

# Ruby/Tk は本当にダメな子なのか？



永井 秀利 ([nagai@ai.kyutech.ac.jp](mailto:nagai@ai.kyutech.ac.jp))

九州工業大学 / Rubyist 九州



## Ruby/Tk とは

- Ruby 用 GUI ライブラリの一つ
  - 現状, Ruby の GUI ライブラリ中, 唯一の標準添付
  - お手軽 GUI プログラミングで定評のある Tcl/Tk の wrapper



## Ruby/Tk に対する批判，非難

- 見掛けが悪い
- 機能が貧弱
- ウィジェットの種類が少ない
- 開発環境，GUI ビルダがない
- Tk はオブジェクト指向じゃない
- Tk は古くて時代遅れ
- ドキュメント不足
- とにかく Tk だから嫌（Ruby/Tk は使ったことないけど）
- なんとなく嫌（理由はないけど，悪い噂を聞くし…）



## オブジェクト指向じゃない？古くて時代遅れ？

- 素の Tk はともかく， Ruby/Tk はオブジェクト指向です！
- Perl/Tk や Python/tkinter よりもオブジェクト指向です！
- Tcl/Tk 自体の開発，進化は全く止まっていません！
- 他の GUI ライブラリが持たない Tcl/Tk 独自の機能により，Tcl/Tk から離れられない人もたくさんいます！

Ruby/Tk を知らない人間による  
的外れな批判では？



## "The 2008 Ruby GUI Survey"

- Ruby プログラマを対象とした GUI ライブラリに関する調査
- Ruby/Tk に関しては…
  - 機能不足で GUI 開発要求に適合しないとする意見が多い
  - 対象中唯一 現在使用中の人数 > 将来の利用希望人数

… ってことになってはいるけれど…



- 回答者の「GUIプログラミング経験」の回答数 395 件中,
    - 経験皆無 141 件 (35.7%)
    - 半年未満 126 件 (31.9%)      合計すると全体の 2/3
- … って,



- 回答者の「GUI プログラミング経験」の回答数 395 件中,
  - 経験皆無 141 件 (35.7%)
  - 半年未満 126 件 (31.9%)      合計すると全体の 2/3

…って,

回答者は本当にきちんと理解した上で  
評価してるの？



- こういう回答者集合だと，  
「現在使用中の人数 > 将来の利用希望人数」  
…って言っても，

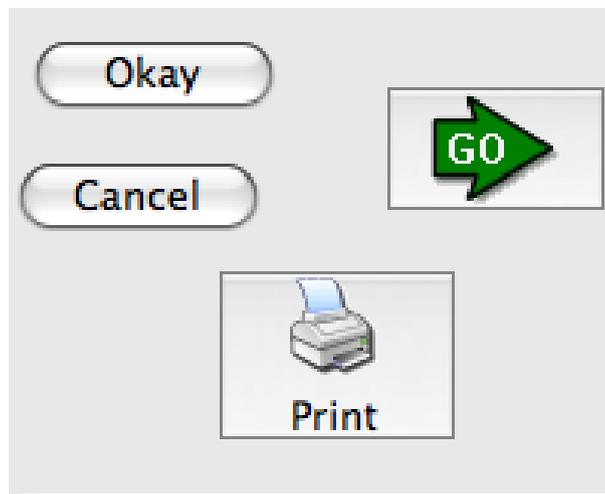
「とりあえず標準添付品を触ってみた」  
という程度の初心者が，  
他に興味を示しているだけなのでは？



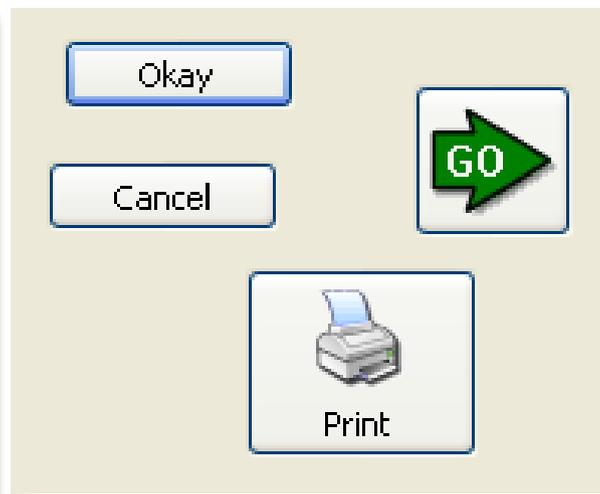
## Ruby/Tk は「見掛けが悪い」？

- 今となっては「昔はそうだったよねえ」って話

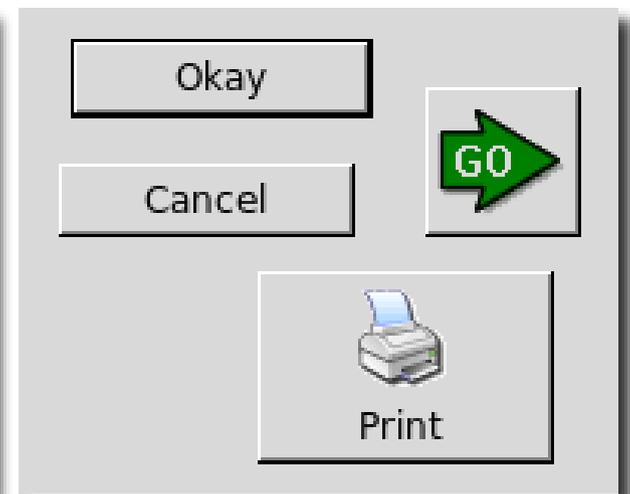
TkDocs (<http://www.tkdocs.com/>) のチュートリアルより最近の Tk で標準化したテーマ付きボタンウィジェット例



Mac OS X



Windows



Linux

他と比べて見掛けで劣るとも思えないけど？



## Survey に関連しての ML 上でのやりとり

- 「テーマ付きウィジェット群が標準化されてるよ」
- 「でも追加インストールが要るんじゃないの？」
- 「いや，以前から標準の Ruby/Tk で使えるよ」
- 「でも存在を知らなきゃ使えないから，  
基本ウィジェットだけで評価しないと…」

…って，**知らなきゃ使えないなんて話は  
他の GUI ライブラリでも一緒だよな？**

Ruby/Tk についてだけ機能を絞って比較されてもねえ…



## 機能が貧弱？ ウィジェットの種類が少ない？

- Tk は多用途に使える少数種類のウィジェットで構成  
→ Ruby の Array クラスが多用途に使えるのと類似した話

単に使いこなせてないだけでは？

- Ruby/Tk はオブジェクト指向なのだから，特定用途のウィジェットやメガウィジェットは**ウィジェットクラスを作っ**てしまえばいい  
→ Ruby のソースアーカイブにはそのような Ruby/Tk のサンプルが含まれているし，Ruby/Tk 自身にもそのような**実例がいくつも**含まれている



## 機能が貧弱？ ウィジェットの種類が少ない？

- 更なる機能強化として，世の中には多数の Tk 拡張が存在  
→ 「基本ウィジェットじゃない」と言っちゃって，  
Tcl/Tk の長い歴史の中で培われた種々の Tk 拡張を  
そのまま使えるのも Ruby/Tk の強みのひとつ

特長に足枷を付けて比べられてもねえ…



## 開発環境， GUI ビルダがない

- GUI ビルダが作られてこなかったのは「**必要ないから**」
- Tk のウィジェット配置は論理的構造や属性での指定が基本
- LaTeX や HTML とワープロとの違いのようなもの
- 「配置」は GUI 作成上の初期の 1 ステップに過ぎない
- 良い GUI に大事なものは，ウィジェット配置完了後の GUI 全体での**細かい動作のコントロール**
  - もちろん Ruby/Tk には十分に可能



ドキュメント不足？

ハイ，不足しています。  
ごめんなさい。

TkDocs(<http://www.tkdocs.com>) が頑張ってくれてるので  
とりあえずはそれに期待ということで(^\_^)；



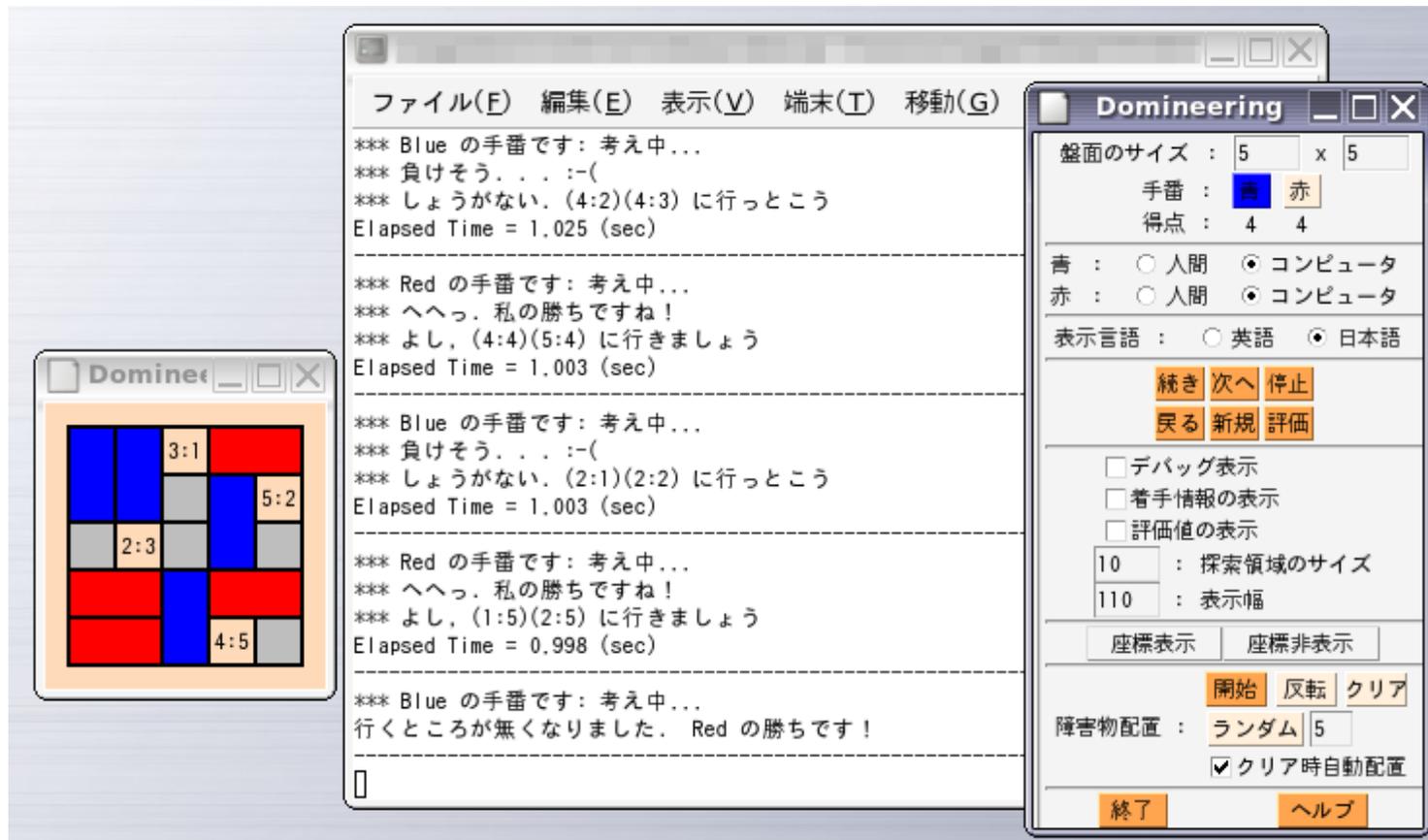
GUI って，外部にサービス提供できないよね？

- ネットワークを介しての不特定多数への GUI サービス
  - 他の GUI ライブラリのことには知らないけれど，  
Ruby/Tk にはそんなフレームワークもなくはないです



# Ruby/TkORCA (Ruby/Tk On RFB Canvas)

例えばこんな GUI アプリを…

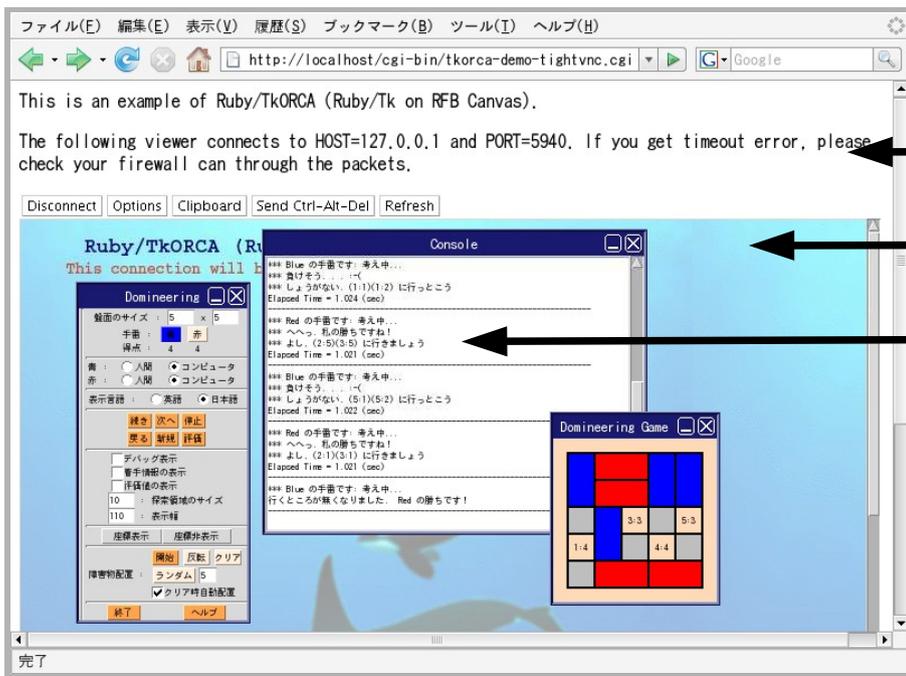


マルチウィンドウ, コンソール利用, ...



# Ruby/TkORCA (Ruby/Tk On RFB Canvas)

こんな風に同時に複数人に独立にサービス提供できます  
(もちろんすべての操作は監視可能で、サーバ側からの割り込みも可能)



普通のブラウザ

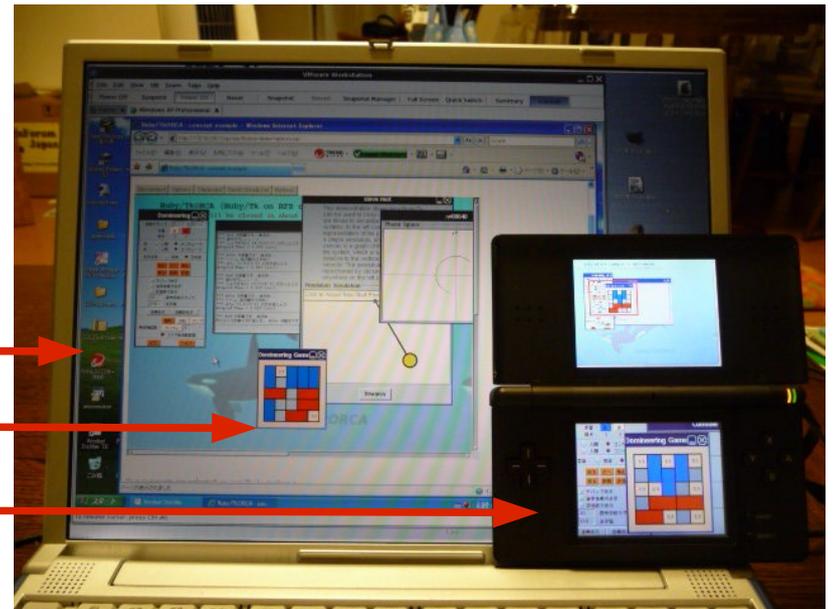
Ruby/Tkによるウィンドウマネージャ

Ruby/Tkによるコンソール

サーバ (Linux)

クライアント (Windows)

クライアント (Nintendo DS)





## まとめ

- Ruby/Tk への批判の中には不当と思えるものも多い.
- Tk ならでは, Ruby/Tk ならではの機能もたくさん.
- 入口が広いだけでなく, **実は非常に奥が深い** Ruby/Tk をもう一度見直してみませんか?