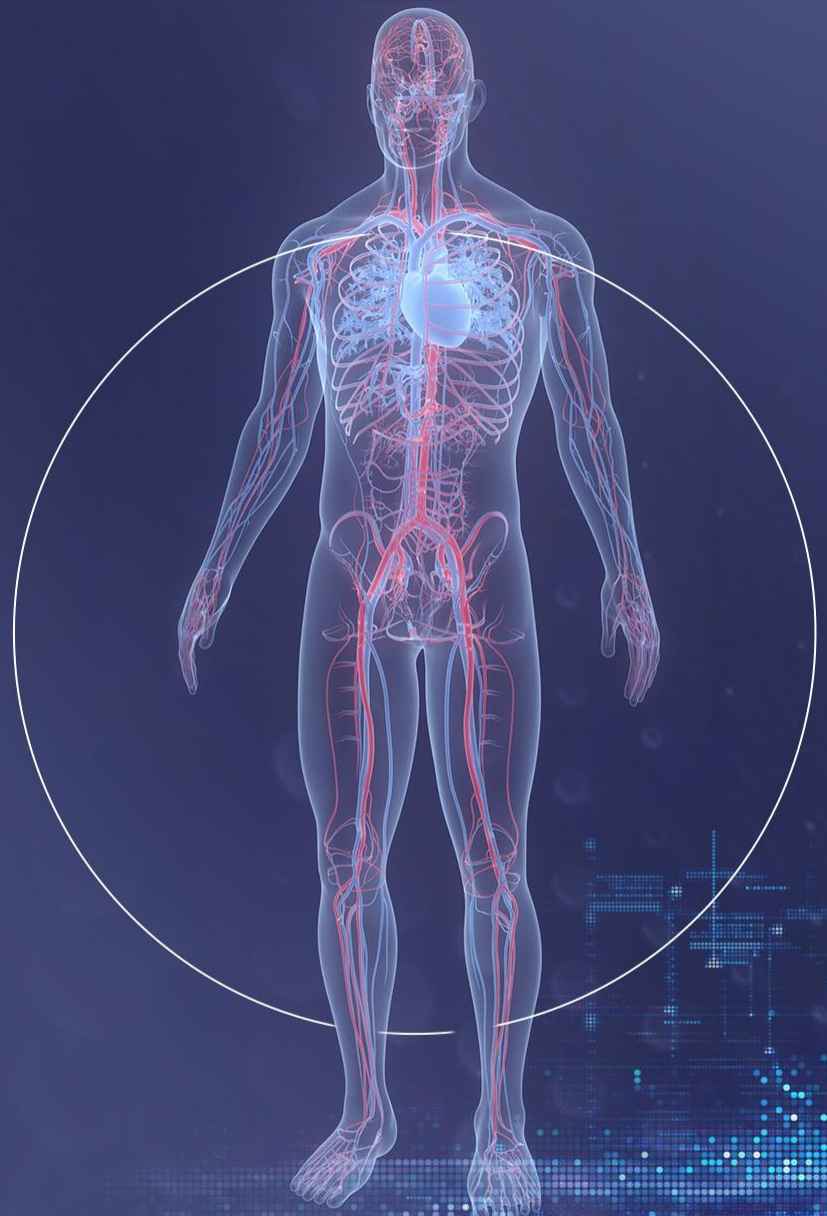
esqlabs
we empower life sciences

OSMOSES

Modularization and Qualification of complex PB-QSP models with OSP version 12



Pavel Balazki @OSP Community Conference 2024

OSMOSES

Modularization of PB-QSP projects as a solution for

1) Management of ever-growing model complexity
2) Model re-usability
3) Automated (re-)qualification

OSMOSES

OSPS Version 12

Modularization of PB-QSP projects as a solution for

1) Management of ever-growing model complexity
2) Model re-usability
3) (Automated (re-)qualification)

Acknowledgements

Collaborators & Sponsors

Sponsors

Acknowledgements

Collaborators & Sponsors

Sponsors

Federal Ministry of Education and Research

Boehringer Ingelheim

sanofi

esQlabs
we empower life sciences

Developers

- Juri Solodenko
- Michael Sevestre
- Robert McIntosh
- Benjamin Mariano Perez
- Abdel Rodriguez (former)
- Georgios Daskalakis (former)

DESIGN{2}CODE

esQlabs
we empower life sciences

Acknowledgements

Collaborators & Sponsors

Sponsors

Federal Ministry of Education and Research

BAYER

Boehringer Ingelheim

sanofi

esQlabs
we empower life sciences

Model library contributors

University of Saarland,
Clinical Pharmacy (Prof. Dr. Thorsten Lehr)

UNIVERSITÄT DES SAARLANDES

01

QSP model lifecycle

Need for new approaches

Model management through the entire lifecycle

Development, Qualification, and Application & Delivery

Model Development → Model Qualification → Model Application

- **Model Development**
  - Biology and literature deep dive, model architecture and design, maths & stats formalization, implementation, parameter estimation
- **Parameter Database**
  - Same structure, different parameter sets for different use cases
- **Module library**
  - Combination of available and new model structures

- **Evaluation of model performance**
- Requirement for **automated** Model **Evaluation and Reporting** Framework

- **Distribution and Deployment**
- **Re-usability**
- **Maintenance**

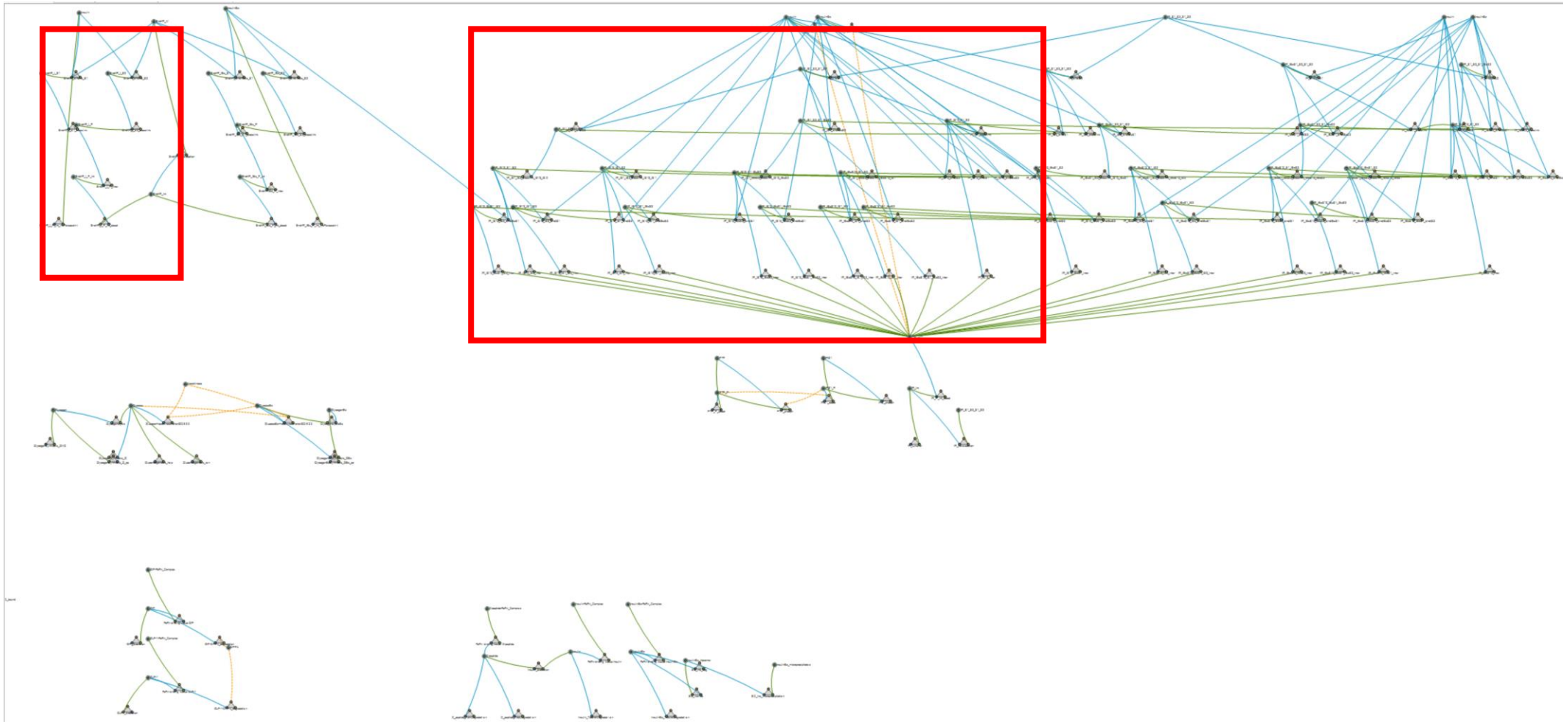
Problem: high model complexity

- Growing model complexity makes maintenance and debugging challenging
  - Both during model development and application

Problem: high model complexity

Example – Diabetes Platform

- G



Problem: re-usability

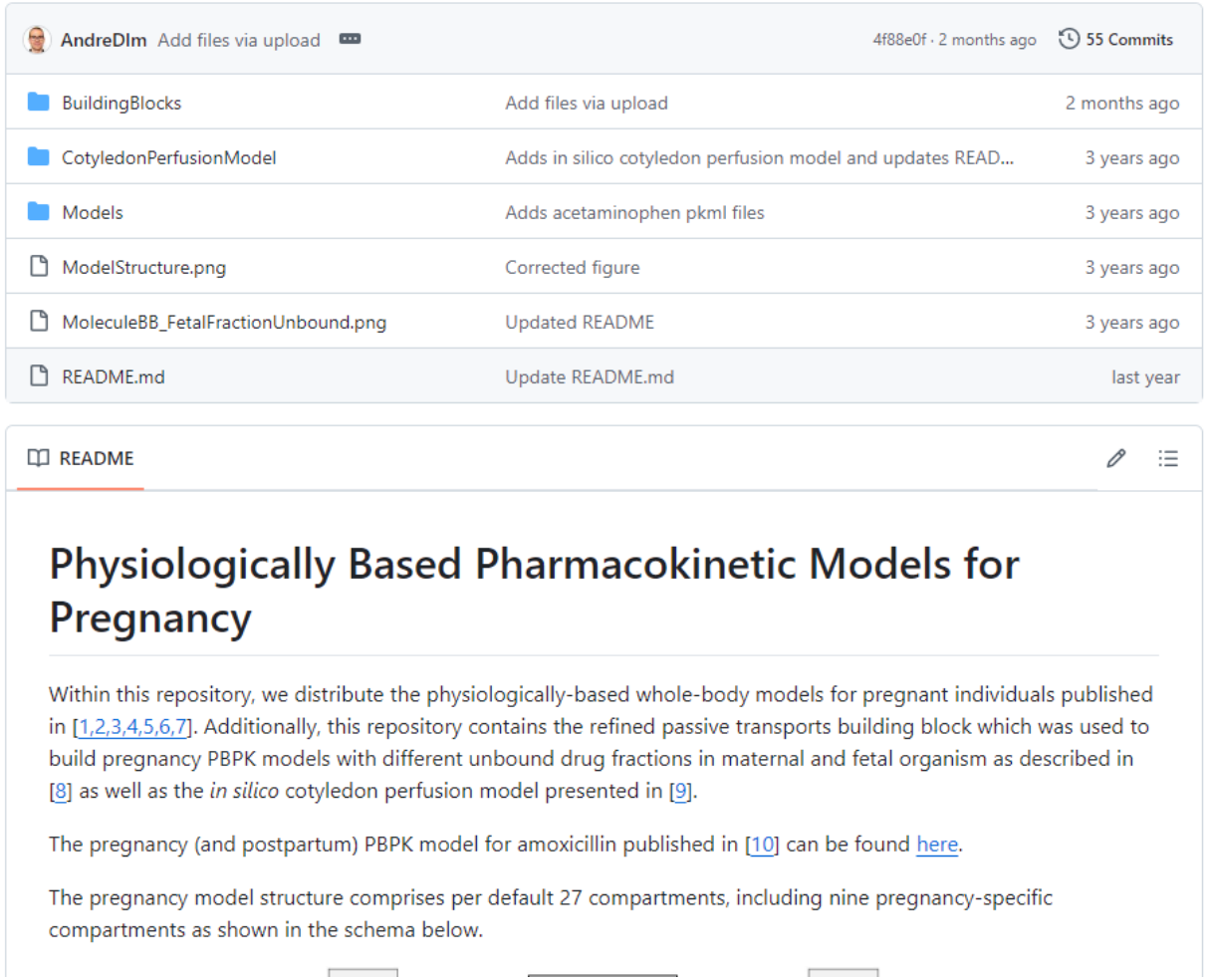
- Growing model complexity makes maintenance and debugging challenging
- Models are developed for a specific use case with limited re-usability, or…

Problem: re-usability

- Growing model complexity makes maintenance and debugging challenging
- Models are developed for a specific use case with limited re-usability, or…
- Integration into new models is tedious and requires multiple manual steps

Problem: re-usability

- Growing model comple
- Models are developed
- Integration into new mo



AndreDlm Add files via upload — 4f88e0f · 2 months ago — 55 Commits

BuildingBlocks	Add files via upload	2 months ago
CotyledonPerfusionModel	Adds in silico cotyledon perfusion model and updates READ...	3 years ago
Models	Adds acetaminophen pkml files	3 years ago
ModelStructure.png	Corrected figure	3 years ago
MoleculeBB_FetalFractionUnbound.png	Updated README	3 years ago
README.md	Update README.md	last year

README

Physiologically Based Pharmacokinetic Models for Pregnancy

Within this repository, we distribute the physiologically-based whole-body models for pregnant individuals published in [1,2,3,4,5,6,7]. Additionally, this repository contains the refined passive transports building block which was used to build pregnancy PBPK models with different unbound drug fractions in maternal and fetal organism as described in [8] as well as the *in silico* cotyledon perfusion model presented in [9].

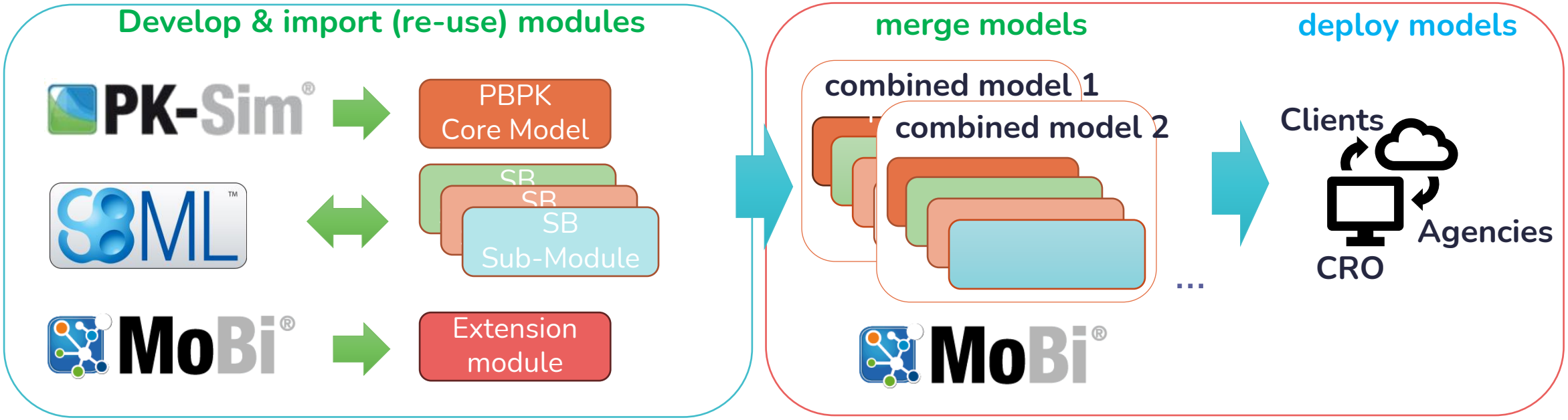
The pregnancy (and postpartum) PBPK model for amoxicillin published in [10] can be found here.

The pregnancy model structure comprises per default 27 compartments, including nine pregnancy-specific compartments as shown in the schema below.

Problem: maintenance

- Growing model complexity makes maintenance and debugging challenging
- Models are developed for a specific use case with limited re-usability, or…
- Integration into new models is tedious and requires multiple manual steps
- Maintenance (e.g., update to new PK-Sim version)

The future of QSP Platform management - Modularization



02

Modularization

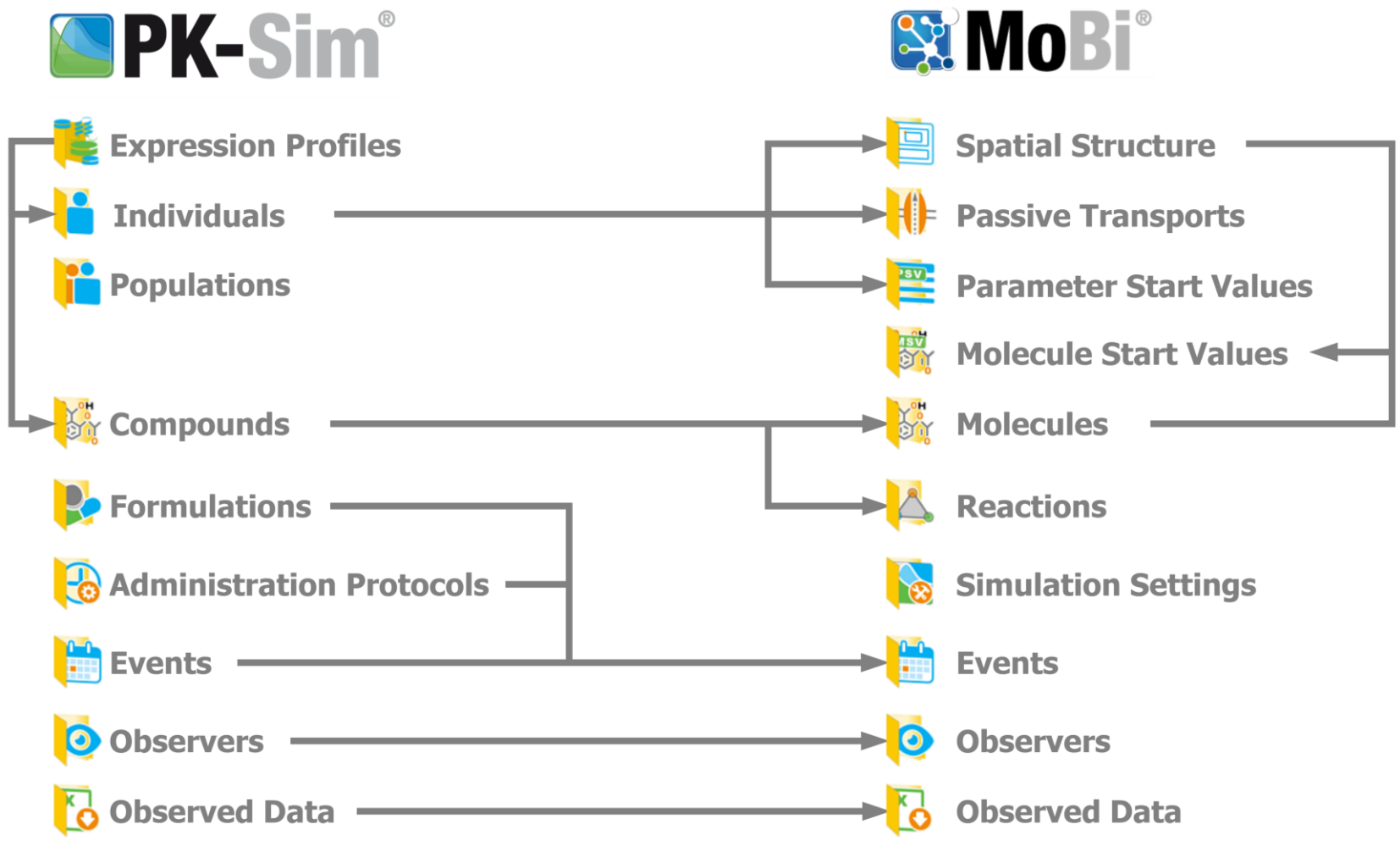
Implementation in OSPS v12

Version 12 is work in progress

Shown GUI and functionality may change in the final release

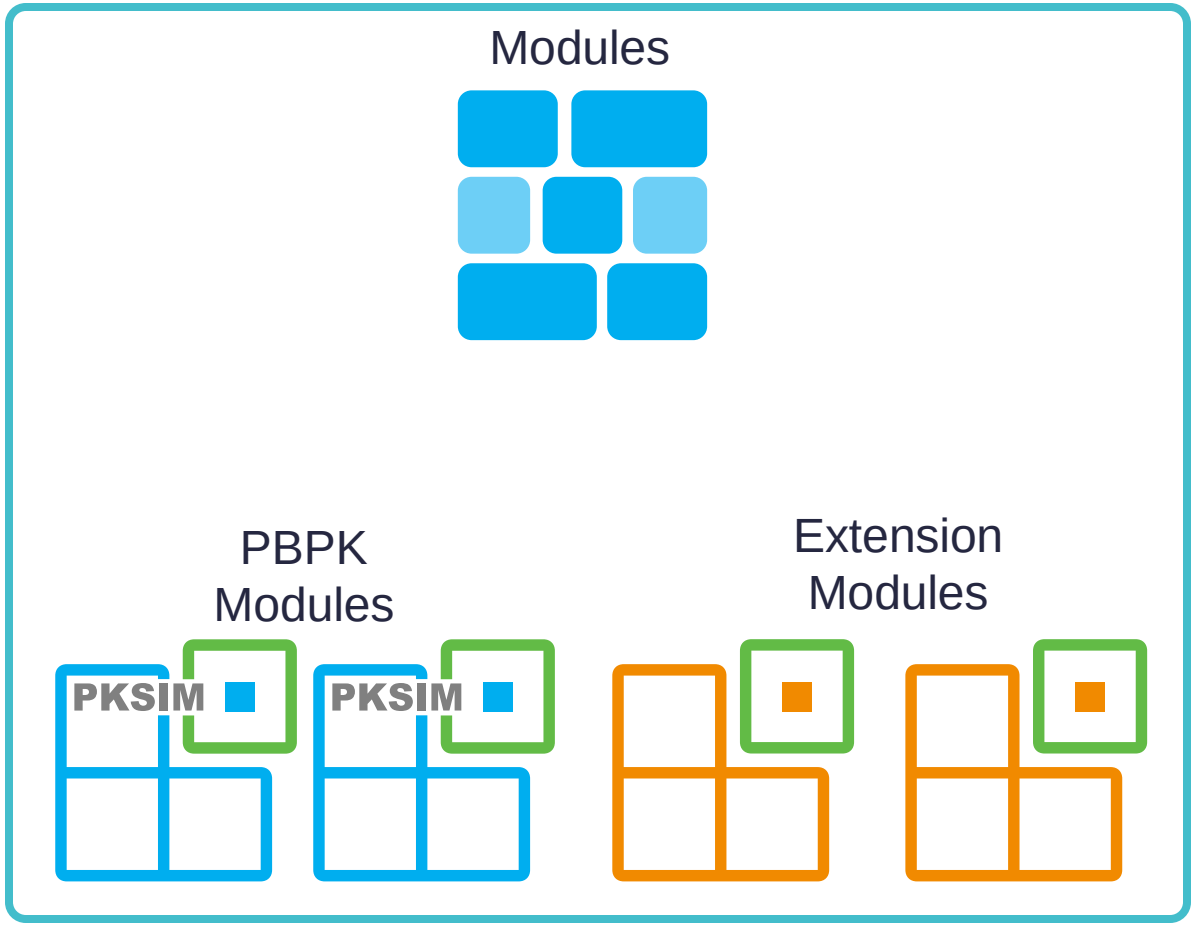
Project organization

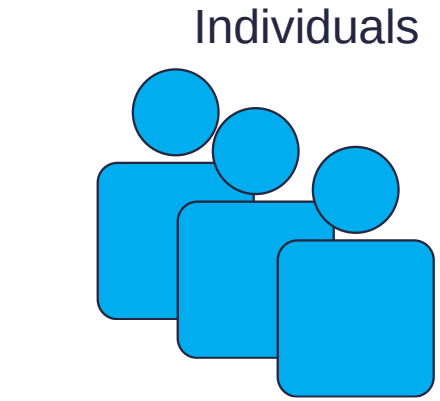
Status quo (<v12)

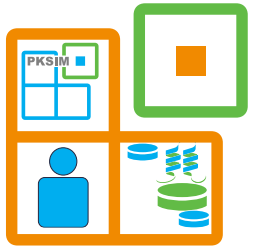


Project organization

MoBi v12



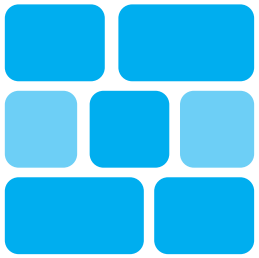


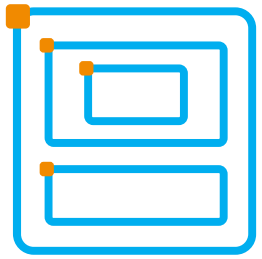


Project organization

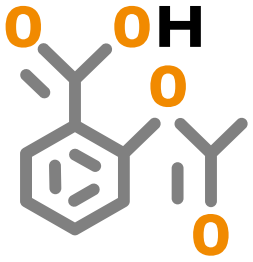
MoBi v12

A module

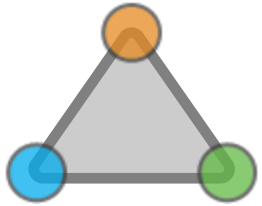




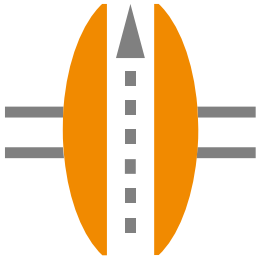
Spatial Structure



Molecules



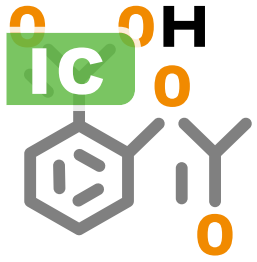
Reactions



Passive Transports

Observers

Events



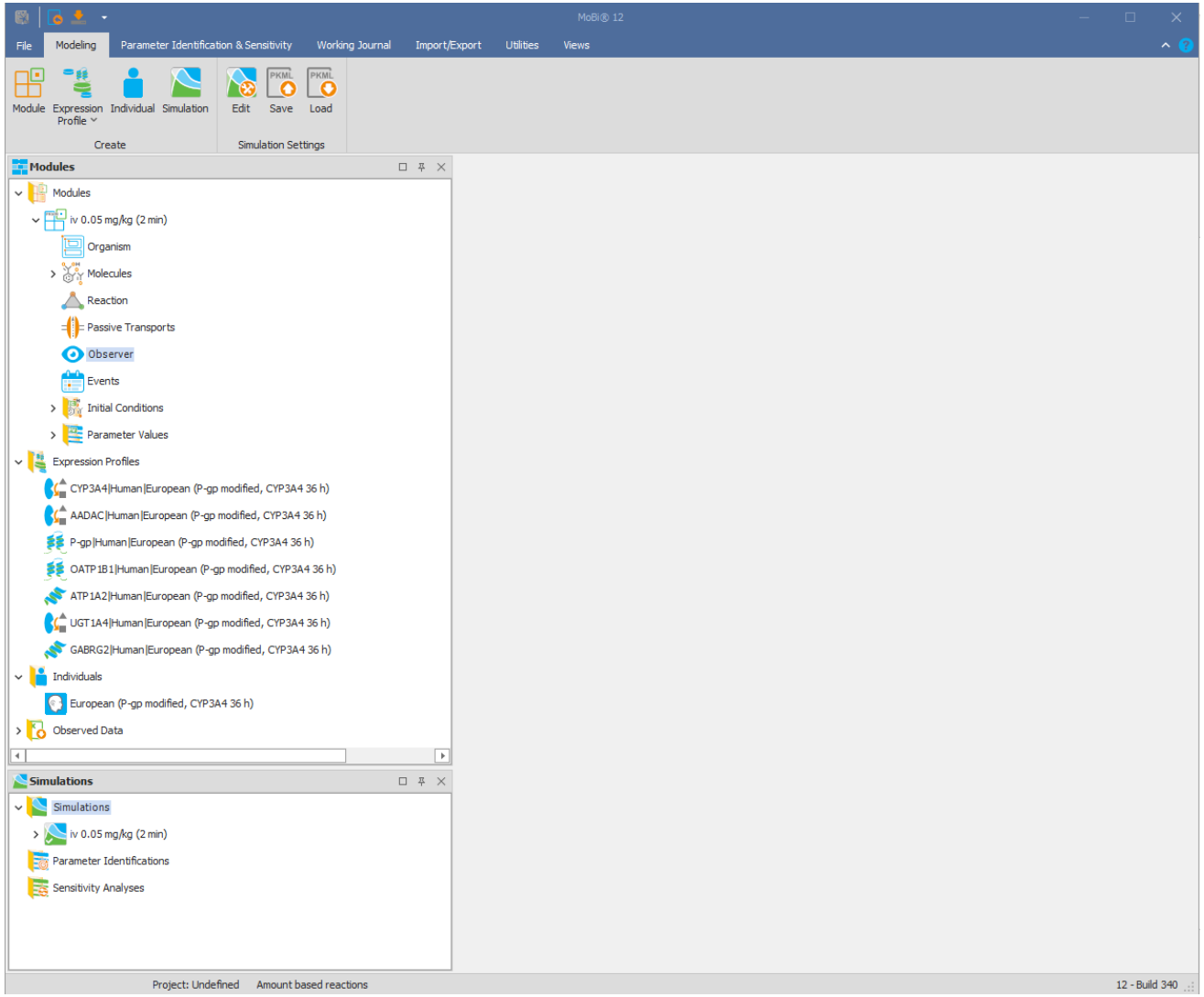
Initial Conditions

Parameter Values

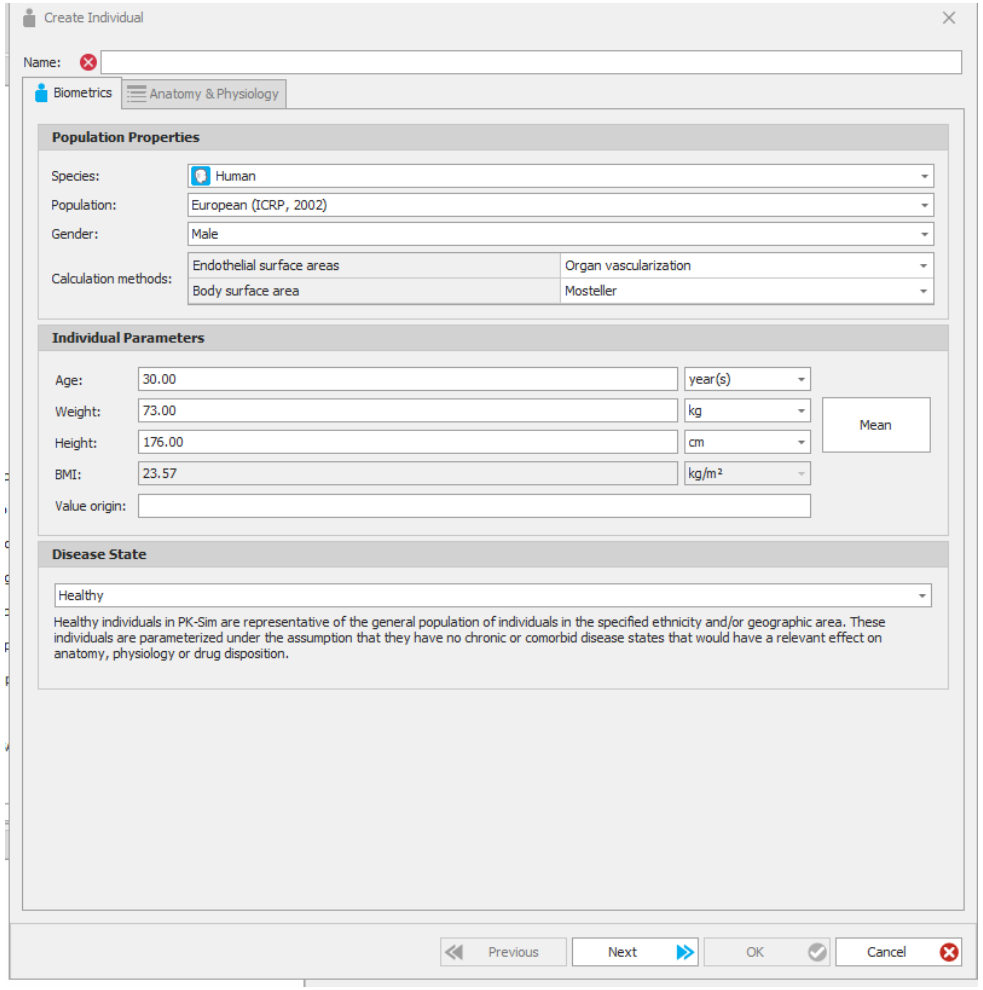
Project organization

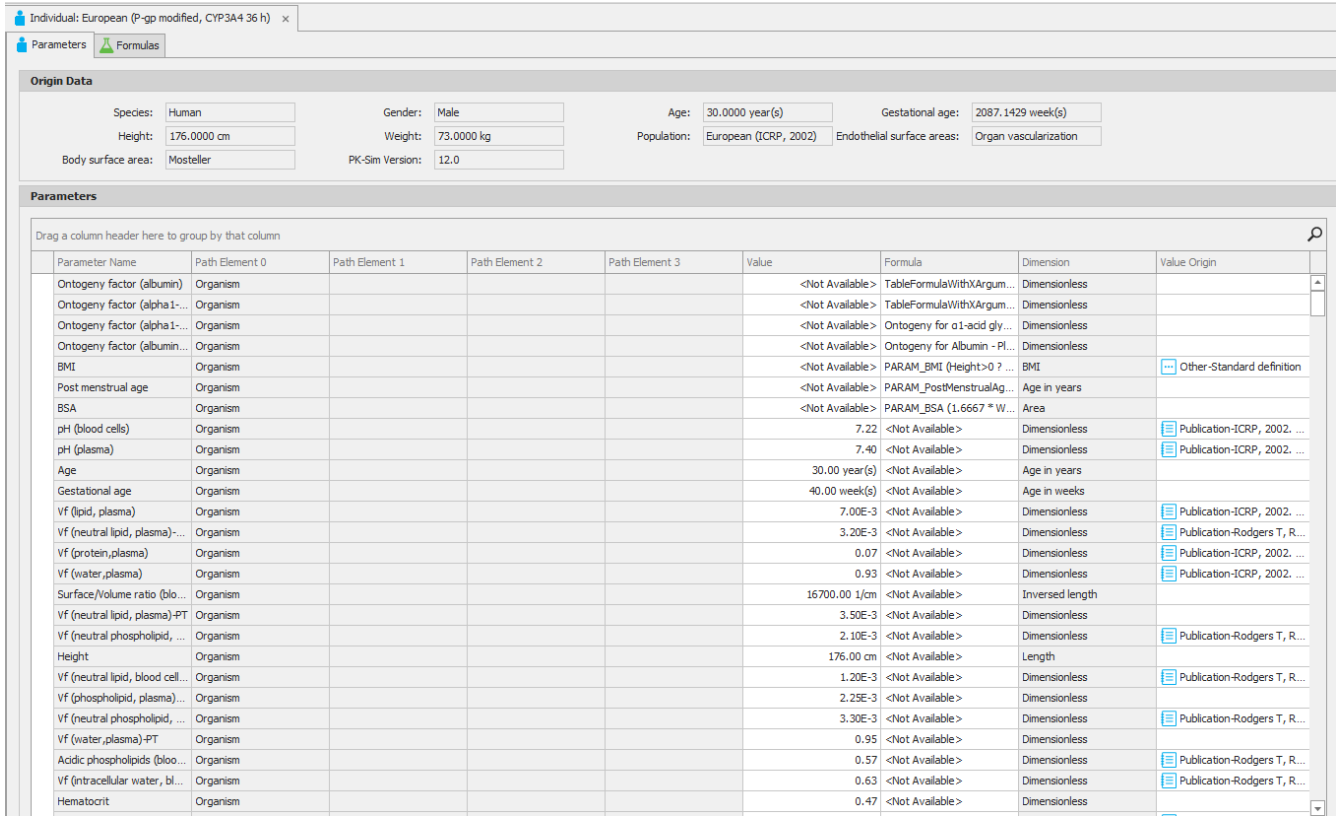
PK-Sim to MoBi transfer

- One PK-Sim module
- Expression Profile(s)
- Individuals

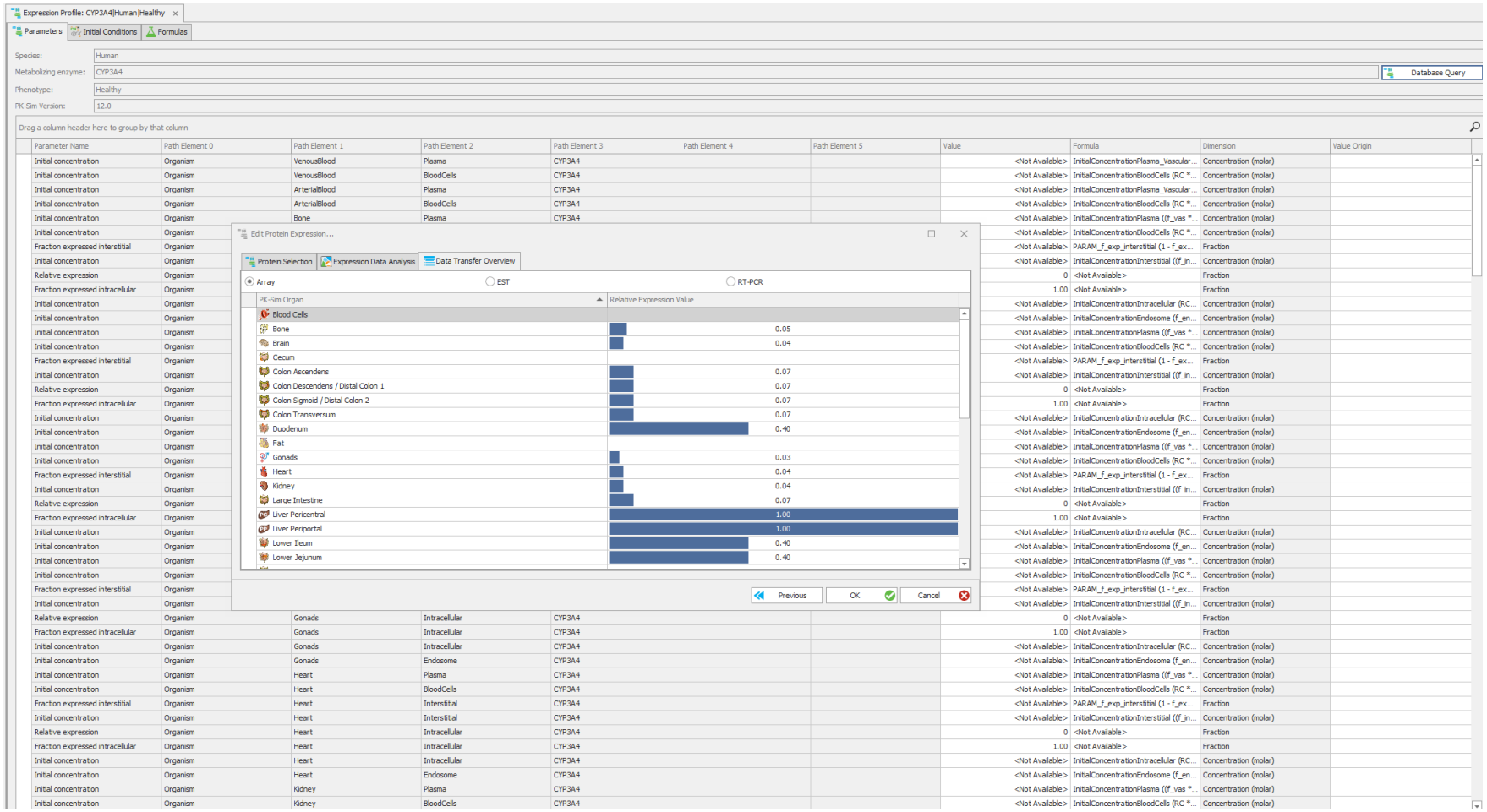


Individuals





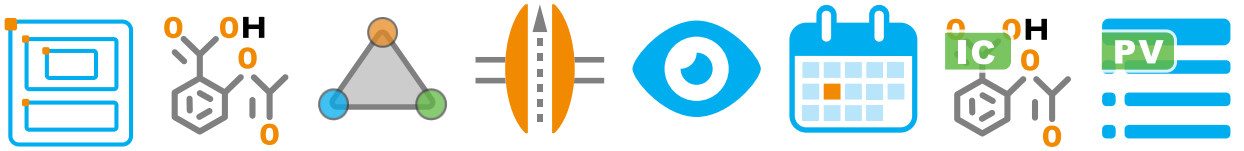
Expression Profiles



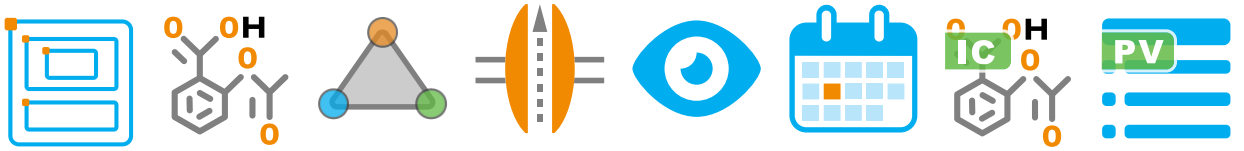
Project organization

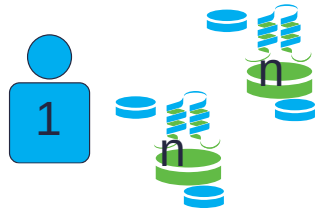
Examples of combination

PKSIM CompoundA



PKSIM CompoundB



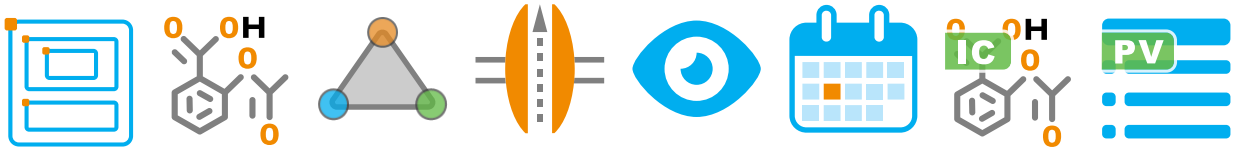


30 years old male
Normal metabolizer

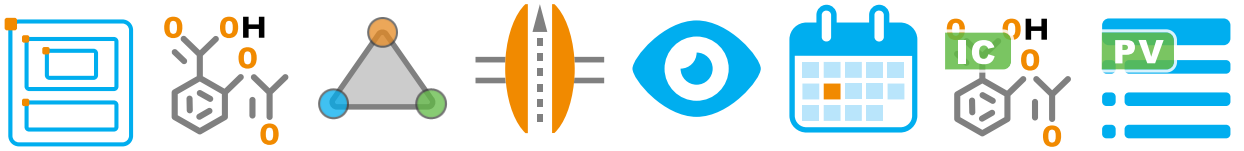
Project organization

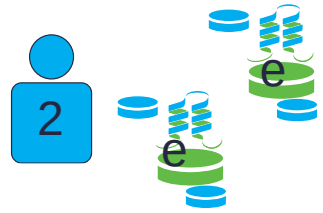
Examples of combination

PKSIM CompoundA



PKSIM CompoundB





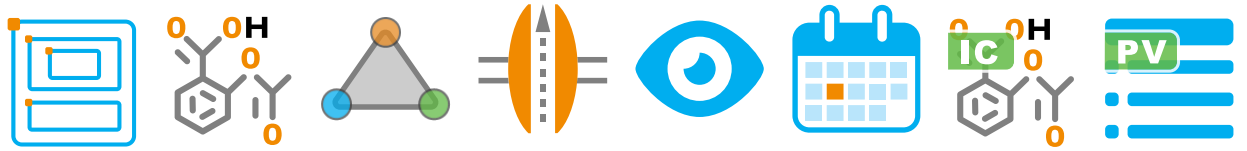
31 years old female
Extensive metabolizer

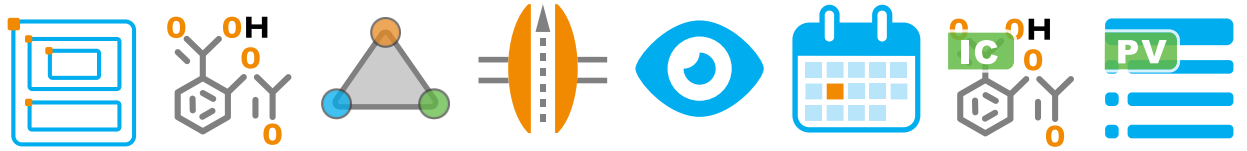
Project organization

Examples of combination

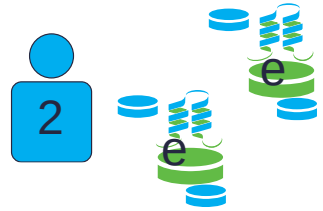
PKSIM CompoundA

PKSIM CompoundB

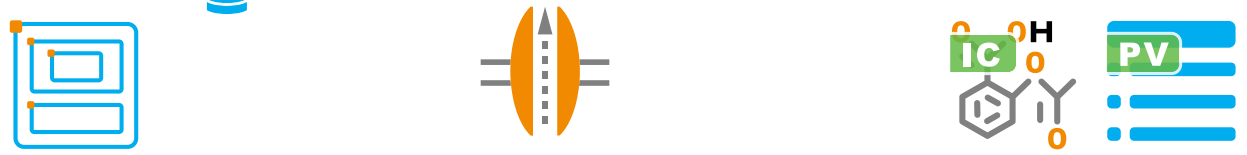




31 years old female
Extensive metabolizer



Pregnancy

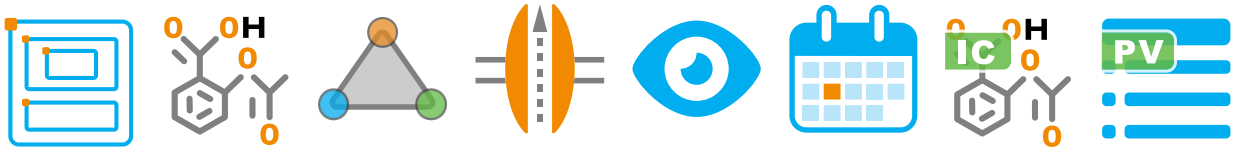


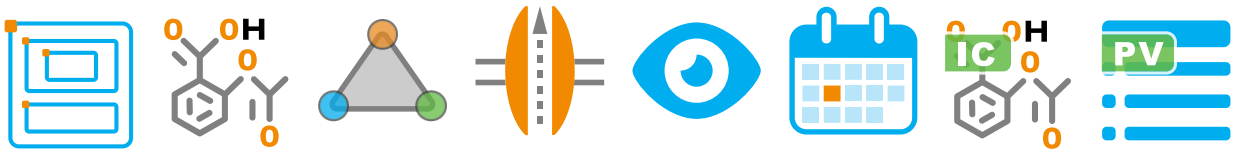
Project organization

Examples of combination

CompoundA

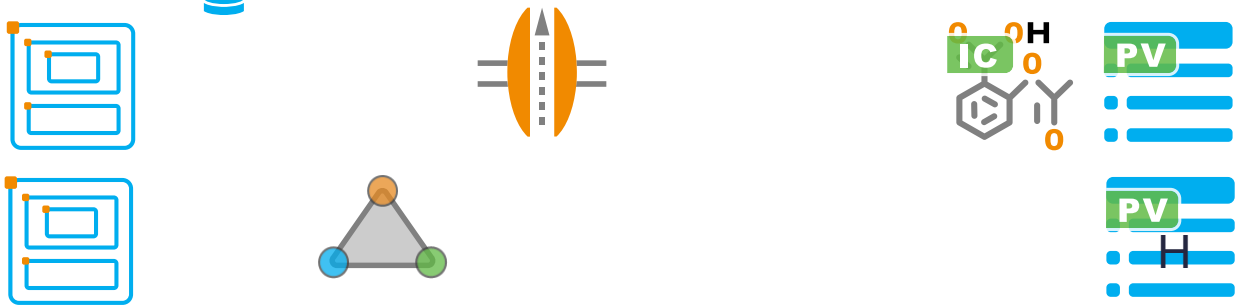
CompoundB





31 years old female
Extensive metabolizer

Pregnancy



CompoundA effect model
Healthy population

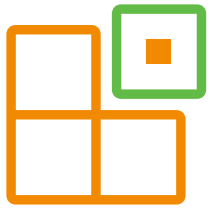
Project organization

Examples of combination

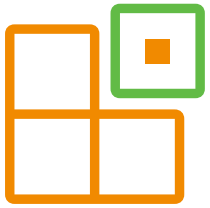
CompoundA

CompoundB

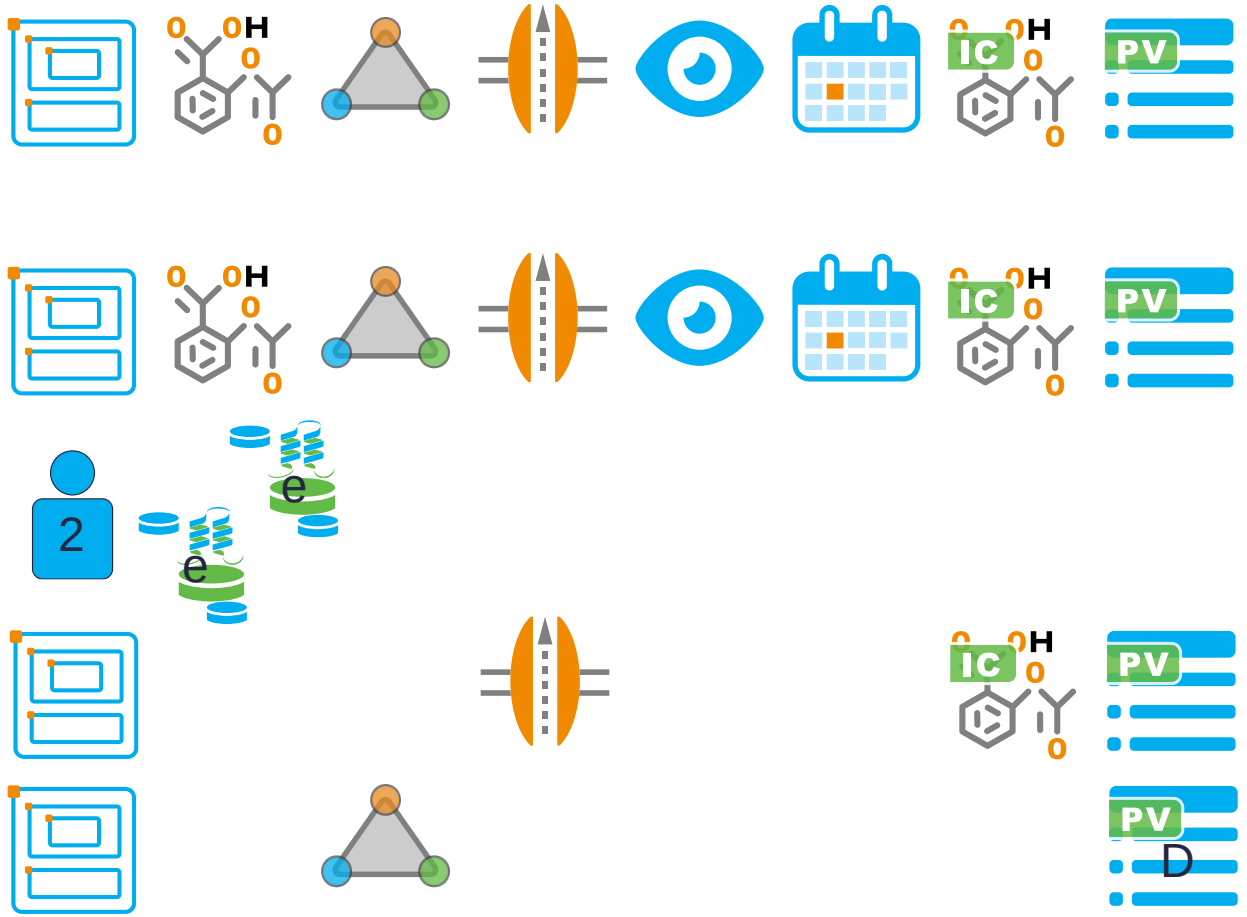
31 years old female
Extensive metabolizer



Pregnancy



CompoundA effect model
Disease population



Use case

- Pregnancy model

Use case – pregnancy model

- How to implement pregnancy as a module?
- Based on PBPK model by Dallmann et al.
  https://github.com/Open-Systems-Pharmacology/Pregnancy-Models

- Goal: a module that can be used with any PBPK model with minimal changes!

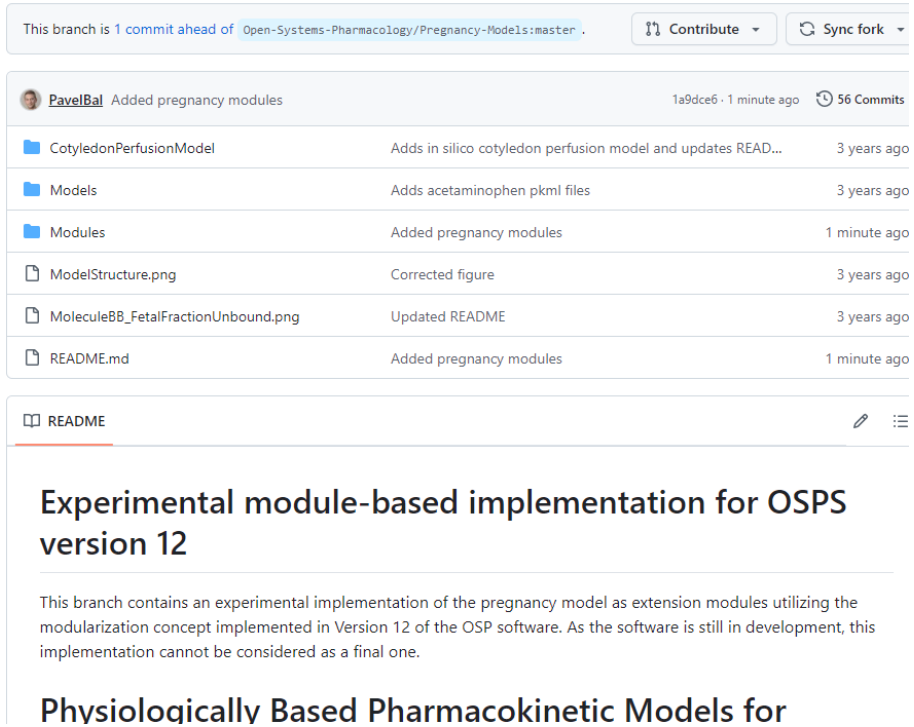
Human pregnancy module

- General approach – implement all changes as separate modules, not within the PK-Sim modules
- Add new structures (e.g., organs) or overwrite existing structures (e.g., volume parameters of organs)
- Increasing complexity can be implemented as modules building upon each other

Human pregnancy module

The here presented modules are available on GitHub!

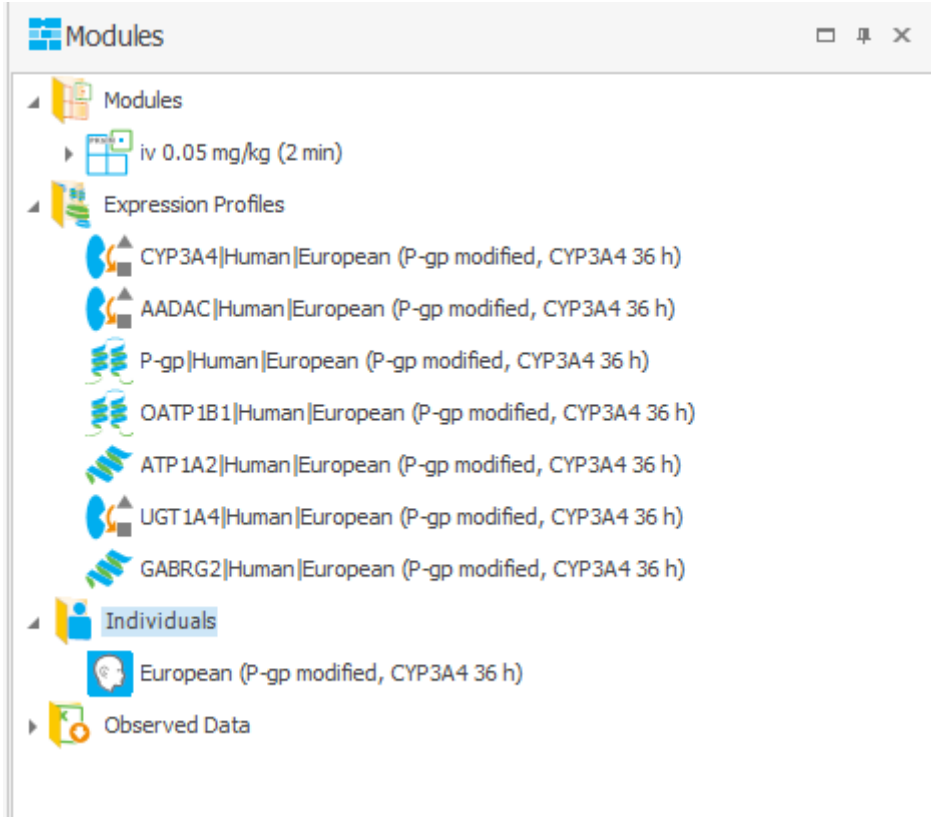
https://github.com/PavelBal/Pregnancy-Models/tree/v12_Modules



Human pregnancy module

Extending a PBPK model

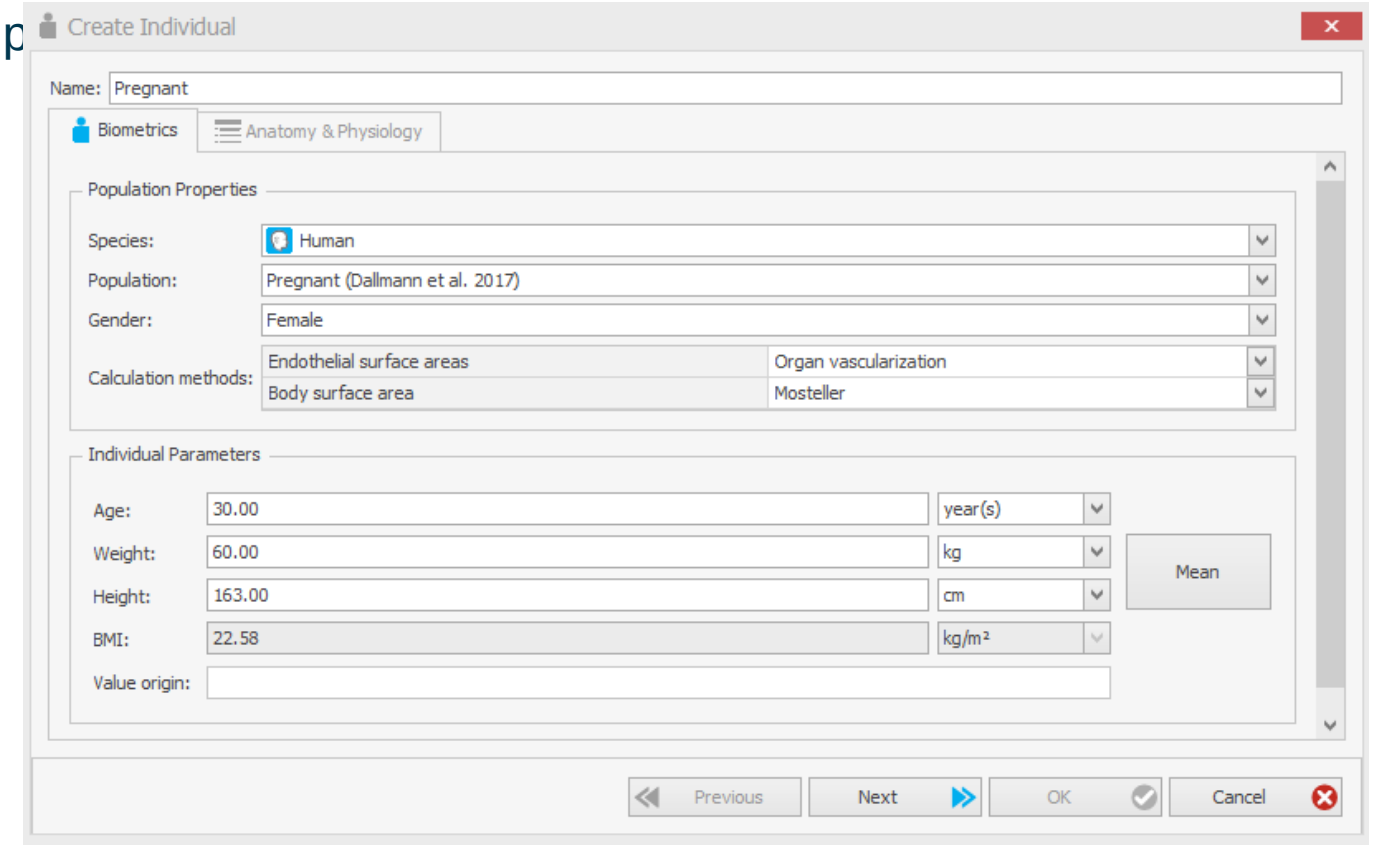
1. Develop a PBPK model in PK-Sim (example – Midazolam)
2. Send the model to PK-Sim



Human pregnancy module

Extending a PBPK model

1. Develop a PBPK model in PK-Sim (examp
2. Send the model to PK-Sim
3. Create a pregnant individual



Human pregnancy module

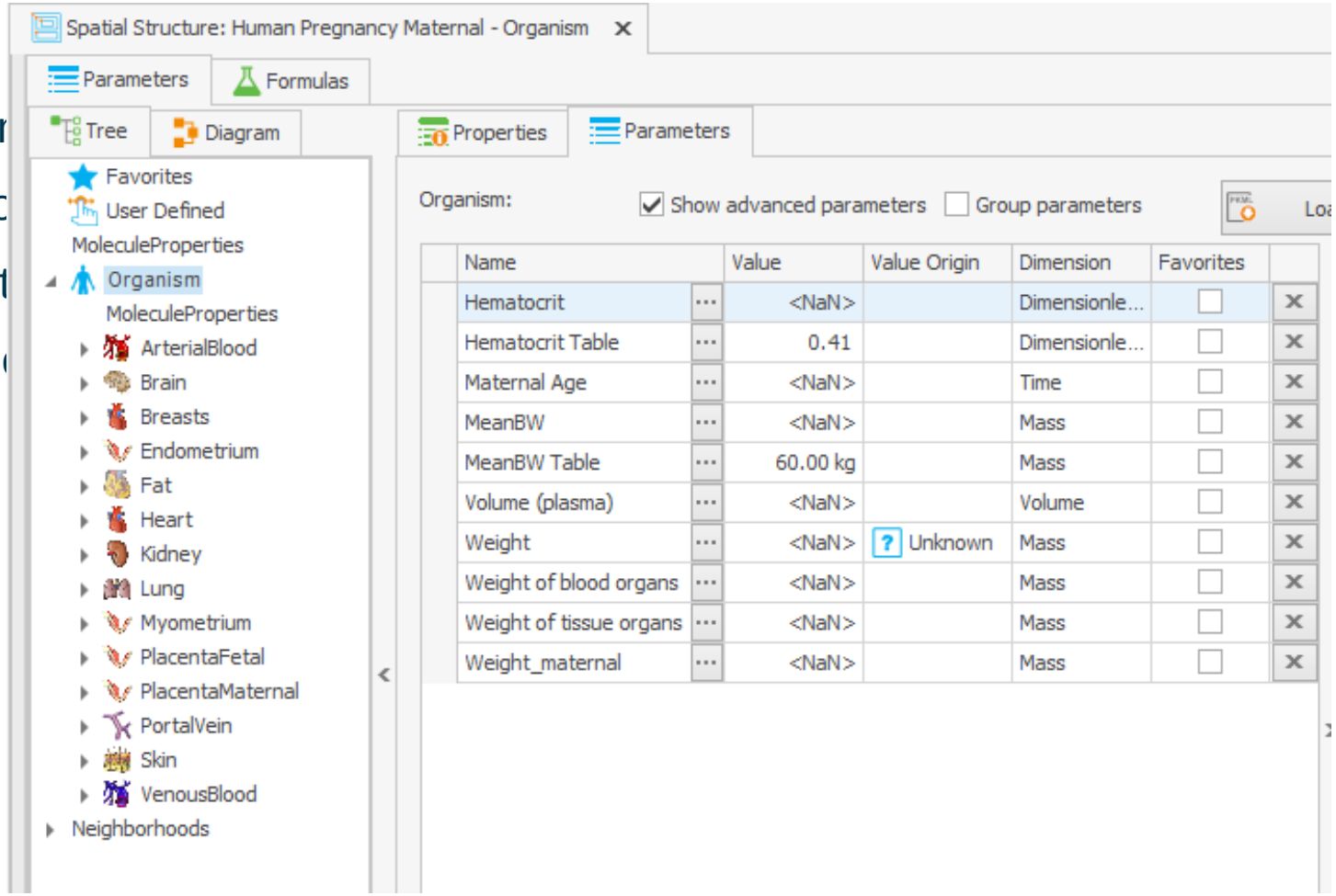
Extending a PBPK model

1. Develop a PBPK model in PK-Sim (example – Midazolam)
2. Send the model to PK-Sim
3. Create a pregnant individual
4. Load the pregnancy modules

Human pregnancy module

Extending a PBPK model

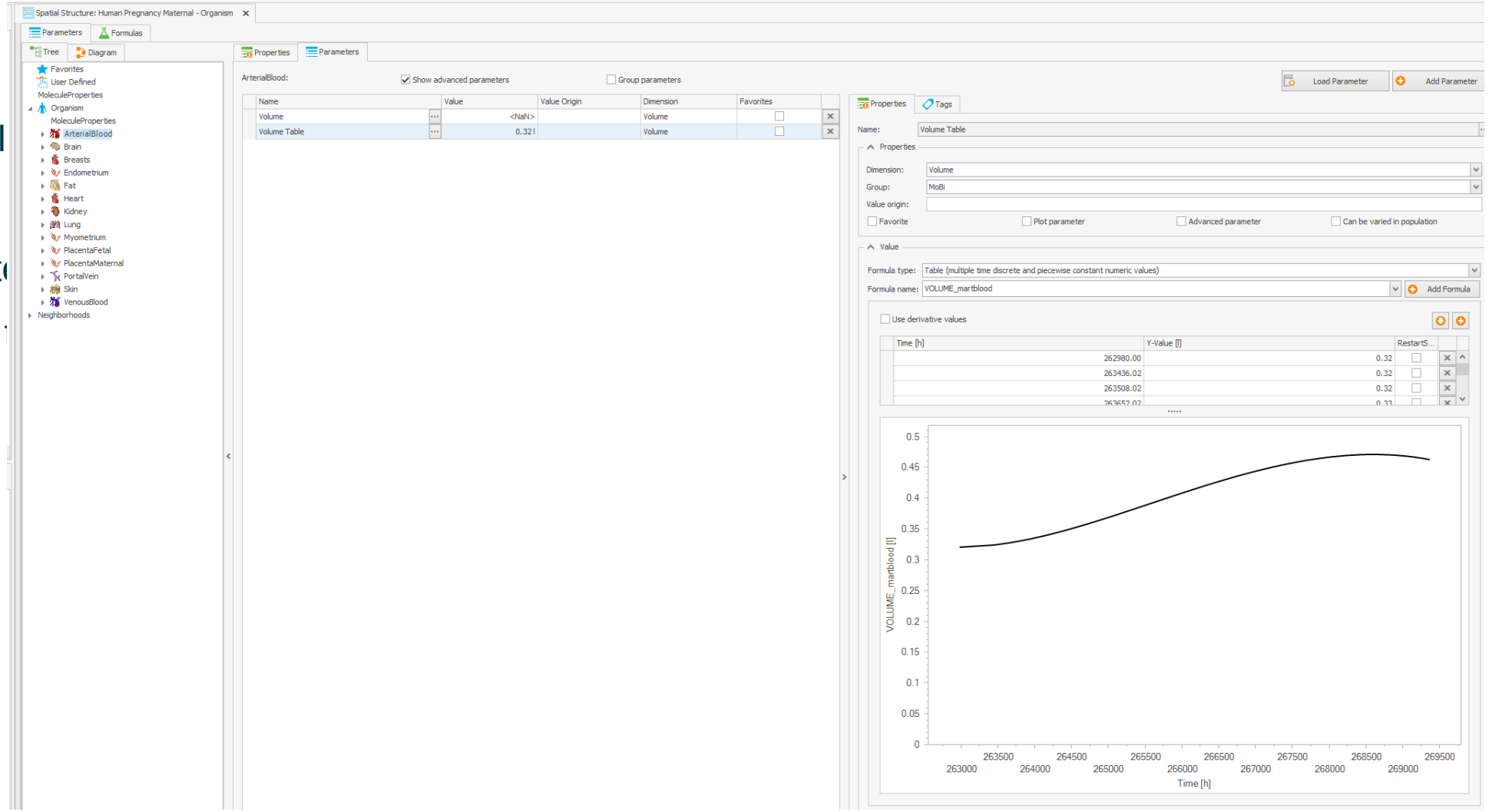
1. Develop a PBPK r
2. Send the model to
3. Create a pregnant
4. Load the pregnan



Human pregnancy module

Extending a PBPK model

1. Devel
2. Send
3. Creat
4. Load



Human pregnancy module

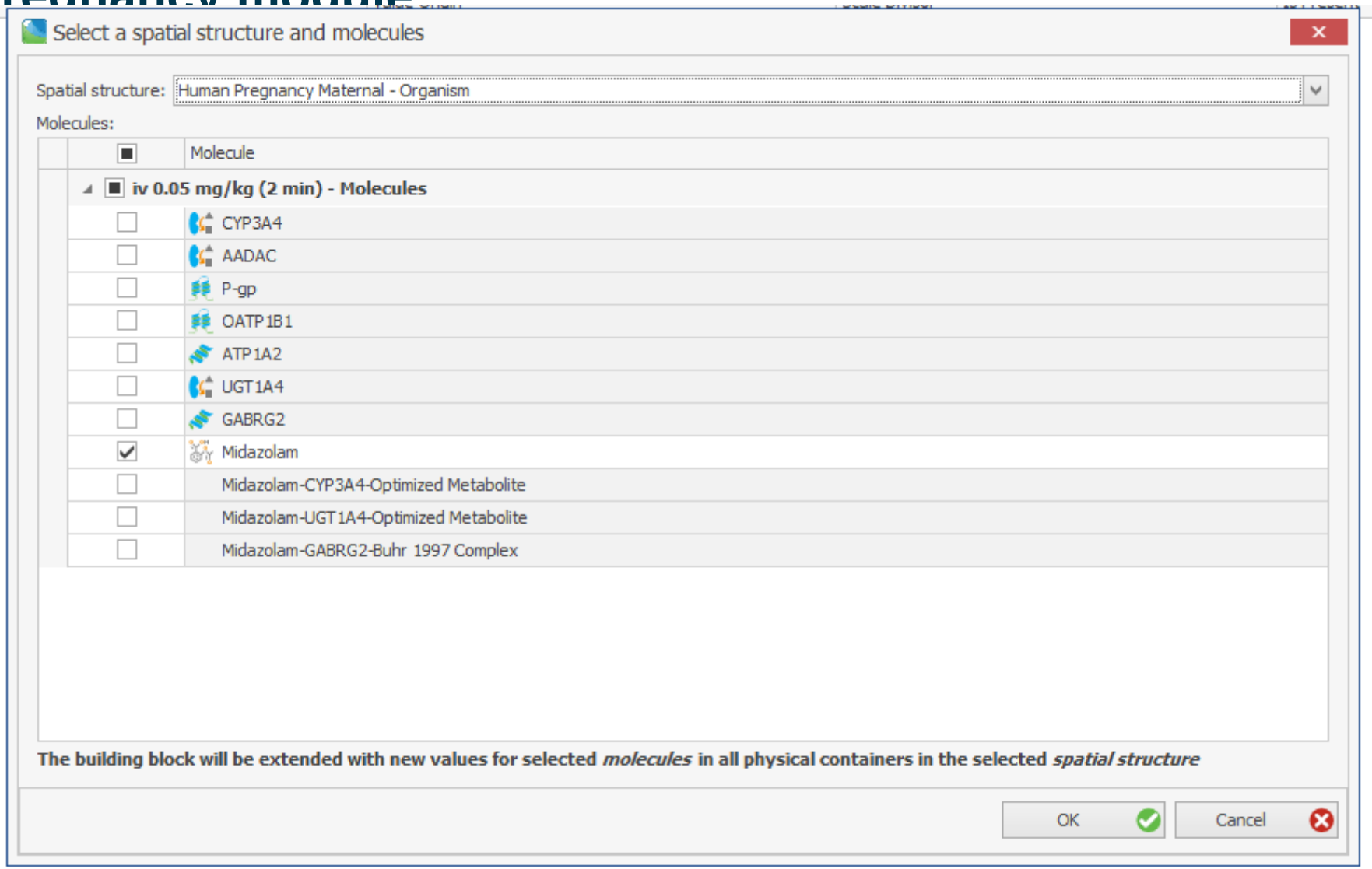
Extending a PBPK model

1. Develop a PBPK model in PK-Sim (example – Midazolam)
2. Send the model to PK-Sim
3. Create a pregnant individual
4. Load the pregnancy modules
5. Extend the Initial Conditions

Human pregnancy module

Extending a

1. Develop
2. Send the
3. Create a
4. Load the
5. Extend th



Human pregnancy module

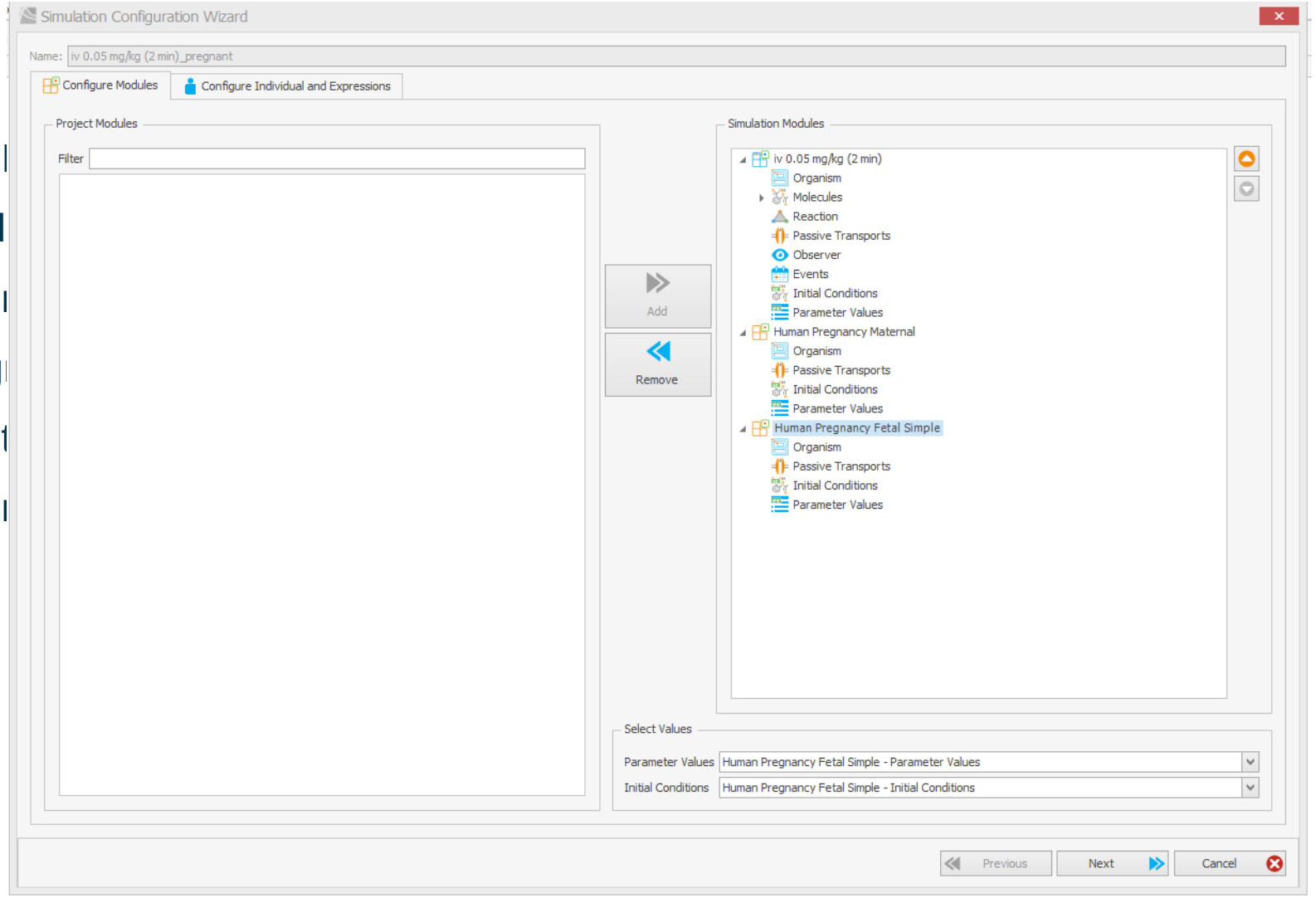
Extending a PBPK model

1. Develop a PBPK model in PK-Sim (example – Midazolam)
2. Send the model to PK-Sim
3. Create a pregnant individual
4. Load the pregnancy modules
5. Extend the Initial Conditions
6. Create a pregnant simulation

Human pregnancy module

Extending a PBPK model

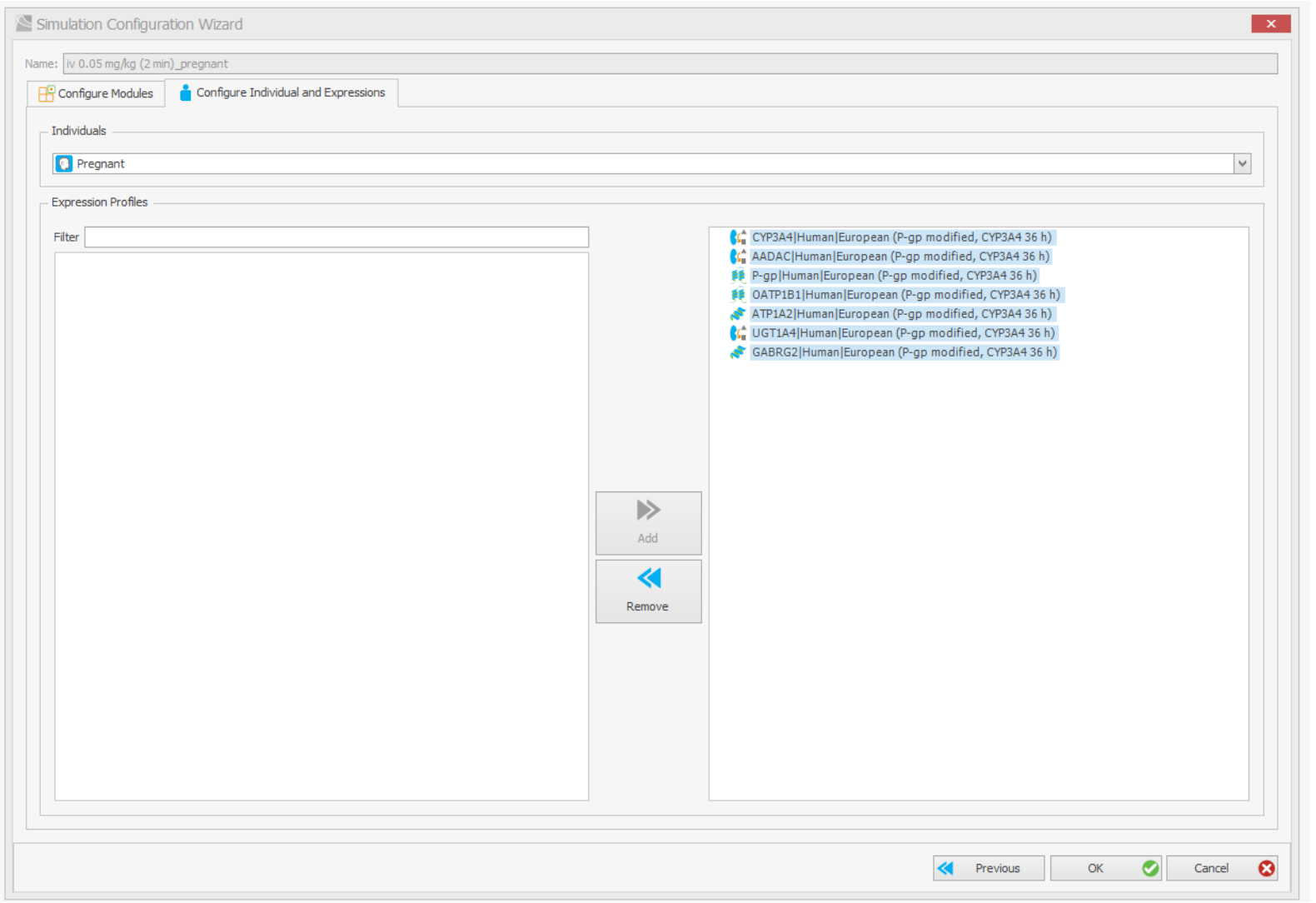
1. Develop a PBI
2. Send the mod
3. Create a pregı
4. Load the pregı
5. Extend the Init
6. Create a pregı



Human pregnancy module

Extending a PBPK model

1. Develop a PBI
2. Send the mod
3. Create a pregi
4. Load the preg
5. Extend the Init
6. Create a pregi



Human pregnancy module

Extending a PBPK model

1. Develop a PBPK model in PK-Sim (example – Midazolam)
2. Send the model to PK-Sim
3. Create a pregnant individual
4. Load the pregnancy modules
5. Extend the Initial Conditions
6. Create a pregnant simulation
7. Simulate!

03

QSP model qualification

Outlook

The Future QSP Platform Management: Modularization

Workflow for **Development, Qualification & Delivery**

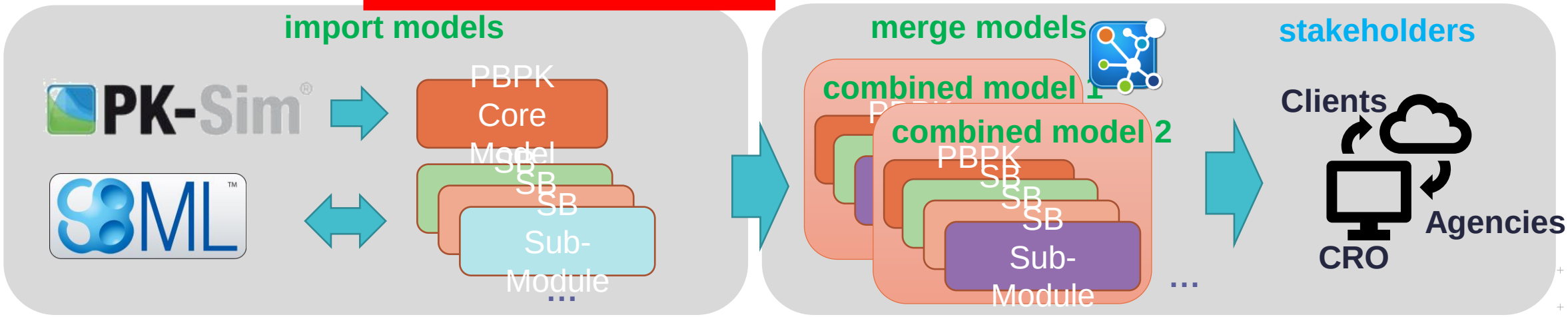
Model Development → Model Qualification → Model Application → Model Distribution

- **Model Development**
- **Model Operations** (***SBML*** Import & Export, Compare, Merge, Clone)
- **Parameter Database**

- **Debugging/Consistency**
- Model **Evaluation** Framework
- **Automated re-qualification** framework

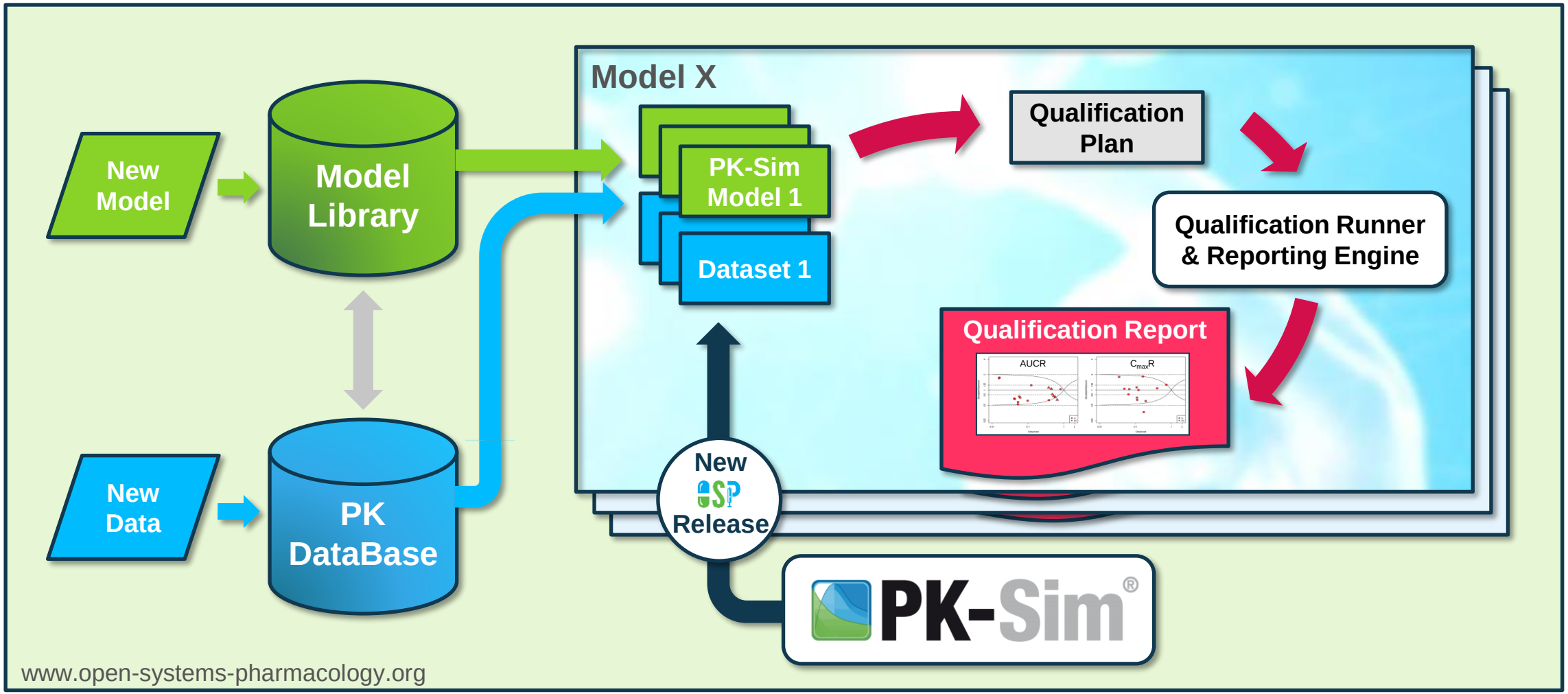
- **Simulation Tools:** Populations, Metrics, …
- **Plotting & Reporting Tools** (Markdown, …)

- **Cloud / Web Interface:** collaborative model presentation & iterative development
- **Model and Data Standards**



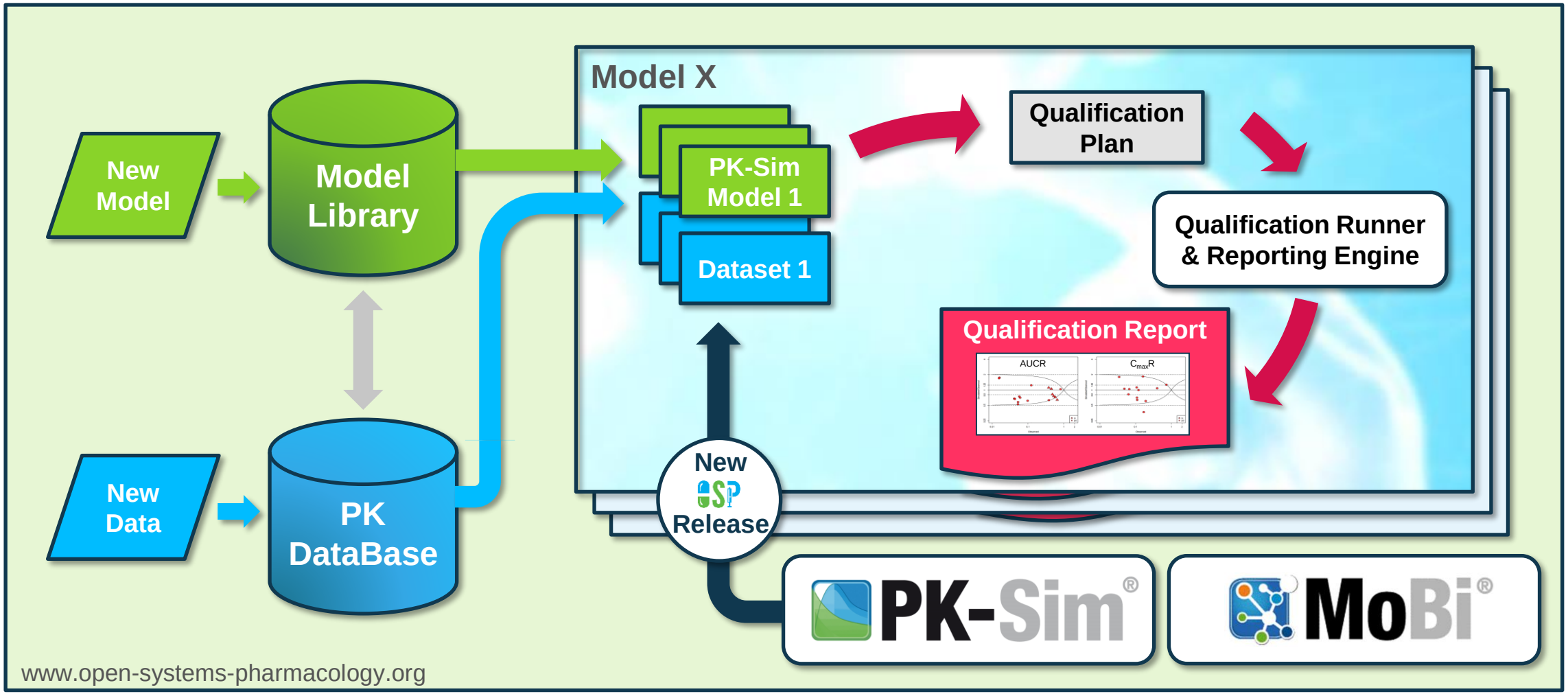
Platform Qualification: Automatic (Re)-qualification Workflow

Sustainable and Agile (Re)-Qualification of Intended Use Scenarios for Regulatory Submissions



Platform Qualification: Automatic (Re)-qualification Workflow

Sustainable and Agile (Re)-Qualification of Intended Use Scenarios for Regulatory Submissions



Platform Qualification

QSP model snapshot

- Key aspect is the **MoBi project snapshot**
  - PK-Sim modules snapshot(s)
  - Extension modules as pkml
  - Observed data
  - Model configurations & models
    - List of all user-defined (parameter) values
- RQ Workflow:
  - Re-create PK-Sim modules
  - Re-create model configurations & models
  - Apply user-defined values
  - Simulate and compare to observed data
  - Create Qualification Report

04

Try v12 now!

The first public beta of Version 12 has just been released!

Get it here:

https://github.com/Open-Systems-Pharmacology/Suite/releases/tag/v12.0_beta1

Getting started with OSMOSES:

https://github.com/Open-Systems-Pharmacology/OSMOSES/blob/develop/Documentation/Getting%20Started.md

Thank you!

Pavel Balazki

esQlabs
we empower life sciences

ESP COMMUNITY CONFERENCE