

プログラミング未経験から AIでソフトを作るまで

Solar Thermal Simulator (STS)
Ver.1.4 開発事例

池田 正規



最初にお伝えしたい結論

コードを書けなくても、
設計の判断はできる。

人が担当

目的・仕様・妥当性判断

AIが担当

説明・実装・修正・検証

完成したのは“AI任せ”ではなく、設計者とAIの共同作業です。

出発点は、専門業務の素朴な困りごと

やりたかったこと

- METPV20気象データを使う
- これまでの設置地点・設置方位・傾斜角度だけ
シミュレーションではなく仰角追尾・方位追
尾・仰角方位2軸追尾方式を加える
- 集熱器の種類や設置方式ごとの集熱量を比較
- 結果をExcelやグラフで確認

分からなかったこと

- Pythonの書き方
- ColabやPowerShellの操作
- Codexへの頼み方
- GitHubのbranch・merge・release

“作りたい計算”は分かる。でも“作り方”は分からない。

最初の相談は、完成した仕様書ではなかった

私の相談（要約）

「PythonやCodexを使って、気象データのMETPV20を元とした角度追尾や方位追尾の太陽熱シミュレーションを計算するプログラムを作れますか？」



AIの返答（要約）

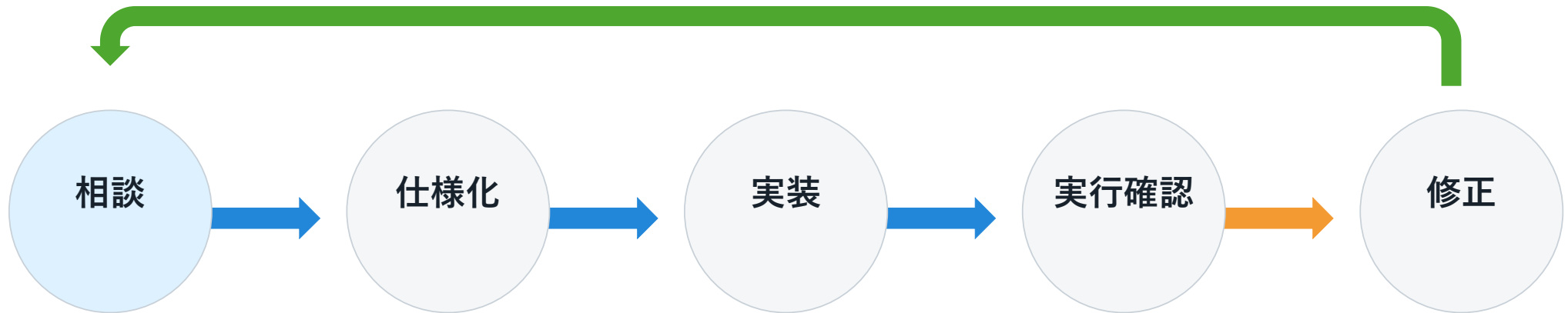
「可能です。まず入力データと計算条件を整理し、段階的にPythonで実装しましょう。」

大切だったのは、最初から完璧なプロンプトを書くことではありません。

5つの役割が、1つの開発チームになった



開発は、この小さな循環の繰り返し



うまくいかない画面をスクリーンショットで送り、AIと原因を切り分けました。

例：古いNotebook／空CSV／別PCの仮想環境／グラフ表示

設計者にしかできない仕事が、品質を決めた

AIはコードを書けても、
“正しい結果”までは決められない。

✓ 福山と宗谷岬の差は自然か

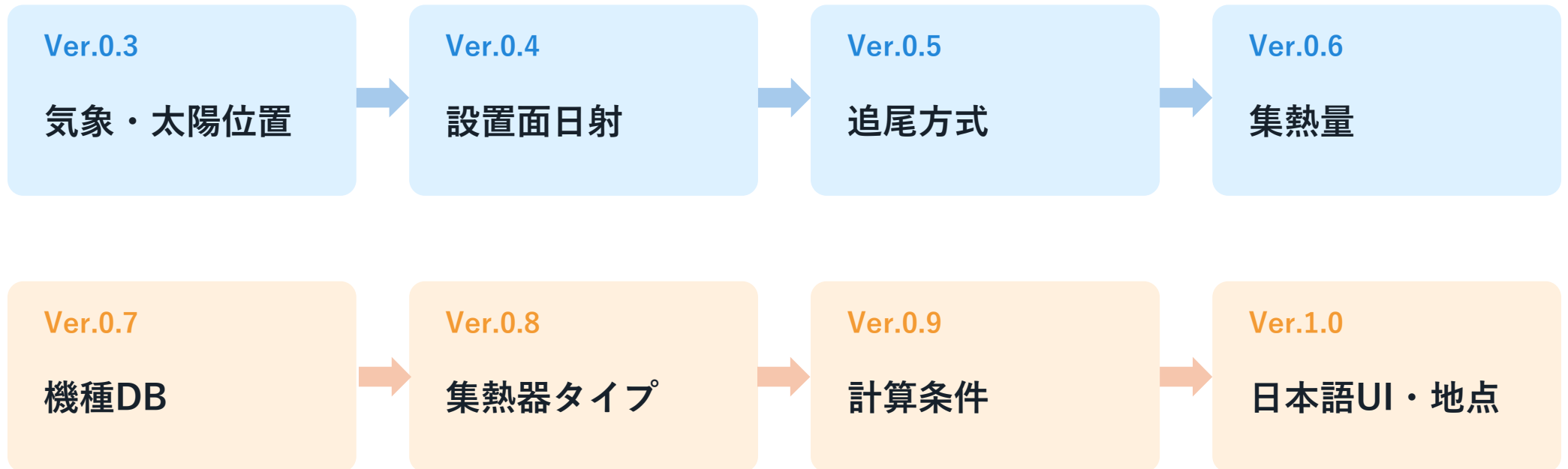
✓ 50°Cと85°Cで集熱量が下がるか

✓ 東90° ・ 南180° ・ 西270° か

✓ 既存燃料量と横並び比較か

AIの出力を“専門家の常識”で検算する

まず、計算の土台を段階的に積み上げた



大きな完成品を一度に頼まず、1バージョン=1つの目的にしました。

Ver.1.1以降は“使えるソフト”へ進化



※ Ver.1.4の詳細機能は、当日の実画面に合わせて差し替え可能です。

GitHubは“公開場所”より先に、安全装置だった

Branch (ブランチ)

新機能を別の枝で試す

Commit (コミット)

変更の区切りを記録

Merge (マージ)

確認後に正式版へ統合

main  develop-ver1.x

失敗しても戻せる。
別PCでも続きを始められる。

分からない言葉も、その場でAIに聞いた

私

「PowerShellとは
何でしょうか？
使ったことはありません。」

AI

「Windowsを文字で操作する
標準アプリです。今回は
あなたが操作しなくても進めら
れます。」

知らない用語を隠さないことが、むしろ安全につながりました。

Chat／Work／Codexは、同じ仕事場に近づいた

以前の体感

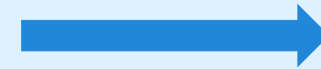
ChatGPTで相談



Codexへ切替・貼付

現在の体感

Workの中で相談



必要に応じて実作業まで連携

結論：設定を変えた可能性より、OpenAI側のUI・機能更新による変化と見るのが自然。

ただし「Chat／Work」表示名や配置は段階的に変わるため、当日の画面を優先。

AIに任せたこと、任せなかったこと

AIに任せた

- コードのたたき台
- エラー原因の説明
- ファイル修正とテスト
- README・指示文の整形
- インストレーター作成

設計者が決めた

- 何を計算するか
- 計算式・係数の妥当性
- 使いやすい入力方法
- 結果が自然かどうか
- いつ正式版にするか

役割分担を曖昧にしないことが、AI活用のコツです。

初心者向け：AIへの頼み方は4項目で十分

1 目的

何を作りたいか

2 入力

どんなデータを使うか

3 出力

何を表示・保存したいか

4 制約

変えてはいけないもの

例：「METPV-20を読み、年間集熱量を計算し、Excelに出してください。
既存の計算式は変更せず、テスト結果も報告してください。」

皆さんが明日から試すなら、小さな業務から

Excel集計

毎月の転記を自動化

計算ツール

社内式を画面化

図面チェック

確認項目をリスト化

報告書

データから下書き

“失敗しても困らない”“正解を自分で確認できる”仕事が最適です。

AIは、専門家を置き換えるのではなく 専門知識を“動く形”に変える相棒

設計者の判断 × AIの実行力
= プログラミング未経験でもソフトを作れる

まずは1つの小さな困りごとを、AIに相談してみてください。