運用業務にも使えると思っている.

Redmineへ統合の是非

- 既存のシステムもあるのに，文書管理をRedmineへ統合するのはナゼ？

- 何かしらの統合は必要である．（論を待たない）
- 実はOSS版ITILツールとして，なかなかいい線をいっている．既存システムを置き換えるポテンシャルがある．

ITILって何？

なぜRedmineがいいの？

- すべてはサービスであり，プロセスとしてそのライフサイクルが管理される.

- 商用のITILツールもチケット駆動型，しかも機能てんこ盛り！でも技術が旧くて，拡張の柔軟性に欠ける.

（オプションを追加して統合化とは，まさに殿様仕様）

…というわけで運用現場へ導入したら…

- 「状況の共有」に大きな進展あり！

  （チーム全員がチーム内の状況をくまなく把握する方向へ収束した）

- さらに欲が出て，「情報の共有」にも使いたい！

- 結局，Redmineへ統合化するハメに．

Redmineの文書管理は変？

- 文書，ファイル，リポジトリ…って，よくわらん！

  （redmine.jpの説明を読んでも使い分けがイマイチわかりません）

- そこで一元化するプラグインDMSFが開発されたらしい.

- でも，検索エンジンのXapianは日本語には対応しないという，悲しい現実.

- Hyper Estraierという有名な検索エンジンがDMSFでも利用できるか？

- コードの修正は意外と簡単，ありがとう！＞DMSFの作者さん

...しかし根本的な解決ではない

- DMSFのバージョンアップに（永遠に）追随しなければならない.

- 多バイト言語の文書管理を新たに考えなくては...（今後の課題）

DMSFの修正方針

- できるだけ少々の変更で
- でも構造（アルゴリズム）を変えたくない
- でも可読性を著しく下げたくない

- 各人各様の修正があり得るので，今日は参考情報とする.

それよりもっと厄介な問題が...

- MS-Office等のファイルを対象にした，日本語平文を抽出する機能について，Linux上のOSSのパッケージというものが存在しない.

  （もちろん商用はあるし，Windows版ならばある）

- 当該環境をOOSで構築するのが一苦労.

  各種ライブラリのバージョンを調整するところで十人中十人が躓くハズ.

テキスト抽出に２つの道が

- OpenOfficeサーバーに抽出させる.

  でもOpenOfiiceのバージョンがどんどん上がっていくし，将来性が…

- 既に枯れた各種コマンドを利用する.

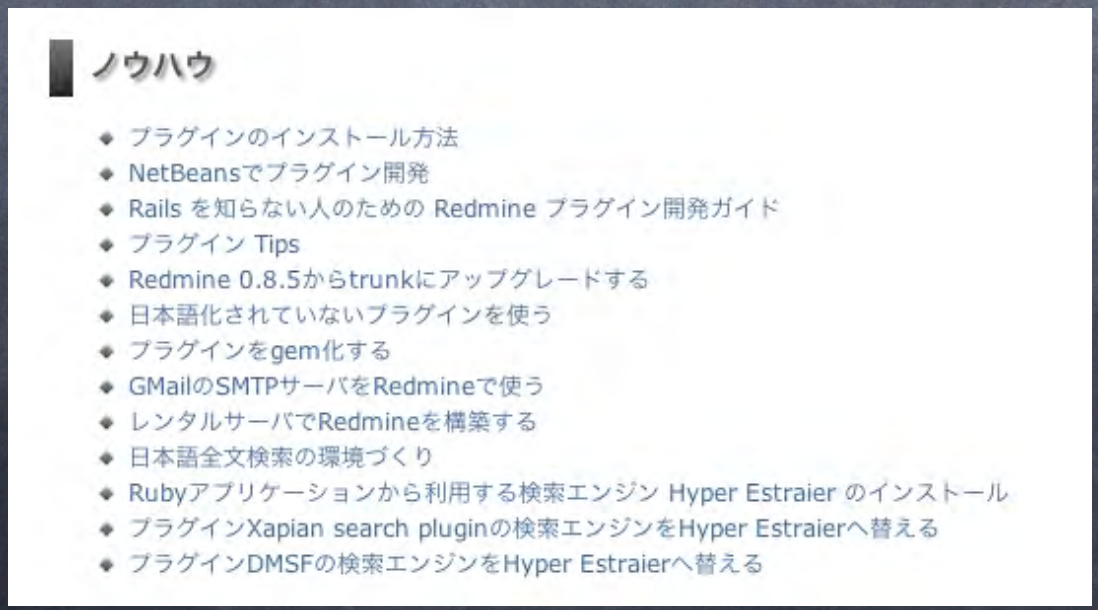
- 今回は「枯れたコマンド」という安全パイを選んだ.

ライブラリを調整する方針

- 全てをソースコードからビルドする！

  osが旧くなるほど試行錯誤が多くなり，苦労が多い．

- r-labsのwiki（ノウハウ）へ書き留めたので，読んで下さい．（http://www.r-labs.org/projects/r-labs/wiki/）（CentOS 5.6を想定している）



ノウハウ

- プラグインのインストール方法
- NetBeansでプラグイン開発
- Rails を知らない人のための Redmine プラグイン開発ガイド
- プラグイン Tips
- Redmine 0.8.5からtrunkにアップグレードする
- 日本語化されていないプラグインを使う
- プラグインをgem化する
- GMailのSMTPサーバをRedmineで使う
- レンタルサーバでRedmineを構築する
- 日本語全文検索の環境づくり
- Rubyアプリケーションから利用する検索エンジン Hyper Estraier のインストール
- プラグインXapian search pluginの検索エンジンをHyper Estraierへ替える
- プラグインDMSFの検索エンジンをHyper Estraierへ替える

取り組む要点をまとめると...

- ファイルから文字列を抽出する環境づくり
- Hyper Estraierのインストール
- DMSFのコードを修正
- インデックス作成を仕掛ける. （cron等）

文字列抽出の環境づくり

- PDF
  GNU PDFプロジェクトの成果を取り入れる.
- オフィス文書
  GNOMEプロジェクトのうち，GNOME Officeの成果を取り入れる.
  （OSが旧い場合は，ここが一番難易度が高いかも）
- PowerPoint
  基本的にxlhtmlプロジェクトの成果を取り入れるが，安定させるためには，修正パッチが必要. （修正コードは日本の方による）

ビルドのポイント

- パッケージの管理を徹底（Paco を利用）
- インストール場所の徹底.　（/usr/local）
- 環境変数やコンパイル・オプションの適切な設定
  （PKG_CONFIG_PATH や LD_LIBRARY_PATH，--disable-xxx）
- ライブラリ等は安定版を.　（新しすぎない）

Paco（インストールしたライブラリの管理）

- Wikiに書かなかったことが...

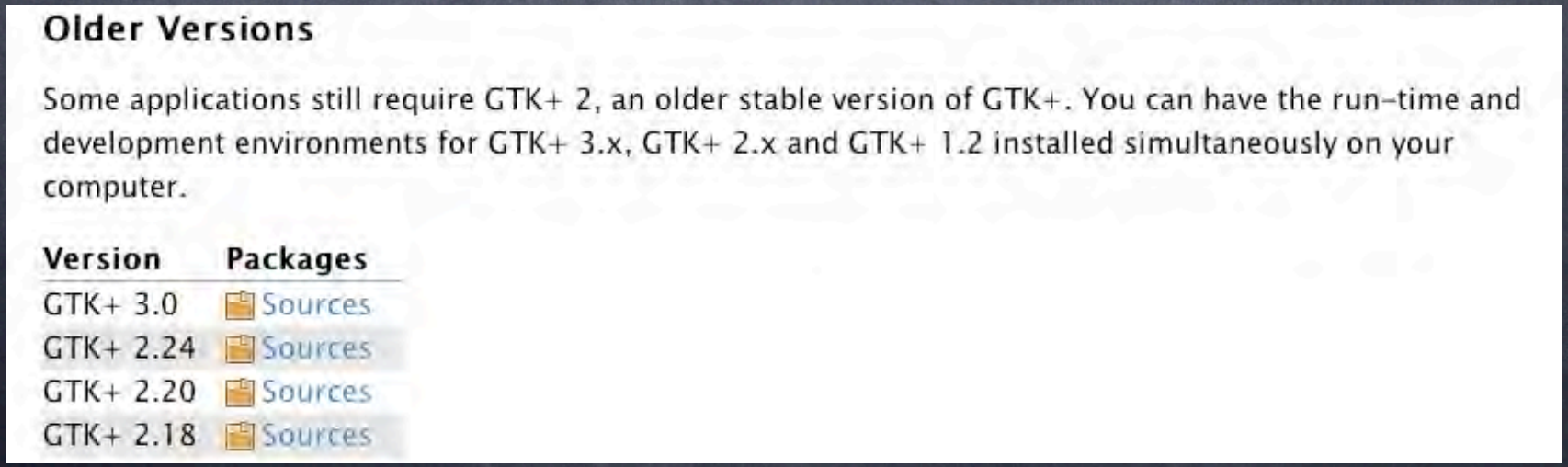
```
$ pwd
/usr/local/src/paco-2.0.9
$ env
      .
      .
      .
PKG_CONFIG_PATH=/usr/local/lib/pkgconfig:/usr/lib/pkgconfig
LD_LIBRARY_PATH=/usr/local/lib:/usr/lib
      .
      .
      .
$ ./configure --prefix=/usr/local --dsiable-gpaco
$ make
$ sudo make install
```

Paco（インストールしたライブラリの管理）

```
$ pwd
/usr/home/taro/fontconfig-2.5.91
$ env PKG_CONFIG_PATH=/usr/local/lib/pkgconfig:/usr/lib/pkgconfig  ./configure --prefix=/usr/local
$ make
$ sudo paco -D make install
$
$ paco -a
fontconfig-2.5.91
      .
      .
      .
$ paco -f fontconfig-2.5.91
fontconfig-2.5.91:
/usr/local/bin/fc-cache
/usr/local/bin/fc-cat
/usr/local/bin/fc-list
/usr/local/bin/fc-match
/usr/local/etc/fonts/conf.avail/10-autohint.conf
/usr/local/etc/fonts/conf.avail/10-no-sub-pixel.conf
      .
      .
      .
```

OSやライブラリのバージョンに左右されるビルド

- r-labsのWikiへ記述したものは，CentOS 5.6以上の例である．
- CentOS 5.6より旧いと，試行錯誤がさらに増えることを覚悟せよ．
- GTK+ライブラリは中心的存在．（http://www.gtk.org/download/linux.php）
  必要最小限のインストールが事実上，不可能．
  GTK+ 3.0は他ライブラリとのすり合わせが難しいので，2.xへ下げる．



Older Versions

Some applications still require GTK+ 2, an older stable version of GTK+. You can have the run-time and development environments for GTK+ 3.x, GTK+ 2.x and GTK+ 1.2 installed simultaneously on your computer.

Version	Packages
GTK+ 3.0	Sources
GTK+ 2.24	Sources
GTK+ 2.20	Sources
GTK+ 2.18	Sources

GTK+ 2.xの選択肢

- もしOSのバージョンが旧かったら... GTK+ 2.14 まで下げる.

```
$ cat /etc/redhat-release
CentOS release 4.7 (Final)
$ pwd
/usr/local/src/gtk+-2.24.8
$ ./configure --prefix=/usr/local
     .
     .
     .
configure: error: Package requirements (glib-2.0 >= 2.27.3    atk >= 1.29.2    pango >= 1.20    cairo >= 1.6
gdk-pixbuf-2.0 >= 2.21.0) were not met:

Requested 'glib-2.0 >= 2.27.3' but version of GLib is 2.4.7
Requested 'atk >= 1.29.2' but version of Atk is 1.8.0
Requested 'pango >= 1.20' but version of Pango is 1.6.0
No package 'cairo' found
Requested 'gdk-pixbuf-2.0 >= 2.21.0' but version of GdkPixbuf is 2.4.13
     .
     .
     .
$ $ /lib/libc.so*
GNU C Library stable release version 2.3.4, by Roland McGrath et al.
Copyright (C) 2005 Free Software Foundation, Inc.
```

GNOMEビルドのヒント

- yumはダメといったが，"*-devel" のインストールには都合が良いので，最初にインストールリストを調べ，必要に応じでインストールしておくほうが良い.
- 実はインストールの順番に意味があったりする.
  特にcairoとpangoの関係

```
$ find  /usr/local/src -name pangocairo.h -print
/usr/local/src/pango-1.17.5/pango/pangocairo.h
```

- configure 時にグラフィクス関連ライブラリが不要と指定する.
  【例】http://developer.gnome.org/gtk3/stable/gtk-building.html

```
$ grep "¥--disable" configure
```

```
$ ./configure --prefix=/usr/local --disable-jpeg
```

- インストール後にライブラリの依存関係をチェックする.

```
$ grep "/usr/lib" /usr/local/lib/*.la | grep dependency_libs
```

CentOS 4.7でビルドした例（1/2）

- さらなる環境変数

  CAIRO_BACKEND_CFLAGS=/usr/local/include/cairo
  CAIRO_BACKEND_LIBS=/usr/local/lib
  FREETYPE_CONFIG=/usr/local/bin/freetype-config

- bashのバージョンアップ（goffice）

  /usr/local/bin/bash-3.1

- インクルード・ヘッダファイルの追加（AbiWord）

  ev_UnixKeyboard.cpp 内の適当な位置へ #include <X11/Xkeysym.h> を追加

- ヘッダファイルを他環境から調達（gdk-pixbuf-csourceがうまく動作しない時の対処療法）

  gnumeric-1.10.16/src/pixmaps/gnumeric-stock-pixbufs.h

- アセンブラ・コードの修正（gdk-pixbuf）

  行先頭に "//" を挿入してコメンアウトするのが２ファイル計４箇所

CentOS 4.7でビルドした例（2/E）

```
$ paco -a
abiword-2.8.6        gdk-pixbuf-2.22.1    libIDL-0.8.14        poppler-0.18.3
atk-1.26.0           gettext-0.18.1.1     libpng-1.0.51        tiff-3.9.2
bash-3.1             glib-2.28.8          libpng-1.5.4         wv-1.2.4
cairo-1.4.12         gnumeric-1.10.16     librsvg-2.34.0       xlhtml
cairo-1.6.4          goffice-0.8.17       libtool-2.4.2        XML-Parser-2.41
fontconfig-2.5.91    gtk+-2.14.7          ORBit2-2.14.19       xz-5.0.3
freetype-2.3.5       intltool-0.40.3      pango-1.17.5         zlib-1.2.5
fribidi-0.19.2       jpeg-8d              pango-1.20.5
GConf-2.4.0.1        libffi-3.0.10        pixman-0.19.2
gdk-pixbuf-2.21.7    libgsf-1.14.22       pkg-config-0.26
```

```
$ yum list installed libxml2* libglade2*
Installed Packages
libglade2.i386                2.4.0-5              installed
libglade2-devel.i386          2.4.0-5              installed
libxml2.i386                  2.6.16-12.6          installed
libxml2-devel.i386            2.6.16-12.6          installed
```

できるだけ上位のバージョンにする「努力」はあまりしていない.

それでも疲れた...

平文抽出とインデックス作成（1/2）

- 平文抽出の例

```
#! /bin/sh

case "$1" in
*.pdf)
        pdftotext "$1" "$2"
        ;;
*.xls)
        tmpdir=`mktemp -d /tmp/estfilter.XXXXXXXX`
        ssconvert -S --export-type Gnumeric_stf:stf_csv "$1" $tmpdir/%n 2>&1 | egrep -v '(^$|^MISSING
anchor for obj |: EXCEL: unhandled excel object of type MS Drawing )'
        cat $tmpdir/* > "$2"
        /bin/rm -rf $tmpdir
        ;;
*.doc)
        # wvWare --charset=UTF-8 --nographics $1 > $2

        abiword --to txt --to-name $2 $1
        ;;
esac
```

平文抽出とインデックス作成（2/E）

```
#! /bin/sh

case "$1" in
*.ppt)
      ppthtml $1 > $2
      ;;
esac
```

Hyper EstraierがHTML対象に検索できることを利用している.

- インデックス作成の例

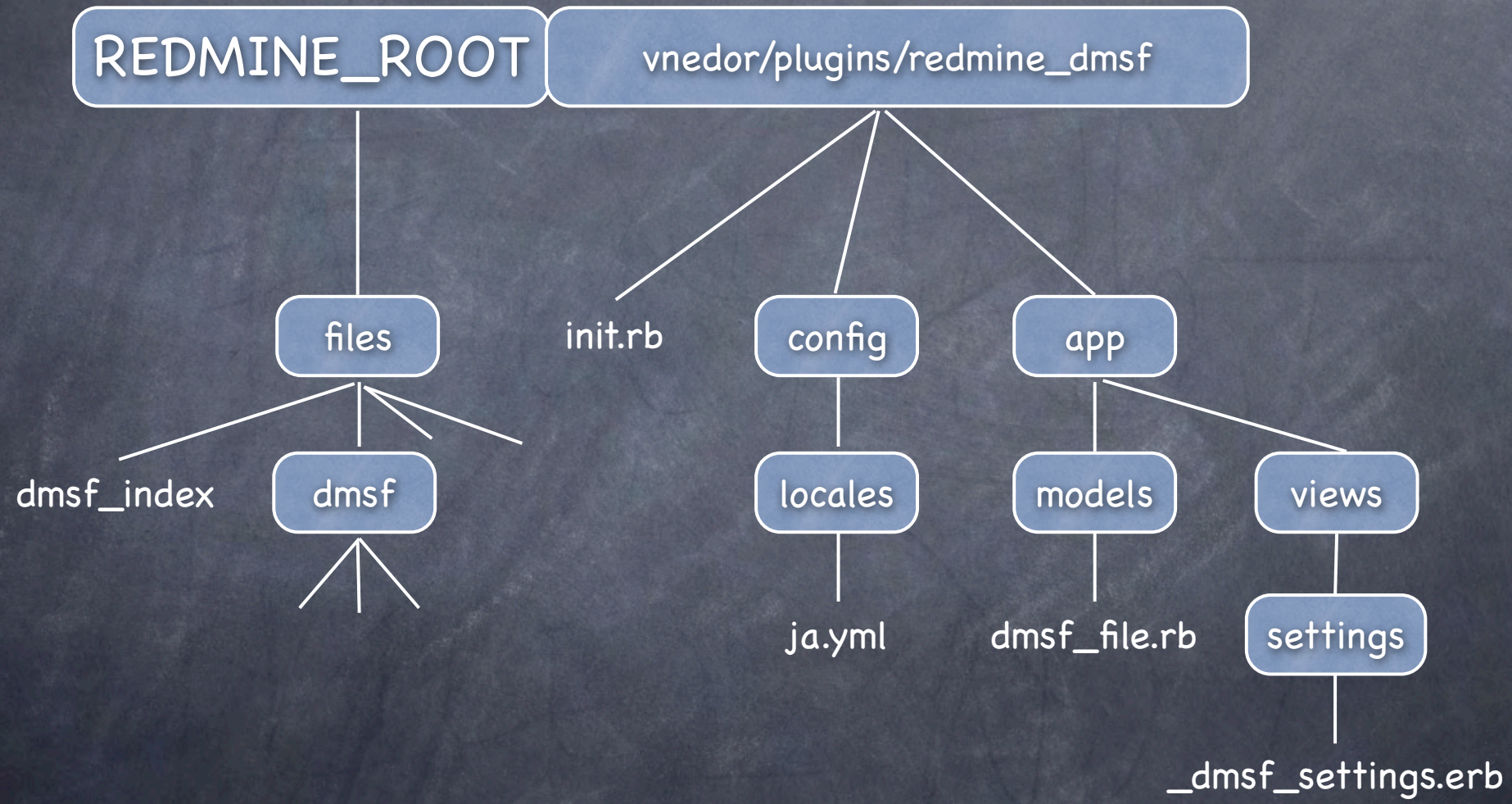
```
#! /bin/sh

echo
date
/usr/local/bin/estcmd gather -il ja -fx .pdf,.xls,.doc T@estfiltert.sh -fx .ppt H@estfilterh.sh -pc utf-8 -lf
-1 -lt -1 $*
```

DMSFとXapian search plugin

- 添付ファイルの全文検索プラグインとどちらが簡単か？
- DMSFの方が悩みが少ない．（後述）

Redmineのディレクトリ構造



config/locales/ja.ymlの修正

- コード中のラベルも含めて，Xapian から Estraier へと修正する．

```
@@ -156,7 +156,7 @@
    :error_file_storage_directory_does_not_exist: "ファイル保存フォルダが存在せず作ることもできません"
    :error_file_can_not_be_created: "ファイルを保存フォルダに作ることができません"
    :error_wrong_zip_encoding: "Zip エンコーディングが正しくありません"
-   :warning_xapian_not_available: "Xapian が利用できる状態になっていません"
+   :warning_estraier_not_available: "Hyper Estraier が利用できる状態になっていません"
    :menu_dmsf: "DMSF"
    :label_physical_file_delete: "物理ファイルの削除"
    :user_is_not_project_member: "あなたはプロジェクトのメンバーではありません"
```

init.rbの修正

- 日本語対応である意味をバージョン番号へ与える.

```
@@ -29,7 +29,7 @@
  name "DMSF"
  author "Vít Jonáš"
  description "Document Management System Features"
-  version "1.2.1"
+  version "1.2.1-JP"
  url "http://code.google.com/p/redmine-dmsf/"
  author_url "mailto:vit.jonas@gmail.com"
```

app/views/settings/_dmsf_settings.erbの修正

- ロードする（require する）ライブラリを xapian から estraier へ変更する.

```
@@ -75,22 +75,22 @@

 <hr />
 <% begin
-      require 'xapian'
-      xapian_disabled = false
+       require 'estraier'
+       estraier_disabled = false
     rescue LoadError    %>
-    <p class="warning"><%= l(:warning_xapian_not_available) %></p>
-<%      xapian_disabled = true
+    <p class="warning"><%= l(:warning_estraier_not_available) %></p>
+<%      estraier_disabled = true
     end %>
```

app/models/dmsf_file.rbの修正

- 検索機能本体の修正

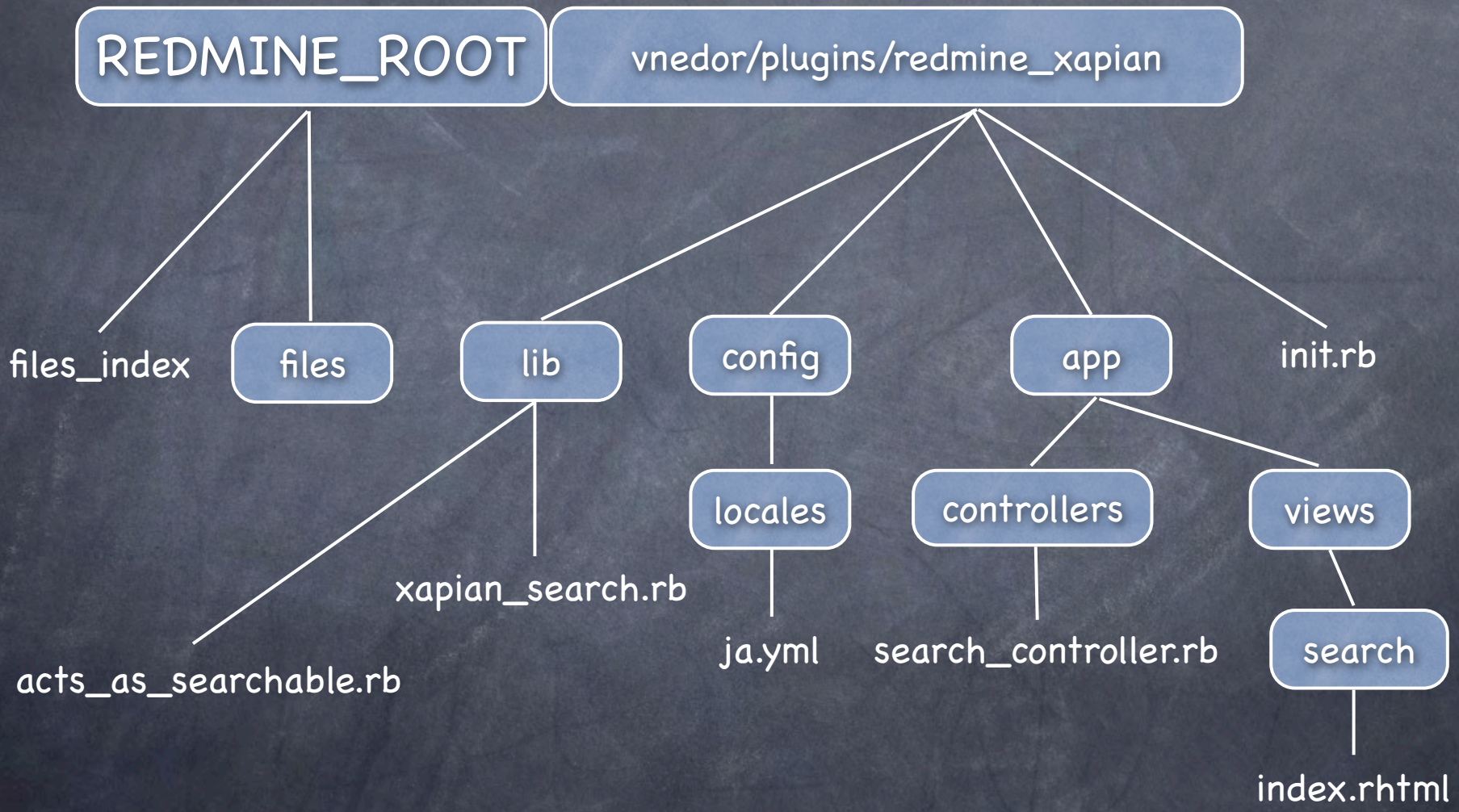
  database.nil 以下は大幅に構造が異なるので，ざっくりと入れ替える．

```
unless database.nil?
# create a search condition object
cond = Estraier::Condition::new
# set the search phrase to the search condition object
queryString = tokens.join(options[:all_words] ? ' AND ': ' OR ')
cond.set_phrase(queryString )
# get the result of search
result = database.search(cond)
if result
     # for each document in the result
    dnum = result.doc_num
    for i in 0...dnum
    # retrieve the document object
       doc = database.get_doc(result.get_doc_id(i), 0)
       next unless doc
       # display attributes
       uri = doc.attr("@uri")
       if uri
          filename = uri.sub(/.*\//, '')
```

Xapian search pluginでは？

- そもそも名前からして悩みどころだ！
  それでも「修正」に留めねば.
- ポリシーを r-labs へ（いっぱい）書き留めたが，トリッキーだったかもしれない.
  （悩み多し）

Redmineのディレクトリ構造



config/locales/ja.ymlを追加

- 英語メニューをベースにして，日本語メニューを作成する.

```
label_enable_redmine_xapian: "hyper estraier による添付ファイル検索を可能にする"
label_index_database: "hyper estraier 検索インデックスディレクトリ"
                    .
                    .
                    .
label_document: "ドキュメント"
                    .
                    .
                    .
label_stemming_lang: "Stemming Language（注：estraier版では未使用）"
label_enable_xapian_on_search: "検索画面でhyper estraier使用可"
                    .
                    .
                    .
label_database_error: "検索インデックスエラー。Hyper Estraier利用環境を構築してください。"
```

init.rbの修正

- ライブラリ名称を xapian から estraier に変更する．警告内の文字列も同様に変更する．
- 日本語対応である意味をバージョン番号へ加える．

```
@ -7,10 +7,10 @@

begin
-    require 'xapian'
+    require 'estraier'
    $xapian_bindings_available = true
rescue LoadError
-    Rails.logger.info "REDMAIN_XAPIAN ERROR: No Ruby bindings for Xapian installed !!.
PLEASE install Xapian search engine interface for Ruby."
+    Rails.logger.info "REDMAIN_XAPIAN ERROR: No Ruby bindings for Hyper Estraier installed !!.
PLEASE install Hyper Estraier search engine interface for Ruby."
                .
                .
                .

-       version '1.2.1'
+       version '1.2.1-JP'
```

app/controllers/search_controller.rbの修正

- 検索エンジン estraier が利用できない場合の警告メッセージを追加する.

```
@@ -96,6 +96,7 @@
     end

     end
+      flash[:warning] = "warning: #{l(:label_database_error)}" unless @titles_only ||
$xapian_bindings_available
     @results = @results.sort {|a,b| b.event_datetime <=> a.event_datetime}
     if params[:previous].nil?
       @pagination_previous_date = @results[0].event_datetime if offset && @results[0]
```

app/views/search/index.rhtmlの修正

- 検索結果画面に表示される Stem等の xapian 関する設定部分を表示させない.

```
@@ -16,6 +16,7 @@
<% end %>
</p>
<% logger.debug "DEBUG: object_types from search: " + Redmine::Search.available_search_types.inspect %>
+<% if false then %>

    <% Setting.plugin_redmine_xapian['stem_langs'].push(Setting.plugin_redmine_xapian['stemming_lang'])
    unless Setting.plugin_redmine_xapian['stem_langs'].include?(Setting.plugin_redmine_xapian
    ['stemming_lang']) %>

    <p>
@@ -31,6 +32,7 @@

    <%end%>
+<%end%>
<p><%= submit_tag l(:button_submit), :name => 'submit' %></p>
<% end %>
</div>
```

lib/acts_as_searchable.rbの修正

- estraier 無しの場合でも，タイトルの検索を可能にさせる.
- ファイルが添付されているチケットやwiki，それぞれの閲覧権限に従って検索させる. （そもそもこれはオリジナルの改善である）
- 最新バージョンは Redmine 1.3 へ対応しているというが，疑わしい.

  REDMINE_ROOT/vendor/plugins/acts_as_searchable を「上書き」する格好になっている.

- コードの詳細については r-labs を読んで下さい.

lib/xapian_search.rbの修正

- xapian と言いながら中身は estraier を利用する.

```
@@ -11,69 +11,56 @@
            Rails.logger.debug "DEBUG: user_stem_lang: " + user_stem_lang.inspect
            Rails.logger.debug "DEBUG: user_stem_strategy: " + user_stem_strategy.inspect
      Rails.logger.debug "DEBUG: databasepath: " + getDatabasePath(user_stem_lang)
-      databasepath = getDatabasePath(user_stem_lang)
+      databasepath = getDatabasePath('')

      begin
-        database = Xapian::Database.new(databasepath)
+        database = Estraier::Database::new
+        unless database.open(databasepath, Estraier::Database::DBREADER)
+          return [xpattachments,0]
+        end
      rescue => error
        raise databasepath
-        return [xpattachments,0]
          end

      # Start an enquire session.

-      enquire = Xapian::Enquire.new(database)
+      enquire = Estraier::Condition::new
+
                  .
                  .
```

感想

- 今後の計画はまだ何もありませんm(_ _;)m
- 少し遅くてもいいから，インデックス無しの検索機能が使えたら，楽だったのだが．特に小規模の場合は．
- この方式は大規模化やセキュリティの点で課題あり．
- 今はLotus NotesからRedmineへ移行させることに注力しており，全文検索機能はそこでも役立っている．
- 【個人的】Mac OS XにはSpotlightのコマンド版があるので，Mac OS X上の私家版Redmineで試してみたい…（というかAndroidのサーバー版って無いの？）