

SAS Macro Libraryの開発/管理/運用

田村 洋介
臨床統計・プログラミング部
アストラゼネカ株式会社



はじめに

SAS Macro好き

SAS言語以外の経験

本日の目的

事例紹介

.....懇親会で問題点の解決策を覚えてもらう.....

新機能の提案

.....一人だと心細いので皆さんの反応をうかがう.....

本日の内容

- 報告 / Report
 - 目標 / Goal
 - したこと / Done
 - できなかったこと / Homework
- 言い訳、泣き言、責任転嫁 / Garbage
- 提案、お願い、まる投げ / Request
- まとめ / Conclusion

環境/前提条件

■ 環境

- Windows 2000
- SAS 8.2
- Office 2000

■ 前提条件

- SASでのMacro作成経験
- SASで他人が作ったMacroの使用経験

汎用SAS Macro集 =SAS Macro Library

報告 / Report 5W2H

- Who(誰が) 私が管理者として
- What(何を) SAS Macro Libraryの開発/管理/運用を
- When(いつ) 2004/08頃から
- Where(どこで) 部門ファイルサーバで
- Why(なぜ)
 - Word出力用のMacroの取り扱いを、部門内で明確にするため。
 - SAS Programmerとして、仕事ができないので、合法的に人のProgram Codeを頂こうと思ひ立ち
- How(どのように)
 - SASの上級者/部門ファイルサーバの管理者に相談しながら、
- How Much(いくらで)
 - 特に、新規のインフラ等はなく、その時あるもので
 - 掛かった時間は...プライスレス？

目標 / Goal どんなSAS Macro Library?

ISO9126

ソフトウェアの品質



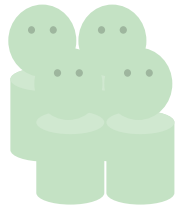
http://www.ogis-ri.co.jp/otc/hiroba/technical/JavaPress_ISO9126/index.html

簡単に！

安心して！

楽しく！

目標 / Goal 簡単に!



Macroの利用者

- 事前の準備/作業なしで
- 使用方法/使用するべき個所/範囲がすぐ判る



Macroの開発者

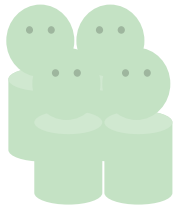
- 明確で有用な開発/標準化手法がある
- 面倒な規則や冗長な手続きがない



Libraryの管理者

- 管理作業が簡単にできる
- 利用者/作成者が規則にしたがっているか、簡単に確認できる

目標 / Goal 安心して!



Macroの利用者

- Fool-Proofにできている
- 副作用がすぐわかる
- サポート



Macroの開発者

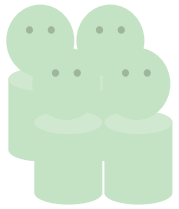
- 追跡性の確保
- 有限責任
- 他のMacroを使用した場合の正確性



Libraryの管理者

- 追跡性の確保
- Fool-Proofにできている
- サポート
- 有限責任

目標 / Goal 楽しく!



Macroの利用者

- 煩雑な作業/嫌な作業を軽減してくれる
- おもしろい機能/動作が実装されている



Macroの開発者

- 使われているかが判る
- Feedbackが受けやすい
- インセンティブが明示的に存在する



Libraryの管理者

- 使われているかが判る
- 利用者/開発者からFeedbackが受けやすい
- インセンティブが明示的に存在する

したこと / Done 方針

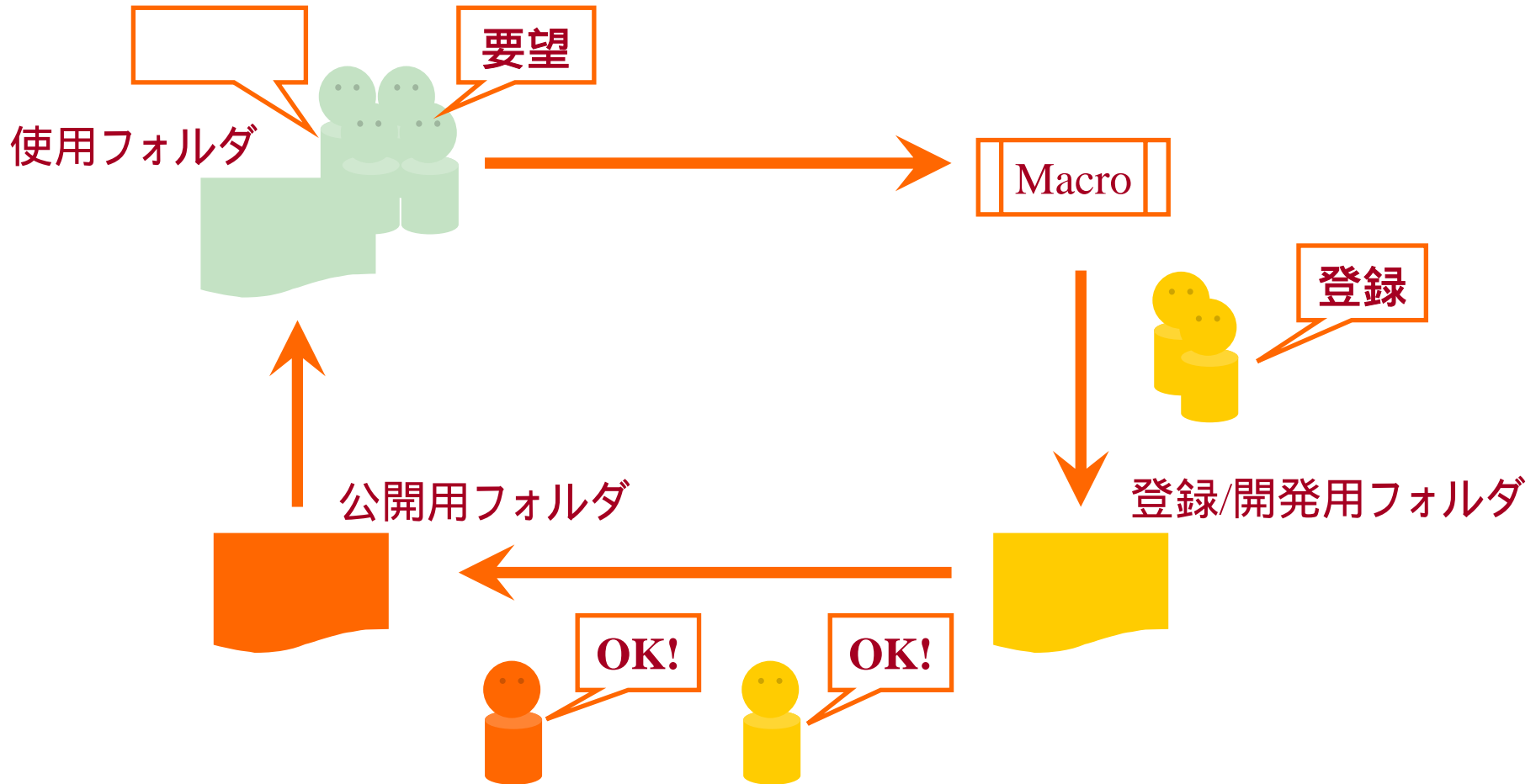
- 規則少なめで!
 - 覚えるのが大変
 - チェックするのが大変
- スコープは小さく!
 - 問題を発生させない<<<発生したら影響する範囲を特定できる

したこと / Done 生成物

- Macroの利用/作成/Validation手法に関する規則
 - [readme.doc](#)
- MacroのValidation用テンプレート
 - [ValidationTemplateforSASMacro CIMNo6](#)
- 保管場所/Access権の設定
- Macroの命名に関するガイドライン
 - [NamingRule.txt](#)

- 副産物は後ほど

Macroの登録/公開方法



Macroの作成方法

- マクロ名:C_***** (任意)01 (Version番号)
- 1Macro/1ファイル
- ヘッダーに、Macroの仕様をすべて記述する[MacroHeader.sas](#)
 - 管理情報
 - 作成日付、作成者、更新日、更新者、SASVersion
 - 仕様
 - 機能、目的、パラメタの説明、Macroの履歴、参照先
 - 制限事項、必要事項、事前準備、禁則事項
 - パラメタ以外で、そのMacroによって設定/作成される以下を記述
 - Macro中で使用しているMacro
 - Read/Write/UpdateされるLocal/Global macro変数
 - Global option/Goption
 - 一時的なデータセット/変数
 - Libname指定、Filename指定
 - 外部ファイル
 - 実行例

MacroのValidation方法/テンプレート

- Validation Plan
 - Macroの想定される使用状況、重要性に基づき、Validation方法を選択
 - レビュー/ロジカルチェック/出力のチェック/ダブルプログラミング
- Test Script Document
 - 具体的なチェック方法と期待される結果をプランとして記述
 - Macroを実行して、結果を記録し、期待される結果と比較して、問題ないか検討する
- User Report
 - Macroを実行して、結果を記録する
 - 問題/要望があれば、記録しておく
- 関連文書や、実行結果も保管
- Validation Report
 - 上記のファイルを元に、最終的なValidation結果を記録する
 - 今後、予想される機能拡張や問題点も記録しておく

保管場所 / Access権の設定 / その他

- 保管場所
 - 部門ファイルサーバにSAS Macro Libraryを作成
- Access権の設定
 - Library 管理者のみ、すべてにRead/Write
 - Library 管理者以外は、登録/開発用フォルダのみRead/Write、それ以外はRead
- その他
 - SAS Macro Library中のMacroをすべて一括で指定したフォルダにCopyするScript
 - MacroがVersionUpした場合、古いMacroを、Copyしているか、管理者が一括でチェックするためのScript
 - SAS Macro Libraryへの要望を出すための
 - MailForm
 - Excelファイル

Macroの命名に関するガイドライン

- あくまで提案、強制でなく
- 現在もメンテナンス中...(一番難しい)
- 以下の為の命名規則
 - Macro名/ Macro パラメタ名
 - Write/UpdateされるLocal/Global macro変数
 - 一時的なデータセット/変数
 - Libname指定、Filename指定
 - 外部ファイル
- 中間生成物か最終成果物か
 - 使用しても良い省略形
 - 名前の統一
 - Prefix/Suffix
 - 処理の範囲を明示

現在

- Macro数
 - 使用中:34個 in 25機能
 - 開発中:8機能
- Programmer:9人
 - 作成者 3人/9人
 - Validation経験者 5人/9人
 - 利用者 11人?/9人
(解析担当者も2人使用しているため)
- 副産物
 - SAS Macro Library以外の部門ファイルサーバのフォルダ構造の標準化
 - コーディングガイドライン
 - SAS Macro 以外の為のValidation用
 - プロセス定義
 - テンプレート
 - 個人的には...

できなかつたこと / Homework

			×
Macroの 利用者	事前の準備/作業なしで	副作用がすぐわかる	使用方法/使用するべき 個所/範囲がすぐ判る
	サポート	おもしろい機能/動作が 実装されている	Fool-Proofにできてい る
	煩雑な作業/嫌な作業を 軽減してくれる		
Macroの 開発者	面倒な規則や冗長な手続 きがない	明確で有用な開発/標準 化手法がある	他のMacroを使用した場 合の正確性
	追跡性の確保	有限責任	インセンティブが明示的 に存在する
	利用者/開発者から Feedbackが受けやすい		使われているかが判る
Library の管理者	管理作業が簡単にできる	利用者/作成者が規則に したがっているか、簡単 に確認できる	インセンティブが明示的 に存在する
	追跡性の確保	有限責任	使われているかが判る
	利用者/開発者から Feedbackが受けやすい		

今後の対策

- 副作用がすぐわかる
 - 仕様書の書き方に追加する/危険度を評価する
- 使用方法/使用するべき個所/範囲がすぐ判る
 - 命名ガイドラインに取り込む
- おもしろい機能/動作が実装されている
 - Newsletter等を発行する?
- 有限責任
 - サポート担当者にValidation担当者も加える
- 使われているかが判る
- インセンティブが明示的に存在する
 - Macro呼び出しが記述されているProgramを検索して、ランキングする
 - ランキングに基づき、上司に報告する/お菓子をやり取りする
- 明確で有用なMacro開発/標準化手法
 - 定期的に見直すことをプロセスに組み込む?

言い訳、泣き言、責任転嫁 / Garbage

- 明確で有用なMacroの開発/標準化手法
 - そもそもSAS Program用の開発/標準化手法がない
 - 全員に対して“明確で有用な開発/標準化手法”は難しい
 - SAS言語の教育から始めるべき?
- 利用者/作成者が規則にしたがっているか
- 他のMacroを使用した場合の正確性
- Fool-Proofにできている
 - 問題があったら実行を止める/元の状態に戻せる
- Programを読んで、実行して、Logと結果を詳細に見ないと判らない
 - MacroのパラメタやDataに基づいて作成/指定される値のチェック
 - 特定の条件下で問題があるのに、エラーがなく実行できる
- Macroの機構上
 - パラメタの型や、値のチェックが煩雑/難しい
 - Macro間での、Macro変数/データセット/変数の安全な受け渡しが難しい
- 煩雑で、且つ効果的でない=開発効率が下がる

規則 / Fool-Proof / 他のMacro

- 通常のSAS開発環境単独で対応するのは難しい
 - パラメタの制限不能 [Sample1.sas](#)
 - Macro変数のスコープの制御が難しい [Sample2.SAS](#)
 - %let / Call symputかでも動作が変わる
 - Macro展開時 / 実行時Errorのハンドリング
- SAS Macroって自由すぎ!
 - 制限すれば?
 - AF等を使用して、Macroを利用するシステム
 - Ex) Ph.Clinical / ETL Server / SAS® Drug Development
 - MacroにParameterを与えるInterfaceを提供する
 - ヒューマンエラーは減らせる
 - そもそもMacro化する必要がない?

提案、お願い、まる投げ / Request

- SASへのお願い
 - Global/Local Macro変数の制御の厳密化
 - 再帰処理のサポート
 - パラメタの制限用の型導入
 - Macro展開時/実行時Error発生をトリガとする、Error制御用ステートメントの追加

...ではなく

- SASの得意なところを伸ばして欲しい

もっと自由を!

SASの得意なところ

■ SASの得意なところ

- Macro変数、オプション等の環境変数の公開
 - 他言語では、メモリ上にのみ存在
- 理解しやすいMacro機構
 - 他言語では、効率化の為の機能は、精通しないと理解が難しい
- 互換性
 - 他言語より互換性をかなり重視してくれている
 - 追加される機能は、新しいステートメント、新しい予約語、新関数、新オブジェクトで実行される
 - V9では
 - *****Z関数、*****X関数(ROUNDZ FLOORZ VVALUEX)
 - %Copy
 - Call Symputx / SYMEXIST / SYMGLOBL / SYMLOCAL
 - Macro変数のスコープの制御に使える!

提案、お願い、まる投げ / Request

- Macro変数に関して
 - Macro変数をGlobal/Local等の中のどのスコープに作成するか指定できる
 - Macro変数をGlobal/Local等の中のどのスコープから読むか指定できる
 - 上記を禁止できる
- Macroに関して
 - 新DictionaryTable
 - Macroのパラメタ情報が格納されたDictionaryTable
 - MacroのProgramが格納されたDictionaryTable
- Macroのデバッグに関して
 - Mprint / Mlogic / Symbolgen等によるLogでの確認だけでなく、Macroの生成した最終的なProgramが、データセット/カタログに出力される

Macro変数に関して

■ Sample3.sas

- Macro変数をGlobal/Local等のどのスコープに作成するか指定できる
 - %let ステートメントを拡張する
 - あわせてCall Symputxも拡張する
- Macro変数をGlobal/Local等のどのスコープから読むか指定できる
 - Macro中での読み取りのスコープを指定できる
 - %Symbol ****;デフォルトはGlobal
- 上記を禁止できる
 - 前Versionとの互換性を考慮して
 - Macro中で禁止できるステートメント
 - 禁止するオプション指定

■ Macro変数のスコープの問題一挙解決!!!

Macroに関して

- [Plan.xls](#) [Sample4.sas](#)
- 新DictionaryTable
 - VMPARAM Macroのパラメタ情報が格納されている
 - VMPROG MacroのProgramが格納されている
 - V9で%Copy機能を使えば、MacroのProgramを呼び出せるので、ありえそう
- Macroのパラメタのチェック、Validationに使える
- Macroを動的に生成!
 - 一部のパラメタを固定して
 - Macro中の特定の部分を置換して

Macroのデバッグに関して

- [Sample5.sas](#)
- 新オプション/ODSの新機能
 - Mprint / Mlogic / Symbolgen等によるLogでの確認だけでなく、Macroの生成した最終的なProgramが、データセット/カタログに出力される
- Macro関連の新機能を追加して、ますますMacroのデバッグが難しくなるであろうことから
- Macroをデバッグが効率的に!
- Macroの学習も効率的に!

皆さんへの要望

ご意見あれば是非!

- SAS Programの開発/標準化手法
- Macroの開発/標準化手法
- 本日の提案に

教えて下さい

- SASへの要望の挙げ方

Yousuke.Tamura@astrazeneca.com

まとめ / Conclusion

- 人の作ったMacroを見ると、勉強になる / 各人の違いに気がつく
- 問題点も、協力して解決 / 解決できなくても共有
- SAS Macro Libraryの開発 / 管理 / 運用は

たのしい、