Open Systems Pharmacology

Software Release Management and (Re-)Qualification Framework

Juri Solodenko

October 2024

- Quality Assurance of Software Changes and Software Releases
  - ✓ OSP Qualification Framework
- OSP Release Planning and Roadmap
  - ✓ Integration of your feature requests

Validation and qualification of software changes and releases

Quality Assurance of the OSP Suite

OSP Suite is an open-source platform that is developed in a fully transparent manner on **GitHub** (the largest hosting platform for open-source software).

Two main steps are used to ensure the quality of the OSP Suite: **Validation** and **Qualification.**

- **Platform Validation**: refers to the process of confirming that the PBPK software platform accurately represents the mathematical models and algorithms it is intended to implement.

  This includes verifying that the software correctly performs the calculations and simulations based on the underlying physiological and pharmacokinetic principles.

- **Platform Qualification** for intended use: involves demonstrating that the PBPK software platform is suitable for the specific research or regulatory purpose for which it is intended.

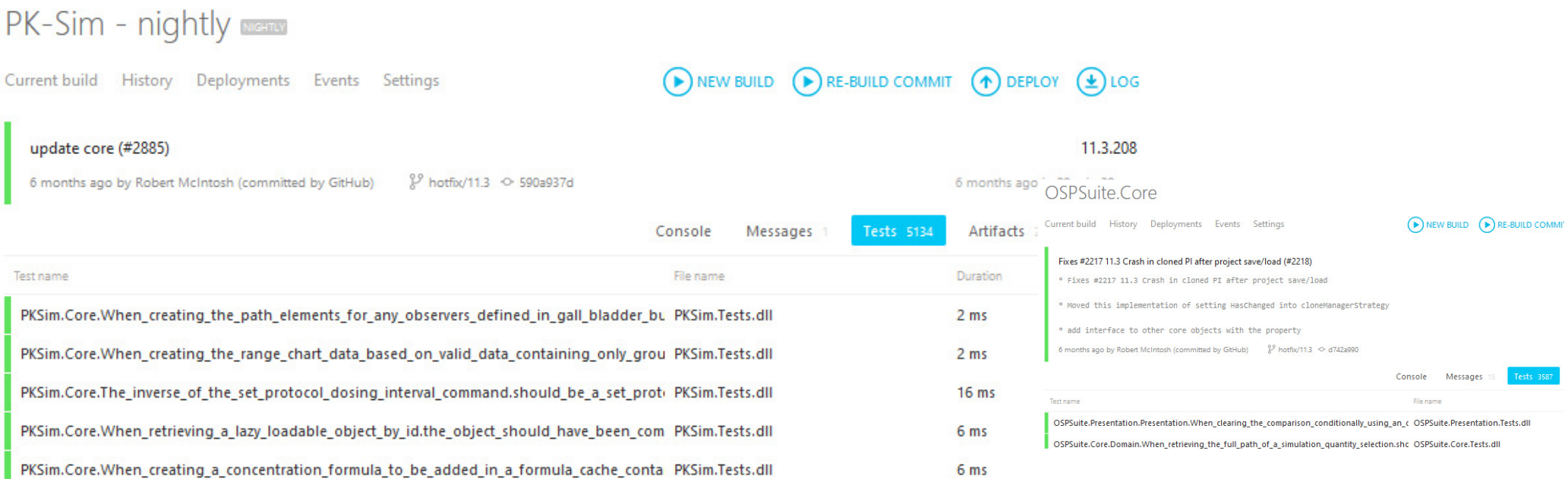
  This goes beyond general validation and includes assessments of the platform's features, functionalities and performance metrics in the context of specific use cases.

  For example, if a platform is intended to predict drug-drug interactions, qualification would include demonstrating that it can accurately model and predict these interactions for a range of compounds.

Quality Assurance of the OSP Suite: Validation

OSP Suite is validated by executing the extensive library of test cases that grows with each new feature release, including manual and automated testing with validated programs. This includes:

1. Automated testing of the correct behavior of software modules.
   - Tests (unit tests, integration tests...) are triggered with every software build (e.g. about 11600 automated tests for the 11.3 release).
   - New changes are integrated only if all tests are passed.
   - Full test logs for every software build and release are documented on GitHub and available for anyone to view.



Quality Assurance of the OSP Suite: Validation

OSP Suite is validated by executing the extensive library of test cases that grows with each new feature release, including manual and automated testing with validated programs. This includes:

1. Automated testing of the correct behavior of software modules.

2. Automated comparison of simulation results between software versions for specific combinations of compounds, organisms, calculation methods and model options..

	File	Modell	Simulations		Individual							Compound					Application		
2			155	Params	Population	Gender	Age	Aging	Enzymes	Transp.	Bind.	Type	pKa	Partition	Permeab.	Process	Dosing	Type	Formulation
43	Human_SingleORAL_Weibull_AsSuspention	4Comp	3	fu/MW/Lipo	ICRP_2002	MALE	30	---	---	---	---	small	Acid	RR	Standard		Single	Oral	Weibull
44	Human_UncompetitiveInhibition	4Comp		---	ICRP_2002	MALE	30		CYP3A4			small	Acid	RR	Standard	Specific_MM CYP3A4	DI_12_12	IntravenousBolus	
45			1	---				---		---	---	small	Acid	RR	Standard	UncompetitiveInhibition CYP3A4	DI_6_6_6_6	IntravenousBolus	
46	Minipig_SingleORAL_Dissolved	4Comp	3	fu/MW/Lipo	Minipig	---	---	---	---	---	---	small	Acid	RR	Standard		Single	Oral	Dissolved
47	Monkey_SingleORAL_Dissolved	4Comp	3	fu/MW/Lipo	Monkey	---	---	---	---	---	---	small	Acid	RR	Standard		Single	Oral	Dissolved
48	Mouse_SingleORAL_Dissolved	4Comp	3	fu/MW/Lipo	Mouse	---	---	---	---	---	---	small	Acid	RR	Standard		Single	Oral	Dissolved
49	Preterm_SingleIV_Age_0_GA_32_CYP3A4	4Comp	1	---	Preterm	MALE	0	X	CYP3A4	---	---	small	Acid	RR	Standard	1stOrder CYP3A4	Single	IntravenousBolus	
50	Preterm_SingleIV_Age_0_GA_32_GFR	4Comp	1	---	Preterm	MALE	0	X	---	---	---	small	Acid	RR	Standard	GFR	Single	Intravenou	Dissolved
51	Preterm_SingleIV_Age_15_GA_32_CYP3A4	4Comp	1	---	Preterm	MALE	0,25	X	CYP3A4	---	---	small	Acid	RR	Standard	1stOrder CYP3A4	Single	IntravenousBolus	
52	Preterm_SingleIV_Age_15_GA_32_GFR	4Comp	1	---	Preterm	MALE	0,25	X	---	---	---	small	Acid	RR	Standard	GFR	Single	Intravenou	Dissolved
53	Rabbit_SingleORAL_Dissolved	4Comp	3	fu/MW/Lipo	Rabbit	---	---	---	---	---	---	small	Acid	RR	Standard		Single	Oral	Dissolved
54	Rat_MultiORAL_6_6_6_6_Dissolved	4Comp	1	---	Rat	---	---	---	---	---	---	small	Acid	RR	Standard		DI_6_6_6_6	Oral	Dissolved
55	Rat_MultiORAL_6_6_12_Dissolved	4Comp	1	---	Rat	---	---	---	---	---	---	small	Acid	RR	Standard		DI_6_6_12	Oral	Dissolved
56	Rat_MultiORAL_8_8_8_Dissolved	4Comp	1	---	Rat	---	---	---	---	---	---	small	Acid	RR	Standard		DI_8_8_8	Oral	Dissolved
57	SingleIV_2Pores_Human	TwoPores	4	Kd (FcRn)_endo C_FcRn_endo(0)	ICRP_2002	MALE	30	---	---	---	---	Large	Acid	Standard	Standard		Single	IntravenousBolus	
58	SingleIV_2Pores_Monkey	TwoPores	3	Kd (FcRn)_endo C_FcRn_endo(0)	Monkey	---	---	---	---	---	---	Large	Acid	Standard	Standard		Single	IntravenousBolus	
59	SingleIV_2Pores_Mouse	TwoPores	4	Kd (FcRn)_endo C_FcRn_endo(0)	Mouse	---	---	---	---	---	---	Large	Acid	Standard	Standard		Single	IntravenousBolus	
	SingleIV_C1_4Comp																		

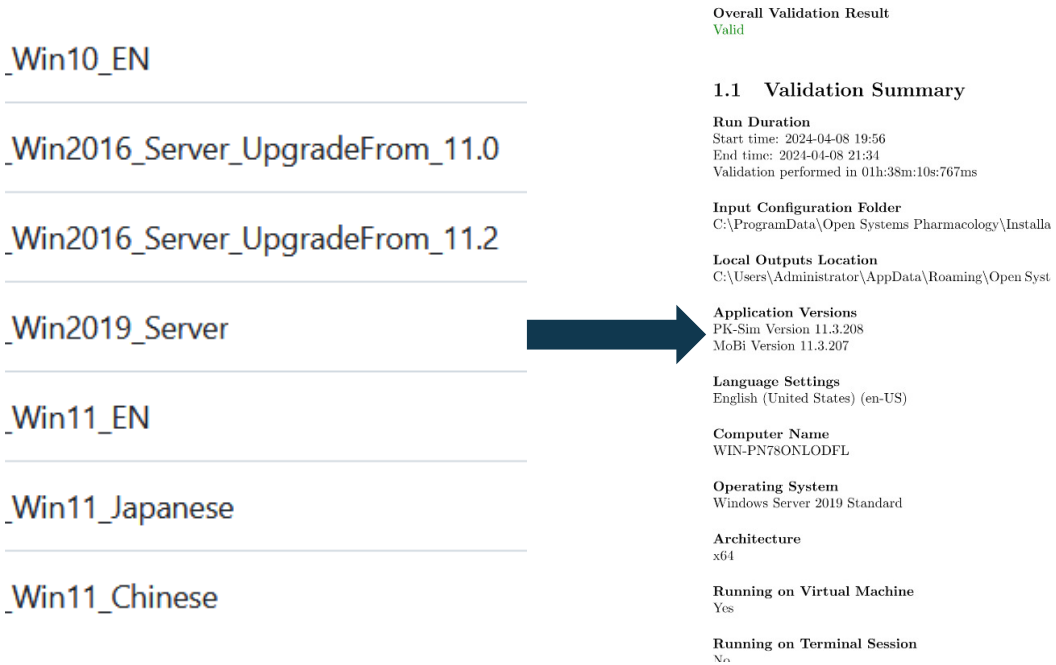
Quality Assurance of the OSP Suite: Validation

OSP Suite is validated by executing the extensive library of test cases that grows with each new feature release, including manual and automated testing with validated programs. This includes:

1. Automated testing of the correct behavior of software modules.

2. Automated comparison of simulation results between software versions for specific combinations of compounds, organisms, calculation methods and model options.

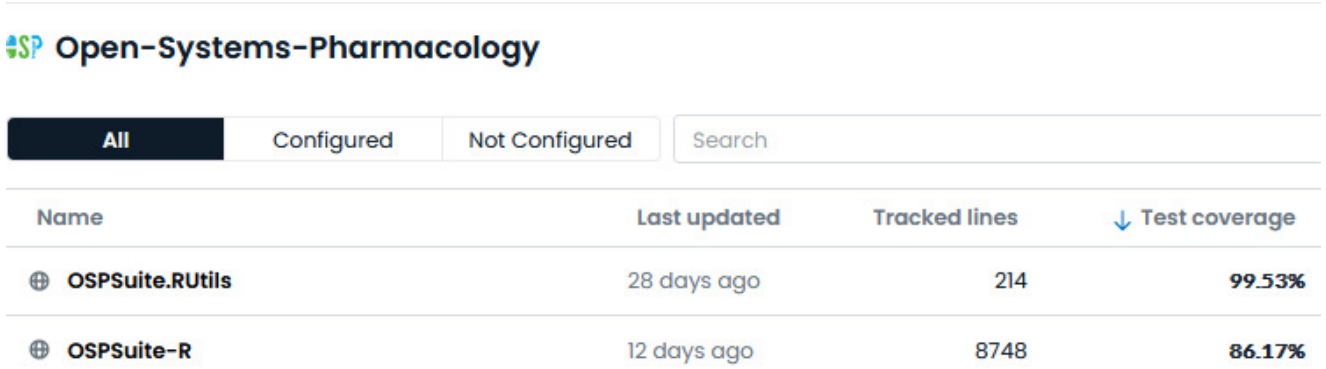
3. Automated testing in different software environments (different operating systems, etc.).



Quality Assurance of the OSP Suite: Validation

OSP Suite is validated by executing the extensive library of test cases that grows with each new feature release, including manual and automated testing with validated programs. This includes:

1. Automated testing of the correct behavior of software modules.
2. Automated comparison of simulation results between software versions for specific combinations of compounds, organisms, calculation methods and model options.
3. Automated testing in different software environments (different operating systems, etc.).
4. Automated code quality analysis (e.g. static code analysis, test coverage).



Quality Assurance of the OSP Suite: Validation

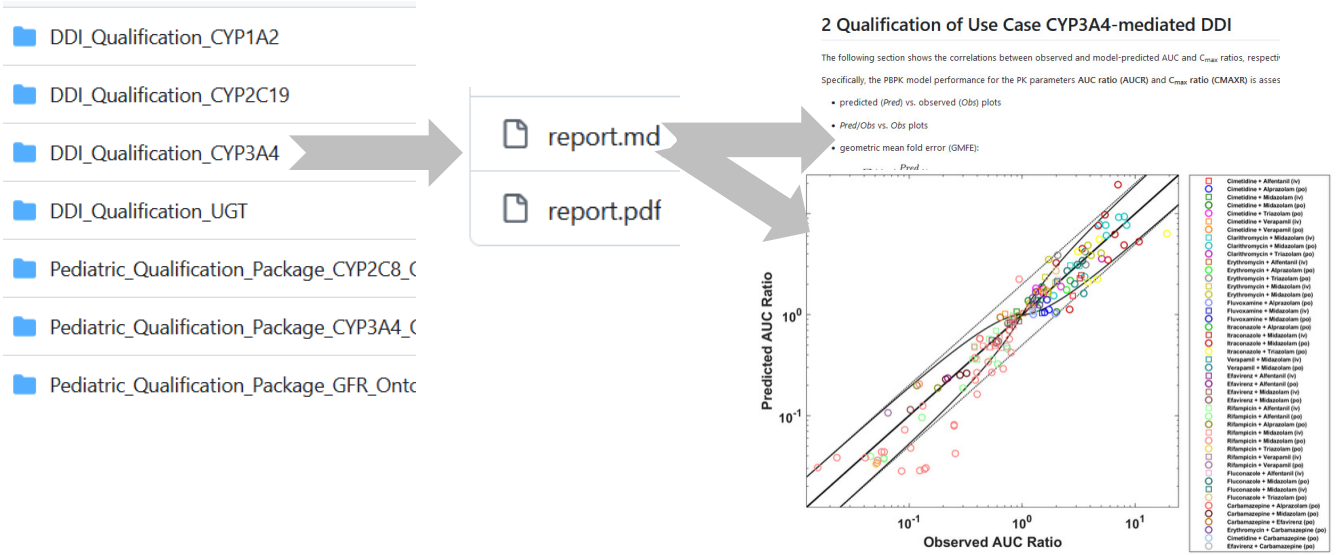
OSP Suite is validated by executing the extensive library of test cases that grows with each new feature release, including manual and automated testing with validated programs. This includes:

1. Automated testing of the correct behavior of software modules.
2. Automated comparison of simulation results between software versions for specific combinations of compounds, organisms, calculation methods and model options.
3. Automated testing in different software environments (different operating systems, etc.).
4. Automated code quality analysis (e.g. static code analysis, test coverage).
5. (Manual) testing of new features by scientific experts.

Quality Assurance of the OSP Suite: Qualification

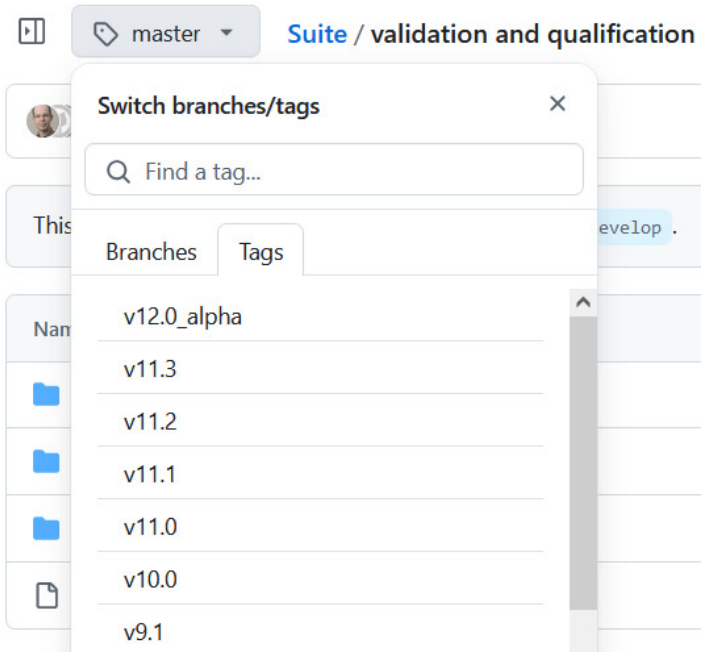
- Qualification of OSP Suite functionality is accomplished by running "Qualification for Intended Use" scenarios and generating qualification reports via the fully automated **OSP (RE-)Qualification Framework**.

- Once a use case is qualified for its purpose, changes and updates to the PBPK platform (e.g., model structure adjusted, model parameterization changed, new software version released) require **Requalification**.



Validation and Qualification

- All final validation/qualification test logs for the latest release of the Suite are documented on GitHub under https://tinyurl.com/osp-release-validation
- Test protocols for previous Suite releases are stored in the tags of the same repository https://tinyurl.com/osp-release-tags



README.md

OSP Qualification Reports

(PBPK) Model Evaluation Reports

CKD Population Qualification Report

Unit and integration tests

- PK-Sim
- MoBi
- Installation Validator
- OSPSuite.Core
- OSPSuite.SimModel
  - Windows
  - Linux (Ubuntu 18.04)
- OSPSuite.SimModel.Solver.CVODES
- OSPSuite.FuncParser
  - Windows
  - Linux (Ubuntu 18.04)
- OSPSuite.DataBinding.DevExpress
- OSPSuite.DataBinding
- OSPSuite.Utility
- OSPSuite.TeXReporting
- OSPSuite.Serializer

Batch comparison tests

Platform tests

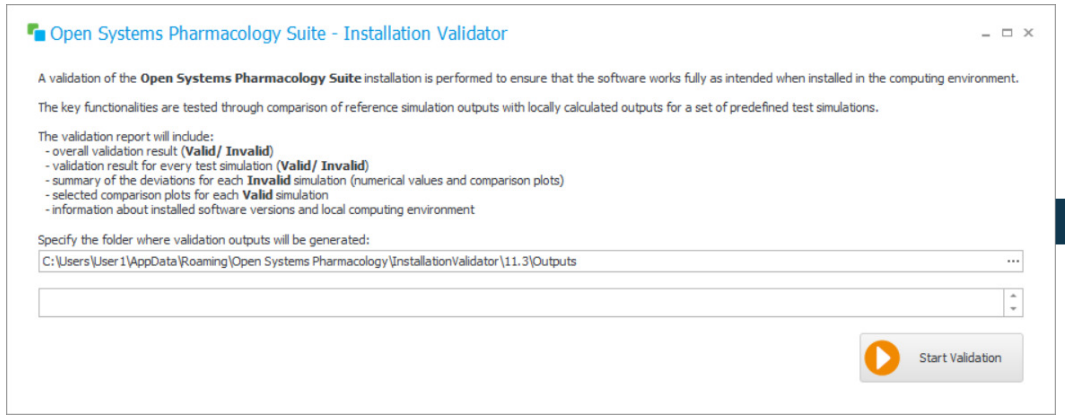
Validation and Qualification

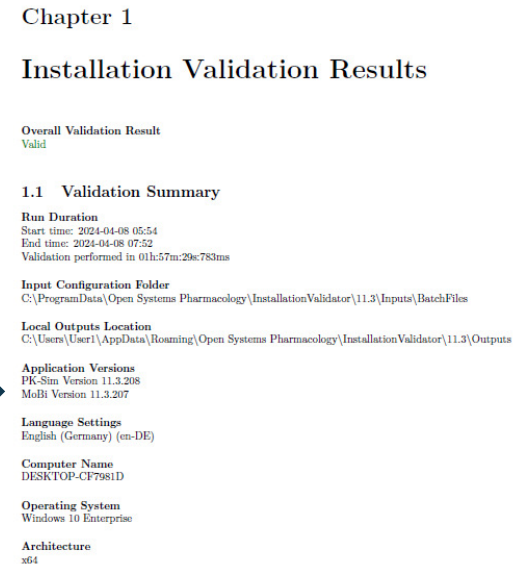
Installation validation on a target system is ensured by the fully automated Installation Validator tool installed as part of the OSP Suite.

- A set of predefined (PBPK) models is being created and simulated in a target modeling environment.

File	Modell	Simulations		Individual							Compound					Application		
		155	Params	Population	Gender	Age	Aging	Enzymes	Transp.	Bind.	Type	pKa	Partition	Permeab.	Process	Dosing	Type	Formulation
Beagle_SingleORAL_Dissolved	4Comp	3	fu/MW/Lipo	Beagle	---	---	---	---	---	---	small	Acid	RR	Standard		Single	Oral	Dissolved
DDI_MultipleCombinations	4Comp	23	---	ICRP_2002	MALE	30	---	Multiple	Multiple	---	small	Multiple	Multiple	Multiple	1st Order/MM Metab. 1st Order/MM Transports ALL DDI Types	Single / Multiple	Oral / IV / UserDef	Dissolved
Dog_MultiORAL_12_12_Dissolved	4Comp	1	---	Dog	---	---	---	---	---	---	small	Acid	RR	Standard		DI_12_12	Oral	Dissolved

- Simulation results are compared to the *reference simulation results*
  (reference simulation results are created and validated during the OSP Release validation).
- A **validation report** is generated for the target environment.
- The whole process is fully automated.





Chapter 1

Installation Validation Results

Overall Validation Result
Valid

1.1 Validation Summary

Run Duration
Start time: 2024-04-08 05:54
End time: 2024-04-08 07:52
Validation performed in 01h:57m:29s:783ms

Input Configuration Folder
C:\ProgramData\Open Systems Pharmacology\InstallationValidator\11.3\Inputs\BatchFiles

Local Outputs Location
C:\Users\User1\AppData\Roaming\Open Systems Pharmacology\InstallationValidator\11.3\Outputs

Application Versions
PK-Sim Version 11.3.208
MoBi Version 11.3.207

Language Settings
English (Germany) (en-DE)

Computer Name
DESKTOP-CP7981D

Operating System
Windows 10 Enterprise

Architecture
x64

OSP Qualification Framework

OSP Qualification Framework: Main Workflow

OSP Qualification Framework provides an automated workflow that generates comprehensive **qualification reports** based on prespecified dedicated **qualification plans**.

- **Qualification report** is a document structured in chapters, starting with a short description of the scientific background of the qualification scenario, followed by a short methodological description (e.g. modeling strategy, available data) and a presentation of the results.

- **Qualification plan**
  - is a human-readable document (in JSON format) that contains all the information needed to generate such a qualification report.
  - defines how static text module content and dynamic simulation-based content are combined:
    - static text modules are taken as is and inserted into the report.
    - dynamic content is re-created each time the qualification workflow is run and may change between OSP versions if there are differences between the previous and new models

Detailed documentation of the OSP Qualification Framework can be found under:
https://tinyurl.com/OSP-Qualification-Framework

OSP Qualification Framework: Qualification Report Structure

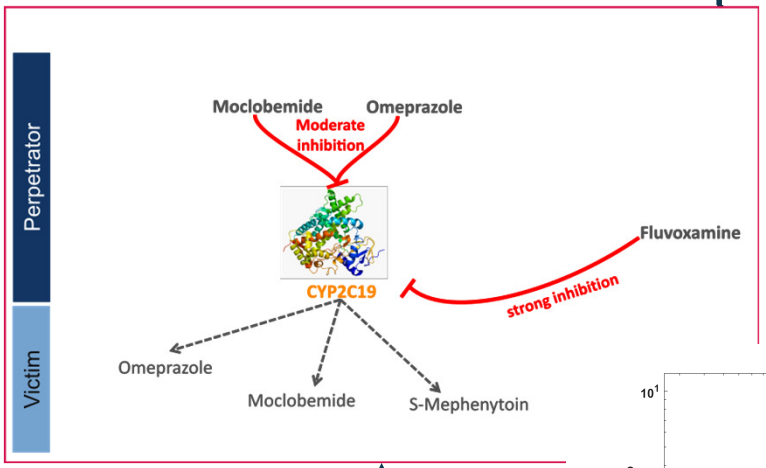
Version	1.2-
Qualification Plan Release	https://github.com/Open-Systems-Pharmacol
OSP Version	11.2
Qualification Framework Version	2.3

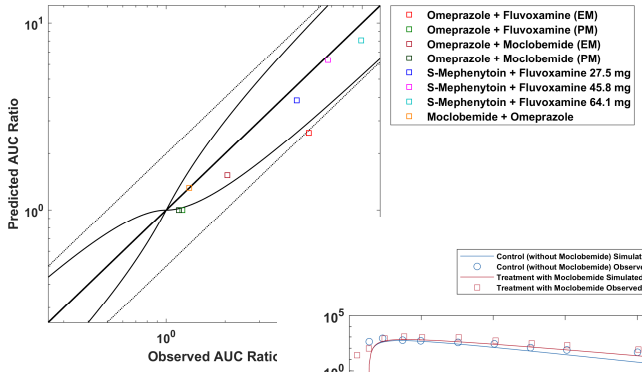
This qualification report is filed at:

https://github.com/Open-Systems-Pharmacology/OSP-Qualification-Reports

Table of Contents

- 1 Introduction
  - 1.1 Objective
  - 1.2 CYP2C19 DDI Network
    - Omeprazole - Moclobemide DDI
    - Omeprazole - Fluvoxamine DDI
    - S-Mephenytoin - Fluvoxamine DDI
    - Moclobemide - Omeprazole DDI
- 2 Qualification of Use Case CYP2C19-mediated DDI
  - Perpetrator
    - Fluvoxamine
    - Moclobemide
    - Omeprazole
  - Victim
    - Moclobemide
    - Omeprazole
    - S-Mephenytoin
- 3 Concentration-Time Profiles
  - 3.1 Omeprazole - Moclobemide DDI
  - 3.2 Omeprazole - Fluvoxamine DDI
  - 3.3 S-Mephenytoin - Fluvoxamine DDI
  - 3.4 Moclobemide - Omeprazole DD
- 4 Conclusion
- 5 References
- 6 Appendix
  - 6.1 Open Systems Pharmacology Suite (OSPS) Introduction
  - 6.2 Mathematical Implementation of Drug-Drug Interactions
  - 6.3 Automatic (re)-qualification workflow
- 7 Glossary





4 Conclusion

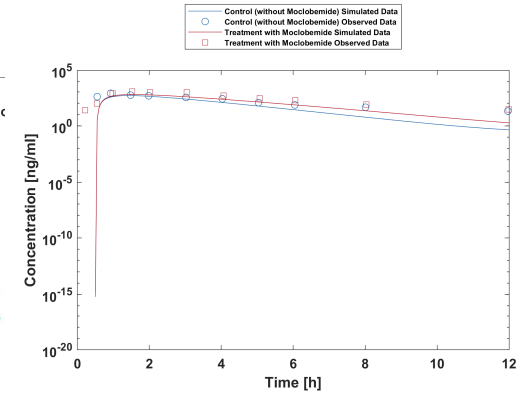
The predicted perpetrator/victim drug concentration-time profiles, DDI AUC and Cmax ratio well suited to characterize the CYP2C19 DDI network over the full range of reported DDI st moderate CYP2C19 inhibitors and substrates moclobemide (204 µM) and omeprazole (R: 5 For the strong inhibition of CYP2C19 by fluvoxamine, substrate dependent Ki values were u S-mephenytoin, respectively.

Fluvoxamine

- CYP2C19 inhibition:
  - DDI simulations with omeprazole as substrate demonstrate an excellent prediction (ratio pred/obs = around 1) of the inhibitory potential of fluvoxamine on CYP2C19 for both EM and PM for CYP2C19.
  - DDI simulations with s-Mephenytoin as substrate demonstrate an excellent prediction (ratio pred/obs = around 1 and within 2-fold) of the inhibitory potential of fluvoxamine on CYP2C19. However, DDI predictions tend to get slightly overpredicted for higher fluvoxamine doses.

Omeprazole

- Perpetrator:
  - DDI simulations with omeprazole as inhibitor of CYP2C19 demonstrated a good prediction of moclobemide levels (within 2-fold).
- Substrate:
  - DDI simulations with fluvoxamine as inhibitor of omeprazole demonstrated an excellent prediction of omeprazole levels on CYP2C19



Cho 2002 (Omeprazole 40 mg po) Asian EM

OSP Qualification Framework: Qualification Plan Structure

```
"Projects": [
    {
        "Id": "Midazolam",
        "Path": "../../Midazolam-Model.json"
    }
],
"ObservedDataSets": [],
"Sections": [
    {…},
    {
        "Reference": "methods",
        "Title": "Methods",
        "Sections": [
            {
                "Reference": "modeling-strategy",
                "Title": "Modeling Strategy",
                "Content": "Content/Modeling_Strategy.md"
            }
        ]
    }
],
"Plots": {
    "AllPlots": [
        {
            "Project": "Midazolam",
            "Simulation": "Sim 1 - intravenous",
            "SectionReference": "ct-profiles"
        }
    ],
    "GOFMergedPlots": […],
    "ComparisonTimeProfilePlots": [],
    "DDIRatioPlots": [],
    "PKRatioPlots": []
},
"Inputs": [
    {
        "Project": "Midazolam",
        "Name": "Midazolam",
        "Type": "Compound",
        "SectionReference": "final-input-parameters"
    }
]
```

- **Projects**: project *snapshots* (see next slide) used in the qualification scenario.
- **ObservedDataSets**: *additional* pharmacokinetic (PK) observed data sets which are not directly included in the above projects.
- **Sections**: defines the chapter structure of a report. Each section can contain a reference to a **static** content file that is generated at the beginning of the chapter and the list of subchapters. All **dynamically** generated content (plots, tables, parameterization of the building blocks/simulations etc.) must have a chapter reference within the qualification plan.
- **Plots**: type of plots (and some additional related information like tables and qualification measures) to be generated for the report.
- **Inputs**: specifies which building blocks and/or simulations should be described in the report.

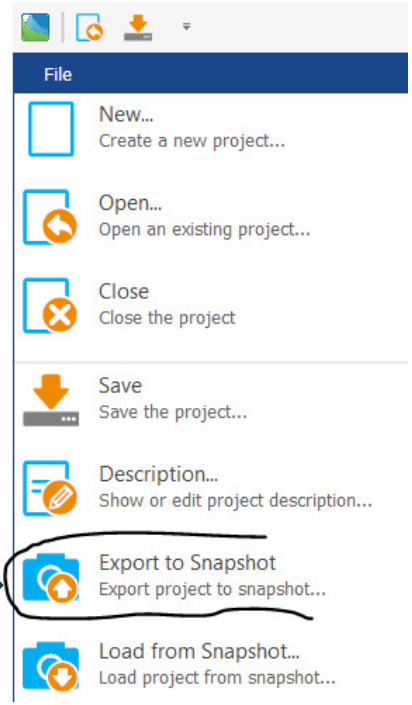
Detailed documentation of the qualification plan structure: https://tinyurl.com/OSP-Qualification-Plan

Qualification Framework: (PK-Sim) Project Snapshots

A *project snapshot* contains the **minimal amount of information** needed to recreate the project from scratch. This includes:

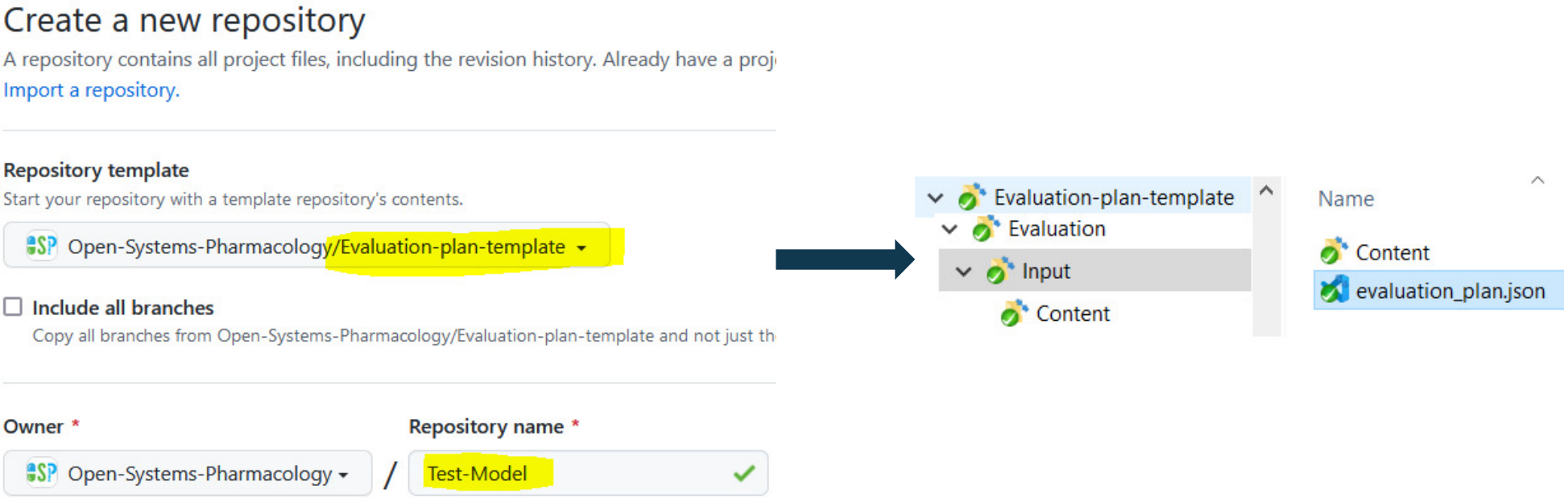
- the information on primary substance specific input parameters (e.g. molecular properties like *molecular weight*, *lipophilicity*, etc.) and required inputs (e.g. demographic characteristics) to define the system parameters. In addition, any changes made in the existing model, such as a change in liver volume, that is not a default value, are stored in the snapshot and included in the new model once recreated from the snapshot
  (s. https://tinyurl.com/PK-Sim-Snapshot for more details)
- Observed data sets used in the project
- Full configuration of plots (selected outputs, plot settings etc.)
- Parameter identifications used in the project

Projects snapshots can be created directly from PK-Sim:



OSP Qualification Framework: Edit Qualification Plan

- Dedicated tool under development (to be released in 2025)
- Currently the best way to edit qualification plans is to use *Visual Studio Code* (VSCode) (free tool from Microsoft, https://code.visualstudio.com/Download).
- OSP provides some extensions for VSCode, that allow to extend/modify qualification plans in a fast and convenient way (described in detail here: https://tinyurl.com/edit-qualification-plan).
- When creating a new model repository on GitHub: use the **model template** repository. This will create a default qualification plan for the model (*model evaluation plan*), which then needs to be customized.



OSP Qualification Framework: Execute Qualification Plan

- Qualification plans are executed in R via the OSP Reporting Engine (RE) (R package **ospsuite.reportingengine**)
  - Install RE following the instructions under https://tinyurl.com/OSP-RE-Install
  - Install OSP Qualification Runner from https://tinyurl.com/OSP-Qualification-Runner
  - OSP Suite must be installed as well
- To execute a qualification plan and create a report in Markdown format (plus optionally in MS-Word format)
  - Download the template for the `createQualificationReport` function from https://tinyurl.com/OSP-Qualification-Template and save it e.g. as *Workflow.R*
  - Adjust Workflow.R if required (s. comments in the template)
  - Execute `createQualificationReport` the function

```
> library(ospsuite.reportingengine)
> createQualificationReport(qualificationRunnerFolder = "C:/QualificationRunner_11.1.130/",
  maxSimulationsPerCore = 4, versionInfo = QualificationVersionInfo$new(qualificationPlanRelease =
  "1.0", osp = "11.3", qualificationFramework = "3.2"))
```

OSP Release Planning and Roadmap

OSP user roles and responsibilities

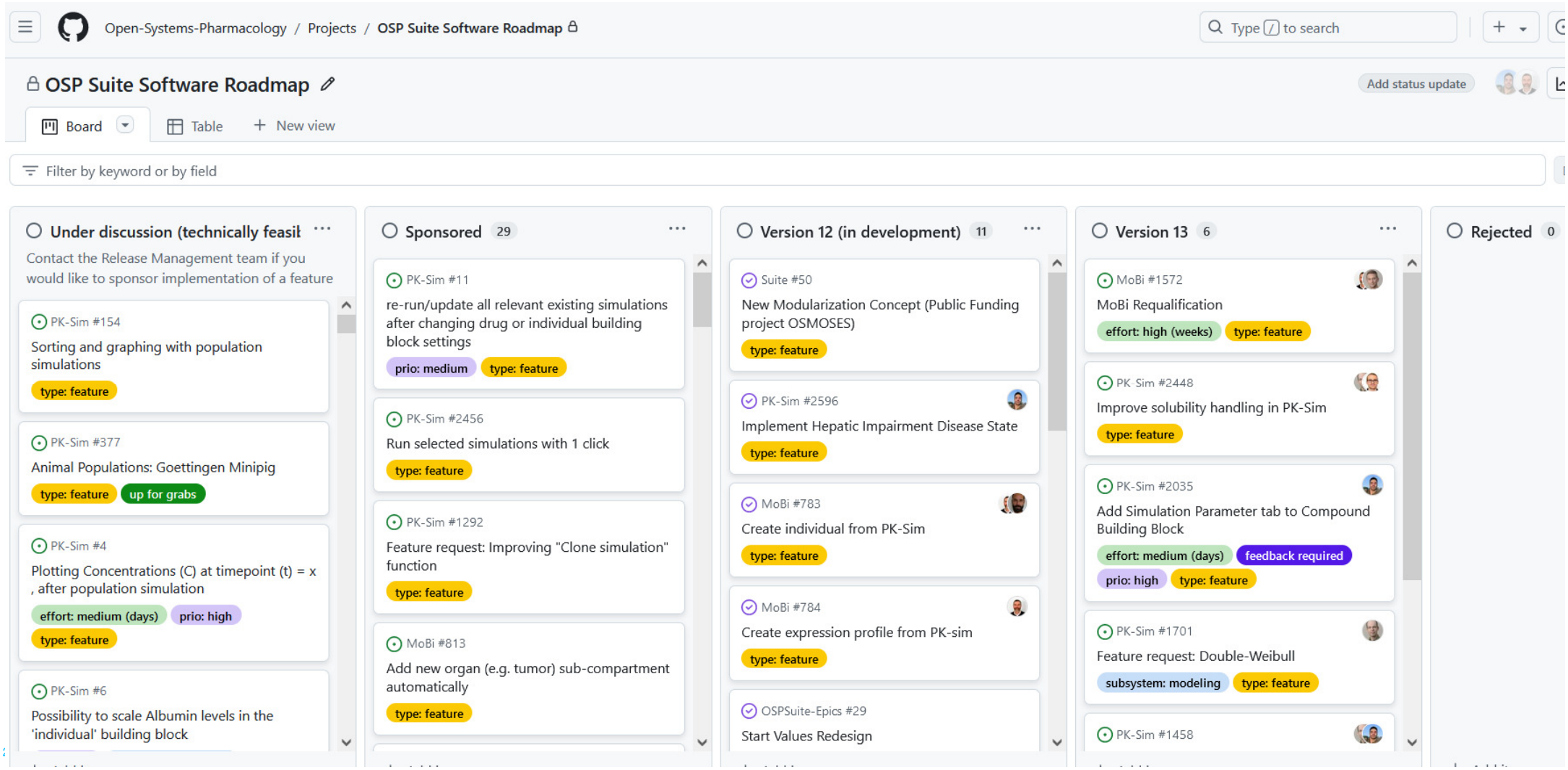
- Management Team (**MT**), Sounding Board (**SB**), OSP Community Members (**OSP Members**) , Users

- Focus Groups (**FG**)
  - Conceptualize and coordinate activities of the respective field
  - For the full list of focus groups, along with their assignments and leaders, please visit https://github.com/Open-Systems-Pharmacology/Roadmap
  - **FG „Suite Release Management and Software Usability“** manages
    - The release process of the OSP Suite and further OSP software components
    - Design, development, maintenance, and testing of software architecture / Infrastructure /User Interface
    - Operation of software integration platform
    - Implementation and integration of Application workflows and methods
    - Members
      - Michael Sevestre (Design2Code; GitHub-UserId: **msevestre**)
      - Pavel Balazki (ESQlabs; GiitHub-UserId: **PavelBal**)
      - Robert McIntosh (ESQlabs; GitHub-UserId: **rwmcintosh**)
      - Juri Solodenko (Lead; Bayer; GitHub-UserId: **Yuri05**)

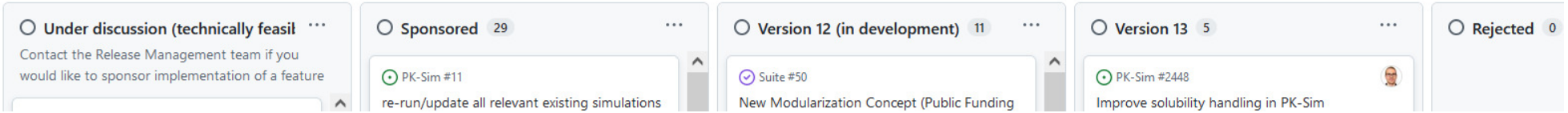
- Core Developers (**DEV**)
  - Development and testing of OSP software platform and tools
  - Review of proposed software changes and their integration into the platform

Release Planning and Roadmap

OSP Suite Software Roadmap: https://tinyurl.com/OSP-Software-Roadmap



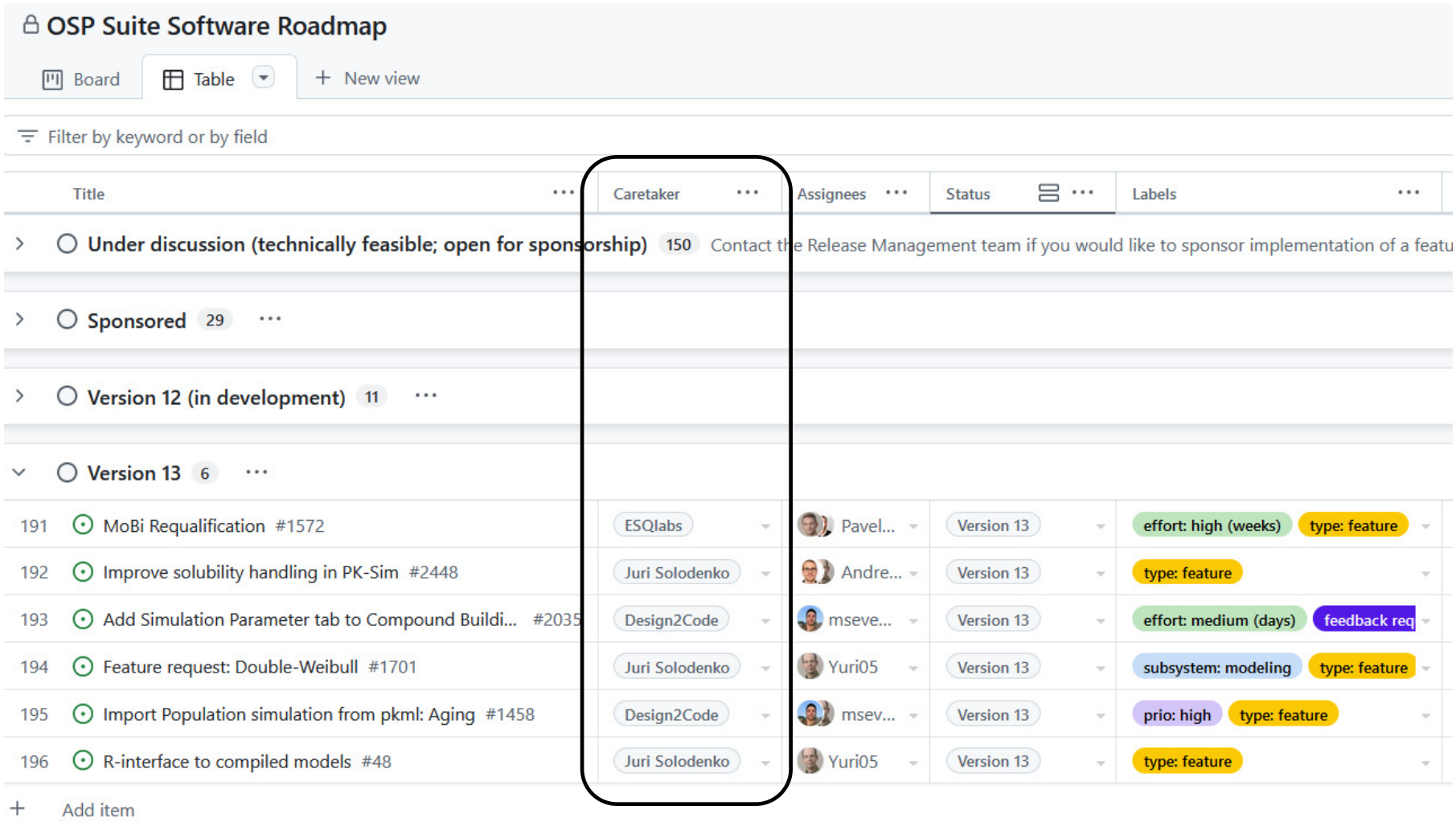
Release Planning and Roadmap



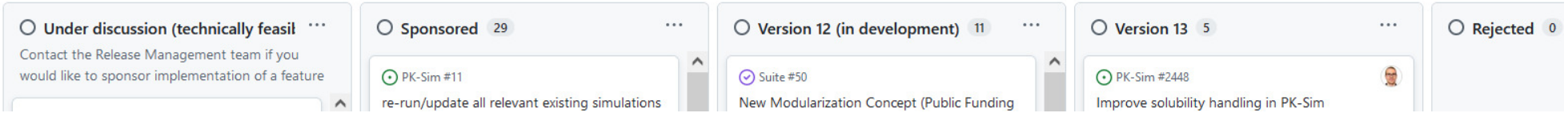
1. If you have a feature wish: enter it either in the OSP forum or as an issue in one of the repositories
   - http://forum.open-systems-pharmacology.org
   - https://github.com/Open-Systems-Pharmacology/PK-Sim/issues
   - https://github.com/Open-Systems-Pharmacology/MoBi/issues
   - …
2. New feature wishes are periodically screened by the Release Management Focus Group (**FG-RM**)
   - Features which are absolutely not feasible/impossible to integrate get the status "**Rejected**" on the roadmap (extremely rare)
   - All other features get the status "**Under discussion (technically feasible; open for sponsorship)**"
3. Each feature to be implemented needs a (financial) sponsor. If you want to sponsor a feature: inform the FG-RM. The feature will then get the status "**Sponsored**" (and you will be assigned as the "Caretaker" of this feature).

Release Planning and Roadmap

Each feature to be implemented needs a (financial) sponsor. If you want to sponsor a feature: inform the FG-RM. The feature will then get the status “**Sponsored**” (and you will be assigned as the “Caretaker” of this feature).



Release Planning and Roadmap

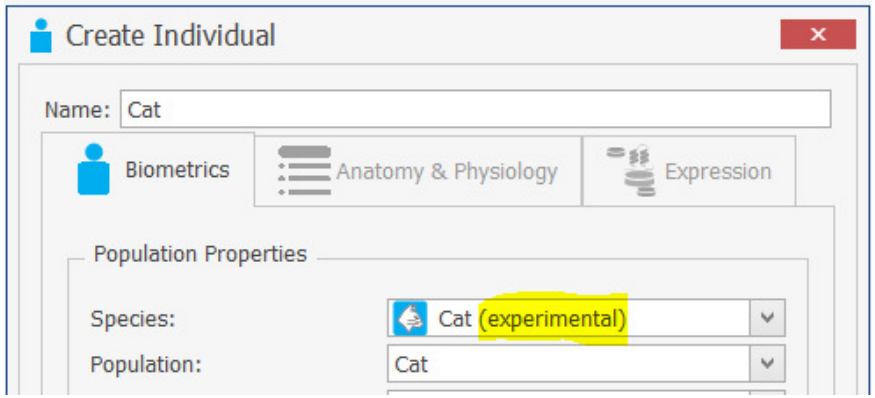


4. Find resources to implement the **scientific** part of the feature (if applicable), e.g.
   - Collect required data (anatomical/physiological parameters) for integration of a new species or population.
   - Create a prototype in MoBi (starting with OSP version 12: as a new **module(s)**) for adjustments/extensions of a PBPK model.
5. Develop a **validation/verification** (not a *qualification*) strategy. E.g.: paper(s) where a model extension was described must be reproduced with acceptable error tolerance.
6. Find resources to implement the **technical** (software) part of the feature and inform the FG-RM.
7. FG-RM will decide if it's feasible to integrate the feature in the current release or in the next release. The feature will then get the status "Version N" or "Version N+1" (where N is the currently developed version)

Release Planning and Roadmap

8. New feature might affect existing models! If the requalification of existing models is not satisfying:
   - Some of the models must be recalibrated (e.g. refit previously fitted model parameters).
   - And/Or the implementation of the new feature must be revised.

9. When the newly developed feature is integrated into the software, it might require development of new qualification scenarios (e.g. when integrating a new animal species or new human population).
   - Until the qualification is performed: new feature will be marked as “experimental” in the software.



10. Once the qualification of the new feature is performed: the “experimental” flag will be removed from the software.