

# 日本語変換エンジンAnthyと 入力モジュールUtena(仮)

田畑悠介  
富士通(今日は無関係)  
KMC(すでにOB)  
IPA認定スーパークリエイター  
(過去の栄光)

# 注意事項

- ここで述べるのは田畑の意見です
- 本プロジェクトで作成したソフトウェアの一部はIPA(情報処理振興事業協会)の未踏ソフトウェア創造事業の(主に)資金的援助を受けました
- 悪ノリdriven
- Just for fun!

# 発表の構成

- 現状の紹介
- プロジェクトの紹介
- かな漢字変換エンジンAnthy
- 汎用入力システムUtena(仮)
- デモandディスカッション

# 現状の紹介(歴史)

- 90年代前半まで
  - 国内の商用UnixベンダによってWnn、Cannaなどの開発が行われる
- Linuxへの対応作業が行われる
- 商用のシステムが登場する
- -----終了-----

# いい加減な年表

デスクトップ	TWM, FVWM	KDE	GNOME	mozilla	OpenOffice
emacs	mule(emacs-19.x)	emacs-20	emacs-21		
日本語入力	Wnn4.2	Canna3.5b2	SKK	kinput2	

# 入力システムの問題点

- 日本語の問題
  - 放置プレーになっている
  - プロジェクト参加への敷居が高い
  - 開発はそもそも難しい
- 国際化の問題
  - 複雑な言語と簡単な言語がある
  - 必ずしも一般化できるものではない
  - 過去の地域化、不適切な一般化

# フリー？

- 商用のシステム
  - 日本人だけ不自由する
  - 日本語だけ不自由する
  - ユーザの数だけライセンス
  - 基本的には企業は何をするのも勝手
    - 特定アーキテクチャのみへの供給
    - 特定ディストリビューションへの提供
      - バージョン固定
      - ABI固定
  - いつでも開発中止

# 真の問題点

- 誰も問題だと思っていない
  - 大企業の持って来たものは正しい
  - 古くから有るものは正しい
  - 誰も開発していないことを無視
    - この状態ではオープンソースvsクローズどころではない
    - (バザールvs伽藍でもない)
- 日本語の問題は日本人しか解決できない
  - 他の問題は放置していても外国で解決される;)



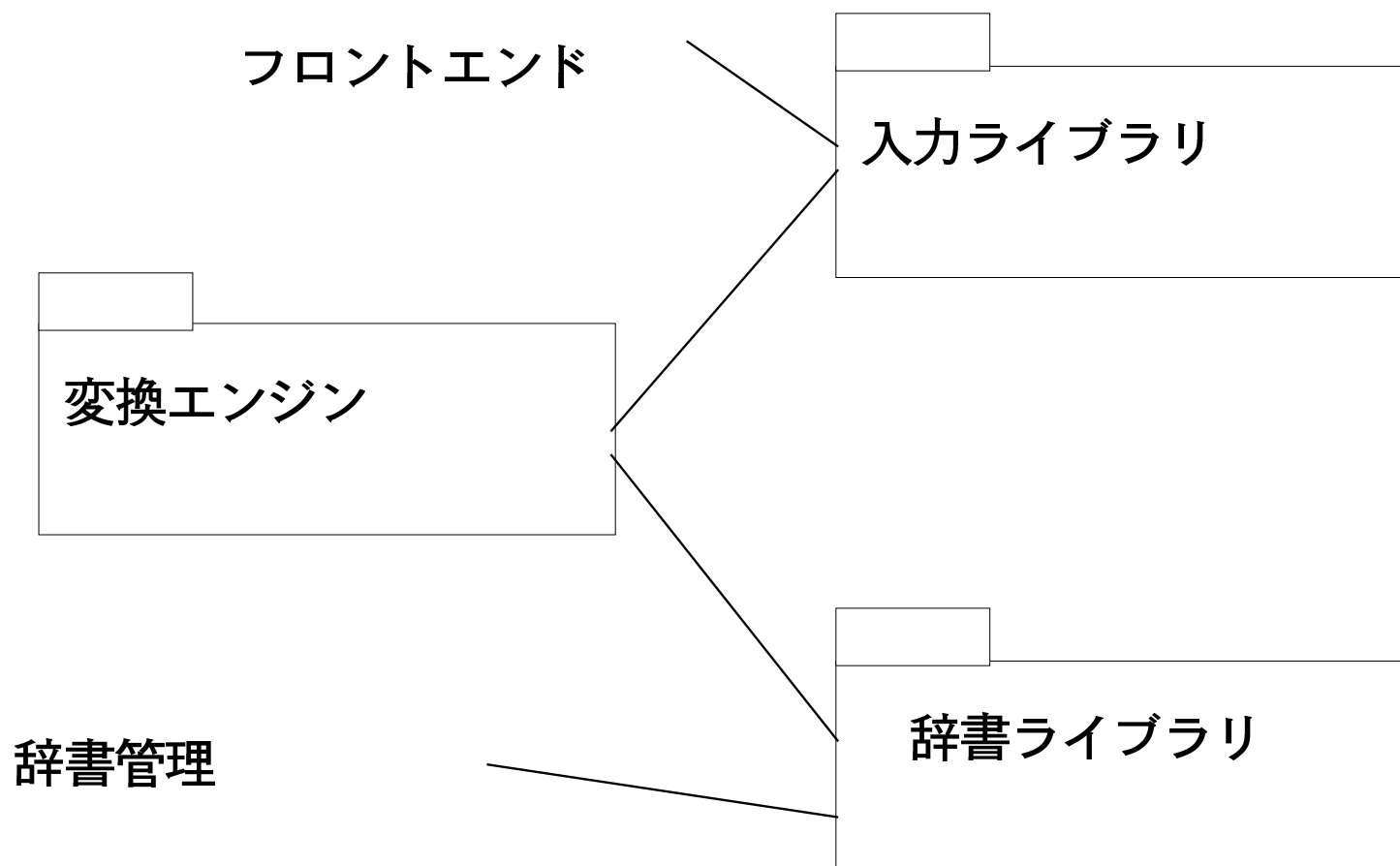
# かな漢字変換エンジンAnthy

- 開発の経緯
  - Wn, CannaのAPIが理解できずに困る
  - 3か月ほど妄想する
  - 平均2時間\*3か月コーディングして動作する
  - アルゴリズムの改良
  - Emacs対応
  - 単語登録
- 特徴
  - シングルユーザ
  - ライブラリとして実装

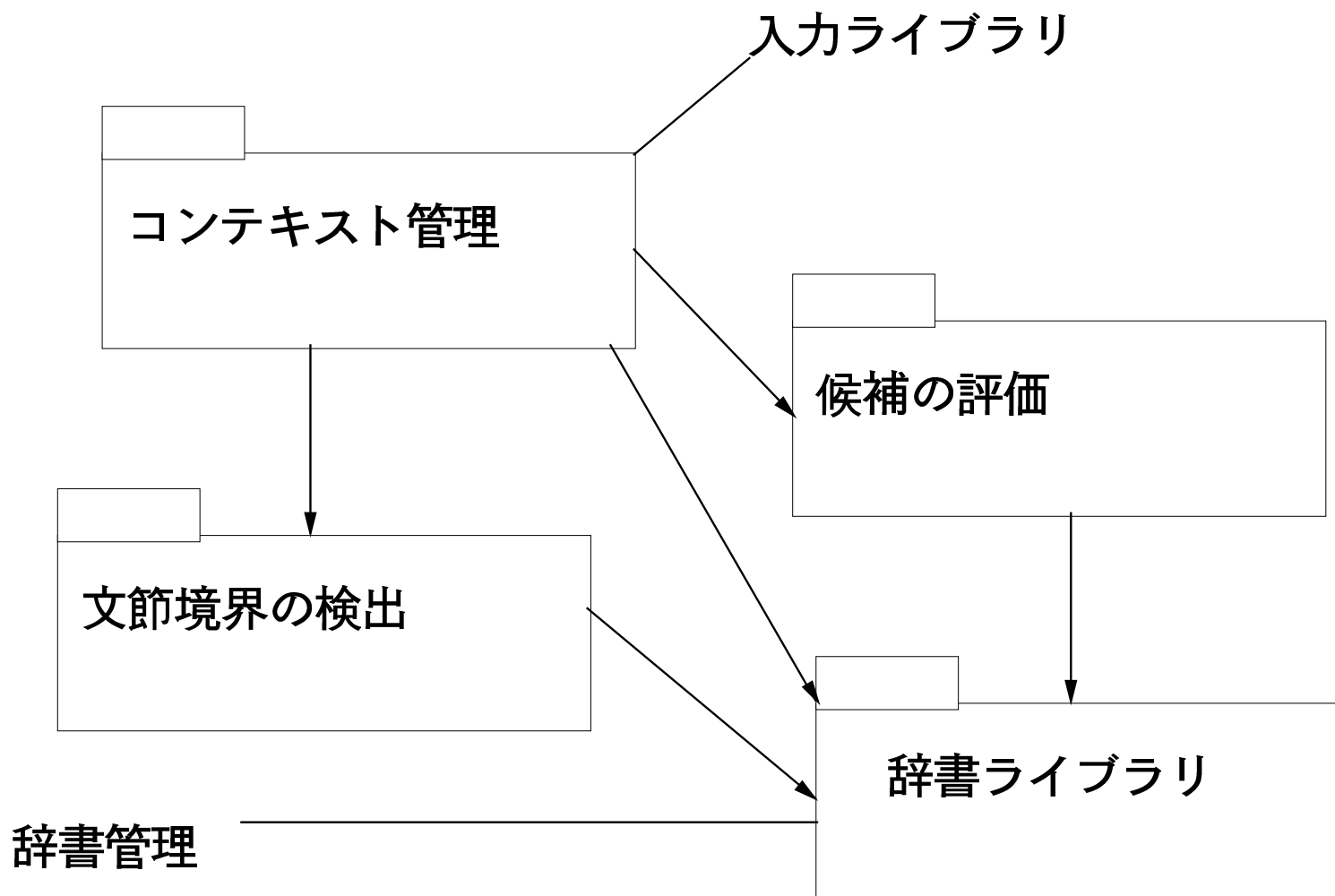
# 現在までの展開

- Linux Zaurus用 imkit-anthy
- EglによるAnthy対応

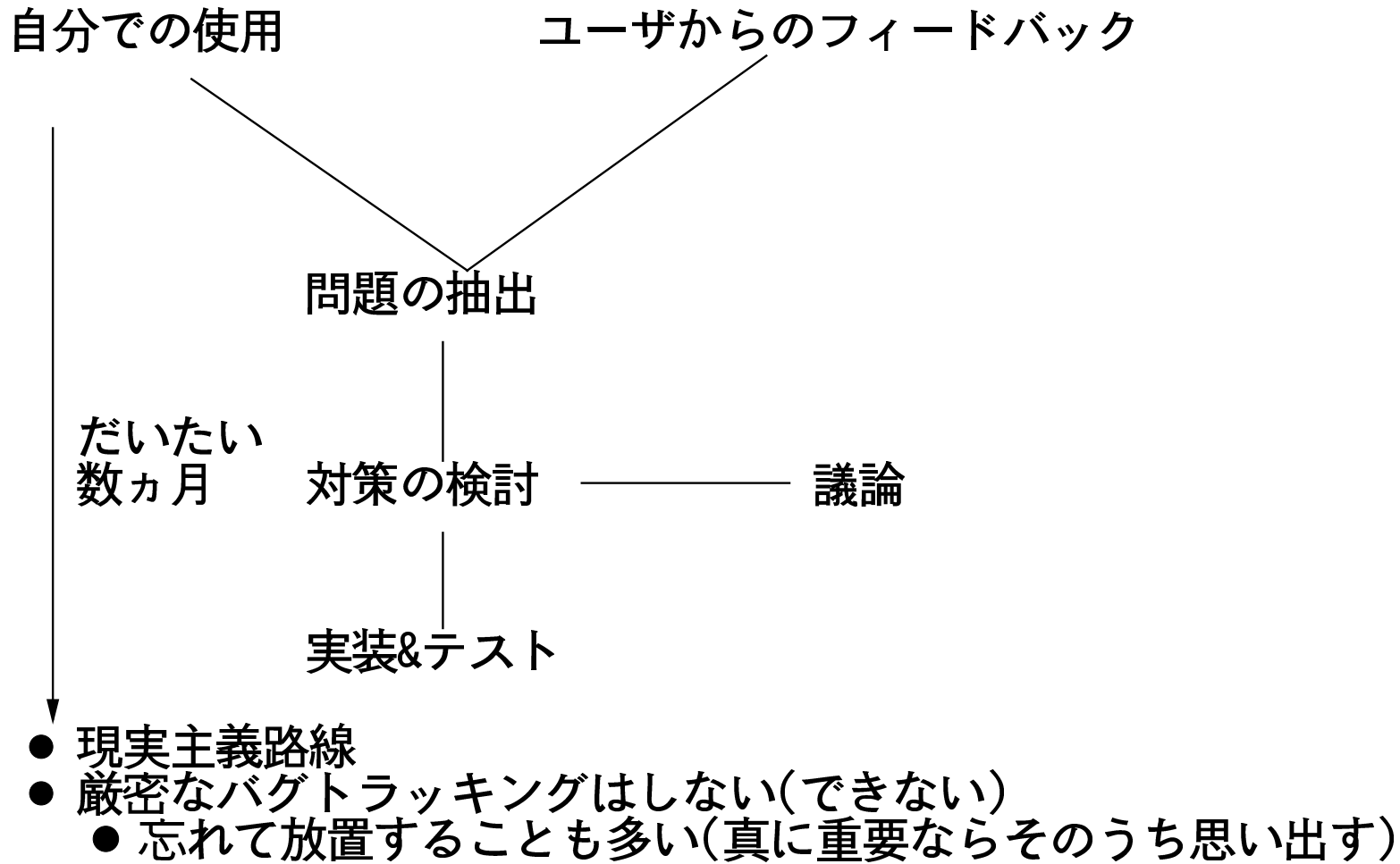
# かな漢字変換エンジンAnthy



# かな漢字変換エンジンAnthy



# 開発プロセス



# 次の課題

- 国際化
- 強力なカスタマイズ機能
- アプリケーションへの組み込みの容易化
- 辞書開発の枠組
- 性能の向上など
  - ユニコード化
  - 柔軟なカスタマイズ
  - 気の効いた学習
  - 候補の評価、文節境界の検出

# 真の課題

- この分野にリソースが必要であることを理解してもらう
  - ここ(LC)できちんと説明
  - 某誌の原稿を早く書く
- Anthyの開発に参加しやすくする
  - コメントを書く
  - MLでまめに説明する
- 新しい変換エンジンを作りやすくする
  - ?

# 入力の枠組

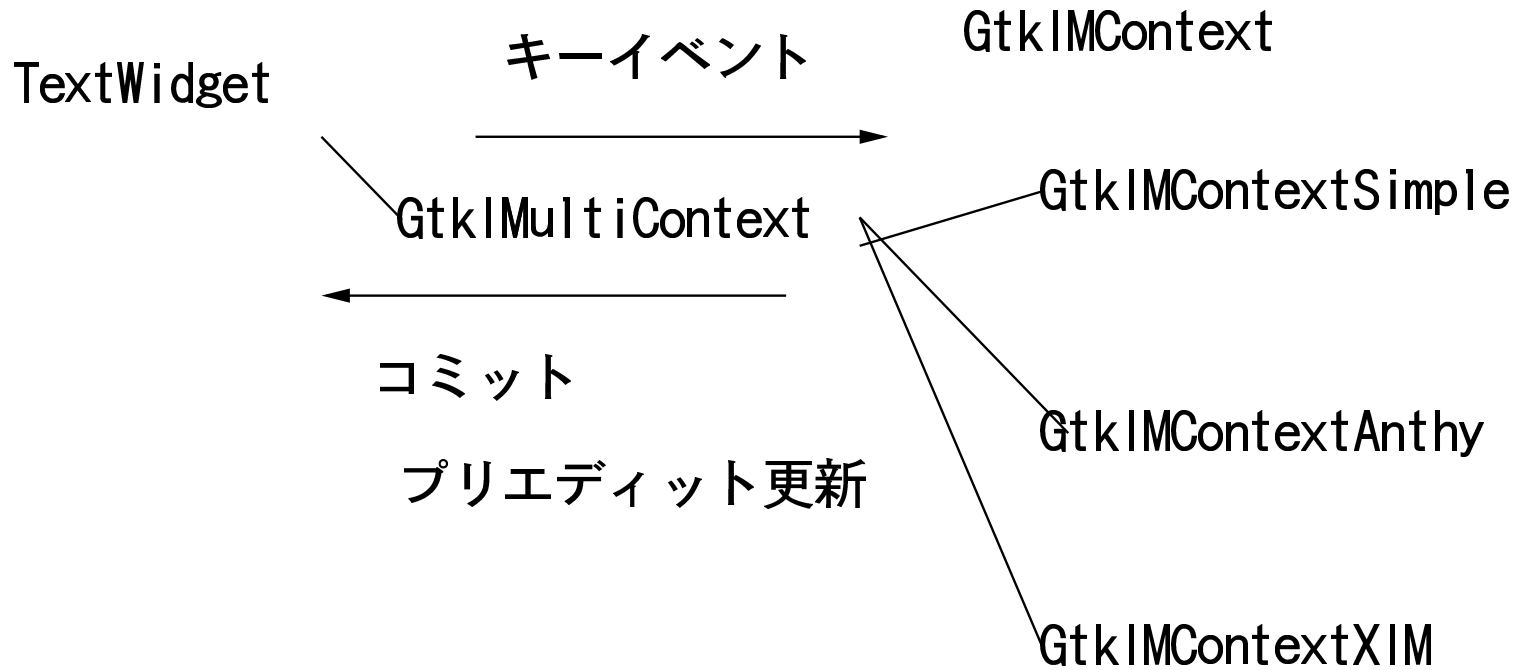
- leim
  - 多くの言語に対応→世界の財産
  - emacsのみ
- XIM
  - Xへの依存
- IIIMF
  - 稚拙な設計
  - XML
  - ユニコードのみ
- gtk+-immodule
  - gtk+-2.0.x
  - ユニコードのみ



# Client/Server System Considered Harmful

- シンプルな言語
  - 通信のオーバヘッド
- 複雑な言語
  - 認証
  - 通信の暗号化
  - 非同期イベント
  - 変換システム
  - 単語登録などのデータの操作
  - 変換エンジンの実装への負担
- 世迷い言
  - 「Webなどと同じインターネットやイントラネットでのサービスの一つとして位置付ける」

# gtk+-immodule



- オブジェクト指向による簡単な構造
- シグナルによる柔軟なメッセージング

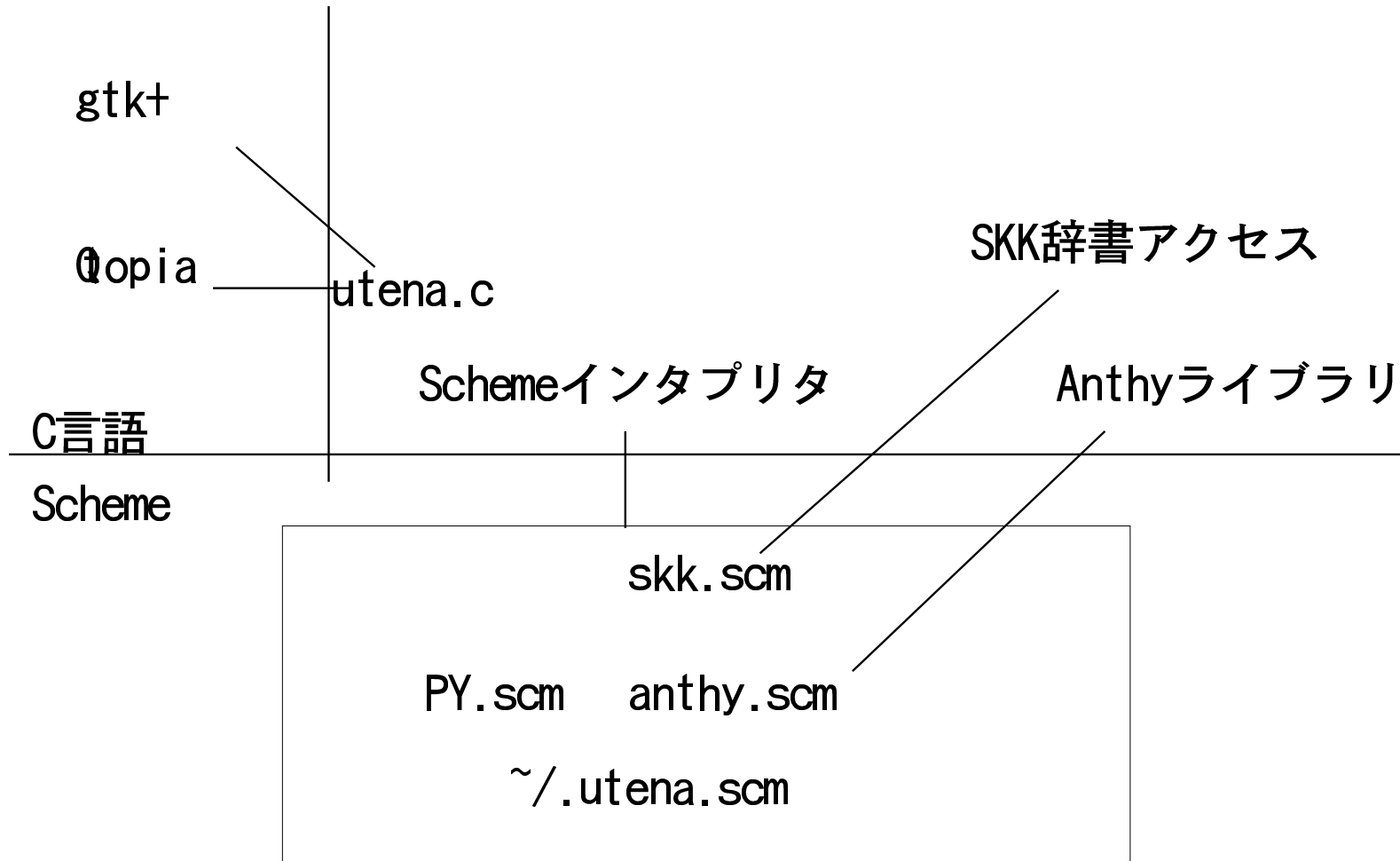
# 独自の入力システム

- Qt
  - QLineEditというクラスを導入
- Qtopia
  - 何故かQtより進んでいる？
- VIM
  - im\_custom

# Utena(名称未定)

- Schemelによる記述をベース
  - 簡単な入力システムは簡単に
  - 複雑なものもそれなりに
- 簡単に使えるライブラリ形式
  - 開発者がその場でテスト
  - いちいちサーバを止めなくてもよい
  - 自分の権限で動作する
- エンドユーザによるカスタマイズ
  - 変態的カスタマイズ(全面書き換えなど)も可能

# Utena(仮)



# Utena(仮)

- 構成
  - /usr/share/utena/\*.scm
  - /usr/include/utena/utena.h
  - /usr/lib/libutena.so\*
  - ~/.utena
- 使いかた
  - #include <utena/utena.h>
  - -lutena

# Utena(仮)

- シェアードライブラリによる機能の追加
  - イレギュラーな機能
    - SKKの再帰学習
    - 単語登録

```
init_plugin()  
{  
    provide("skk-recursive-win");  
  
    init_subr("skk-recursive-win-init", skk_recursive_win_in  
it_gtk);  
    ..  
}  
{if (feature? 'skk-recursive-win)  
    (skk-recursive-win-init))
```

# ロードマップ

- leimからなるべく多くを流用する
- Anthyのフロントエンドを実装
  - カナ入力やその他の入力システムも容易
- SKKの実装
- Qopia(Zaurus)、Qtに対応



# 最後に

- 現状を直視しましょう
- 入力システムにかかわる人が増えることを期待しています
  - 開発に参加する
  - コードフォークする
  - レガシーをサルベージする