

SlerでもSphinxを使いた  
い！ 後編

---

2014/08/26 kawasaki.rb #15

---

@kk\_Ataka

# 自己紹介

- Twitter: @kk\_Ataka
- GitHub: gosyujin



# 前編のあらすじ

# 前編のあらすじ

- **Sphinx**というドキュメンテーションツールを使いたい
- 導入するための理由づけ(政治)のために競合ツールと比較
  - **Office(Word, Excel)**
  - **Wiki, Markdown**
- 結果、初期導入にかかる手間以外は有用そう！

詳しくは**kwskrb #13**の資料参照

# 今日やらないこと

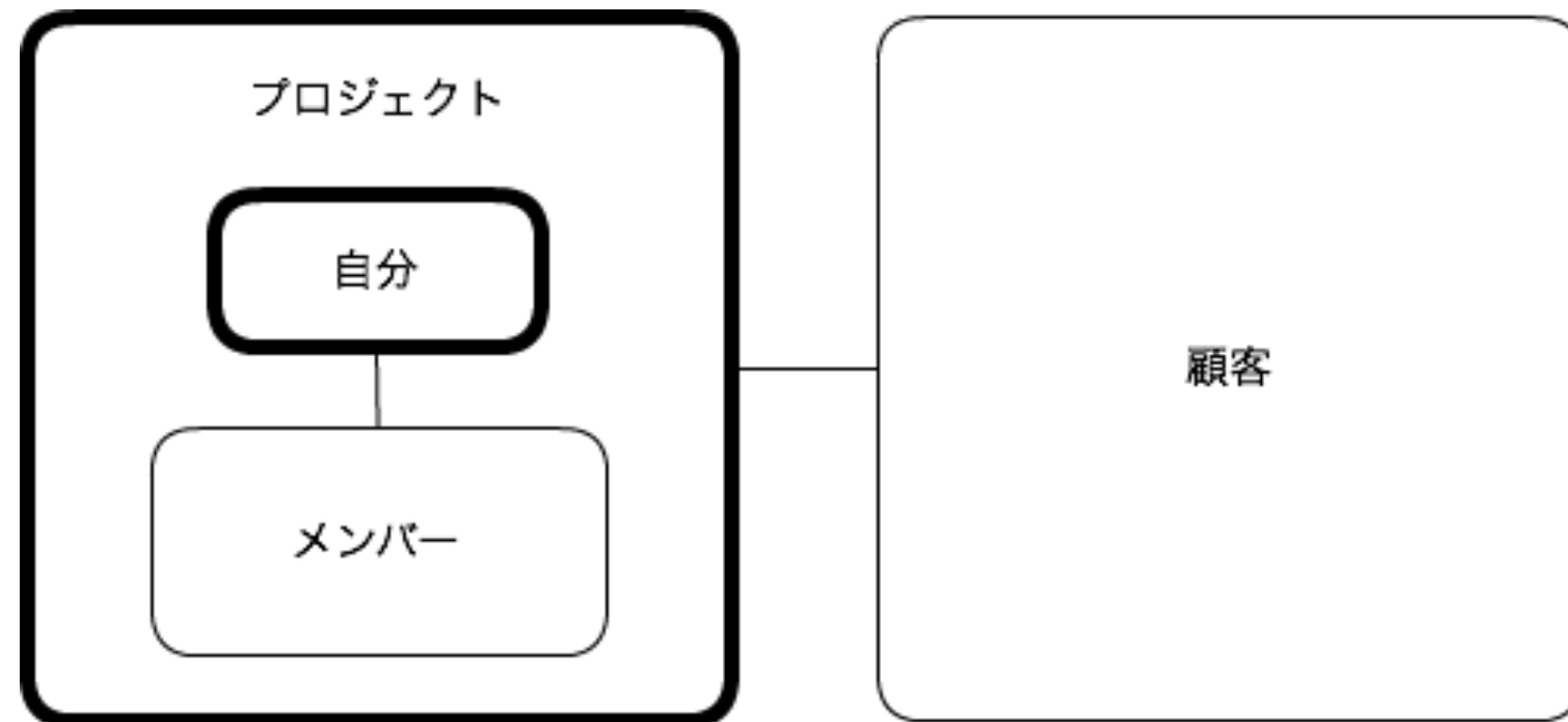
- ➔ Sphinxとはなんぞや
- ➔ 競合ツールとの比較

# アジェンダ

- **Sphinx**とは(前編でやった)
  - さらに
- 競合ツールとの比較(前編でやった)
- 導入のためのあれこれ
- 導入した後どうするか

# 導入するための壁

1. 対プロジェクトメンバー(PM)に対して
2. 対顧客に対して



壁1. 対PM



# 対PM 登場人物

→ 自分

→ **Sphinx**を導入したい人、基本的になんでもやる

→ メンバー

→ 導入した**Sphinx**を使ってほしい人

→ 油断するとすぐ**Office**で日付バージョン管理@共有サーバする

# 対PM 「自分」の仕事

- 今回は**reST**で進めることの明確な宣言
  - 一番大事これができないと負の成果物が生成される...
- メンバーが「特定の**rst**ファイルを開いてドキュメント作成」に注力できる環境を作る
  - **sphinx-quickstart**で下準備
  - ドキュメント自体のアウトライン作成
  - **doctree**の作成

# 対PM 「自分」の仕事

- あわせてビルド環境、デプロイ環境などもお膳立て
  - ビルドは**Jenkins**で拾う
  - デプロイは**Apache**に**html**ファイル配備だろうか

# 対PM 「メンバー」の仕事

- reST記法を覚えてもらう
  - Markdownならなんとか...って場合はMarkdown -> reSTという技もあるにはある
- バージョン管理ツールは使えてください...

# 対PM 課題

- ローカルPCでのプレビュー
  - Python, Sphinxを入れてもらうのは厳しい
  - 確認できるのがサーバにプッシュした時
  - ローカルでreSTプレビューできないだろうか...
  - GitHubだとできるんだけど

# 対PM 課題

- ➡ プロジェクト(会社)の風土にあわせたカスタマイズが必要かもしれない
- ➡ 「変更履歴」出力プラグイン作ってみた

# 壁2. 对顧客

# 対顧客 登場人物

→ プロジェクト

→ 我々側。 **Sphinx**でドキュメント納品します

→ 顧客

→ ドキュメントを納品される側

→ 社内の人 **or** 社外の人

→ 歴史的経緯から **Office**で納品される事が多い

→ 例外は **Javadoc**くらい？



# 対顧客 「プロジェクト」 の仕事

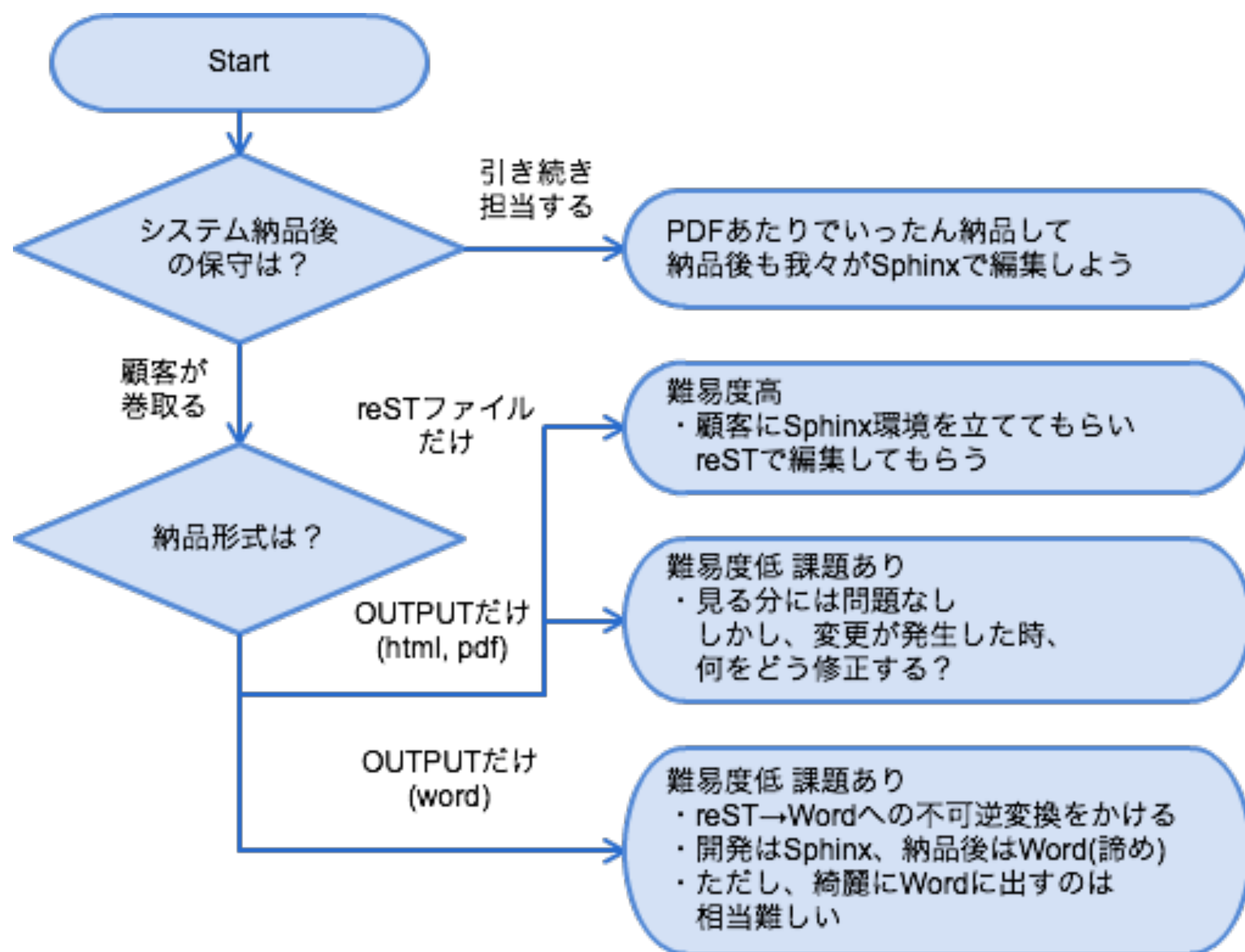
➡ **Sphinx**で作成するドキュメントに関して「今回は**Office**じゃない形式でドキュメント書きますよ」の合意を得とく

# 対顧客 「顧客」 の仕事

➡ 特になし？

# 対顧客 課題

- ➡ **納品形式** 最重要(次ページ参照)
  - ➡ 納品後誰が保守するのが大事
  - ➡ 顧客が巻取ってしまう場合、どうしたらよいか
    - ➡ 今のところ、納品対象外となっているドキュメントだけ**Sphinx**を使うという方針しかない？
  - ➡ 要解決事項、誰かどうやってるか教えてください...



色々考えた上で臨んだPJ

# 現状

- ➡ デモシシステムのためドキュメントから何からすべて一任
- ➡ 紆余曲折があり開発担当が自分ほぼ一人になってしまった  
(リーダー的な人とかは別にいる)
- ➡ やりたい放題ランド
- ➡ 社内テンプレート(謎)仕込まれる前に前倒しでドキュメント作りまくり

# 現状

- バージョン管理、差分管理もできて快適
- 軽い(普通のエディタで編集してるから)
- 意外と営業の人に受けが良かった
  - 「え？なにこれ？なんてツール？あとで教えて」

# 現状

- **Sphinx**導入はできたが、思い描いていた事は実践できず...
- ほぼ一人プロジェクトなので
- しかし、ドキュメント開発スピードなどは加速できている！(まったく定量的でない評価)
- **Office**特有のもっさり感なし！
- 履歴管理など気にする必要なし！



# 結論

- プロジェクト全体でドキュメントを**Sphinx**で書いていく！  
空気を作るのが難しい
  - 一人くらい好意的な「仲間」がいるとやりやすい
- メンバーに**reST**を強いるのが難しい
  - せめて**markdown**を書いたことがあれば敷居は低くなるのに...
- ドキュメントを書くスピードは爆速(個人的に！)