Standardised Output (SO): flexible and tool-independent storage format of typical M&S results

Nadia Terranova (1), Marc Lavielle (2), Mike K. Smith (3), Emmanuelle Comets (4,5), Kajsa Harling (6), Rikard Nordgren (6), Duncan Edwards (7), Andrew C. Hooker (6), France Mentré (4,5), Maciej J. Swat (8)

(1) Merck Institute for Pharmacometrics, Merck Serono S.A., Switzerland, (2) Inria Saclay, France, (3) Global Clinical Pharmacology, Pfizer, UK, (4) INSERM, IAME, UMR 1137, France, (5) Univ Paris Diderot, IAME, UMR 1137, Sorbonne Paris Cité, France, (6) Department of Pharmaceutical Biosciences, Uppsala University, Sweden, (7) Simcyp (a Certara company), Sheffield, UK, (8) EMBL-European Bioinformatics Institute, UK

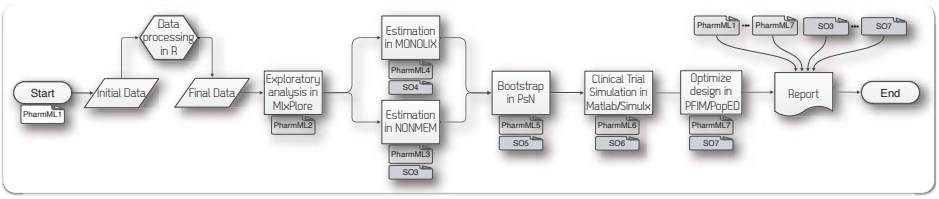
Introduction

Objectives

The definition and implementation of formats enabling a reliable exchange of pharmacometric models across software tools is one of the key goals for efficiently promoting collaborative drug and disease modelling and simulation (M&S) research. **PharmML**, one of the key DDMoRe interoperability platform elements, has been designed to play the role of the exchange medium for mathematical and statistical models [1]. Similarly, the Standardised Output (**SO**) has been developed as a complementary element, tool-independent format, for storing typical output produced in a pharmacometric workflow.

As a generic output model, SO aims at:

- providing a **flexible storage structure for typical results** of M&S analyses performed in any DDMoRe target tool;
- enabling **effective data flow across tasks** to ensure optimal interactions among software tools and, then, extend the modelling capabilities of the workflow;
- facilitating **information retrieval for post-processing and reporting**, by allowing immediate access to M&S results.



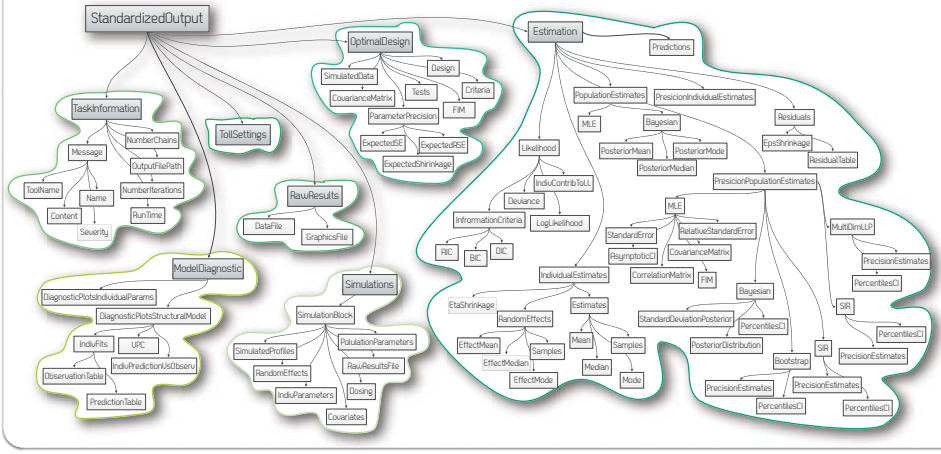
Effective workflow support – the key benefit coming from SO usage

SO

SO Structure

SO consists of the following seven main sections:

- **Tool Settings** – storing the reference to any file containing the tool settings of a performed task.
- **Raw results** – placing references to original output files, both data and graphics, produced by any target tool.
- **Task Information** – holding the information about the modelling step execution.
- **Estimation** – storing typical output of interest resulting from an estimation task.
- **Model Diagnostic** – designed for storing information resulting from typical model diagnostic plots.
- **Simulation** – storing typical results produced in a simulation task.
- **Optimal Design** – storing results coming from a evaluation or optimization step.



Example 1: Population estimates

Typical output for parameter estimation. First the columns – here the names of population parameters – are specified. Then two options exist: either the results are stored inline or in an external data file.

Example 2: Optimal design

In an optimal design task, e.g. evaluation, relevant results such as the Fisher Information Matrix (FIM) and the covariance matrix are captured.

Target tool – SO connection: libSO & connectors

DDMoRe Target Tools

Simulx | PFIM | PopED | Monolix | winBUGS | NONMEM | Simcyp Simulator | PsN | R | Matlab

Target tool – PharmML connection: libPharmML & connectors

PharmML

Model Definition

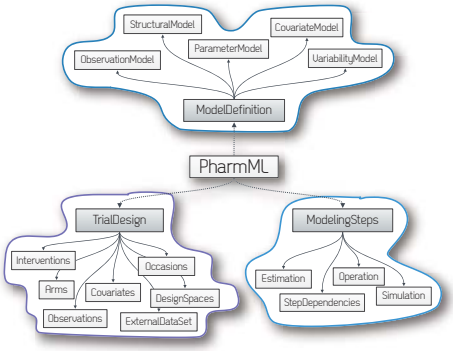
- **Variability Model** – allows to define any number of variability levels as a nested hierarchy.
- **Covariate Model** – describes the covariates, both continuous and discrete, their distribution, transformation and interpolation.
- **Parameter Model** – offers flexible structure to encode structured and equation type parameter models with any number of variability levels and covariates.
- **Structural Model** – prediction PK, PD or disease models can be formulated here as ODE, DDE or algebraic equations.
- **Observation Model** – model for continuous or discrete data models.

Trial Design

Specifies explicitly the structure of a clinical study, used for optimal design and simulations, as an alternative to a design sourced from a dataset.

Modelling Steps

The specification of how a mathematical model and the associated trial design can be used, e.g. for simulation, estimation or optimal design tasks.



Non-linear mixed effects model

The general NLME model for N subjects and n_i measurements per subject i reads as follows [2]

$$\underbrace{y_{ij}}_{\text{Experimental data}} = \underbrace{f(x_{ij},\psi_i)}_{\text{Model prediction}} + \underbrace{g(x_{ij},\psi_i,\xi)\,\epsilon_{ij}}_{\text{Residual error}}, \quad 1 \le i \le N,\ 1 \le j \le n_i$$

y_{ij} - j^{th} observation for subject i
f - structural model prediction
x_{ij} - regression variables
ψ_i - individual parameters
g - standard deviation of the residual error
ξ - parameters of the residual error
ϵ_{ij} - residual error

References

[1] Swat, MJ., Moodie S., Wimalaratne S., et al. (2015). Pharmacometrics Markup Language (PharmML) – Opening New Perspectives for Model Exchange in Drug Development, Accepted for publication in CPT:PSP.

[2] Lavielle, M. Mixed Effects Models for the Population Approach: Models, Tasks, Methods and Tools. Chapman & Hall/CRC Biostatistics Series. 2014.

Acknowledgement: The research leading to these results has received support from the Innovative Medicines Initiative Joint Undertaking under grant agreement n° 115156, resources of which are composed of financial contribution from the European Union's Seventh Framework Programme (FP7/2007-2013) and EFPIA companies' in kind contribution. The DDMoRe project is also financially supported by contributions from Academic and SME partners.