Drug Disease Model Resources

ddmore

DDMoRe Workshop

Lutz Harnisch, Benjamin Ribba, Maciej Swat,

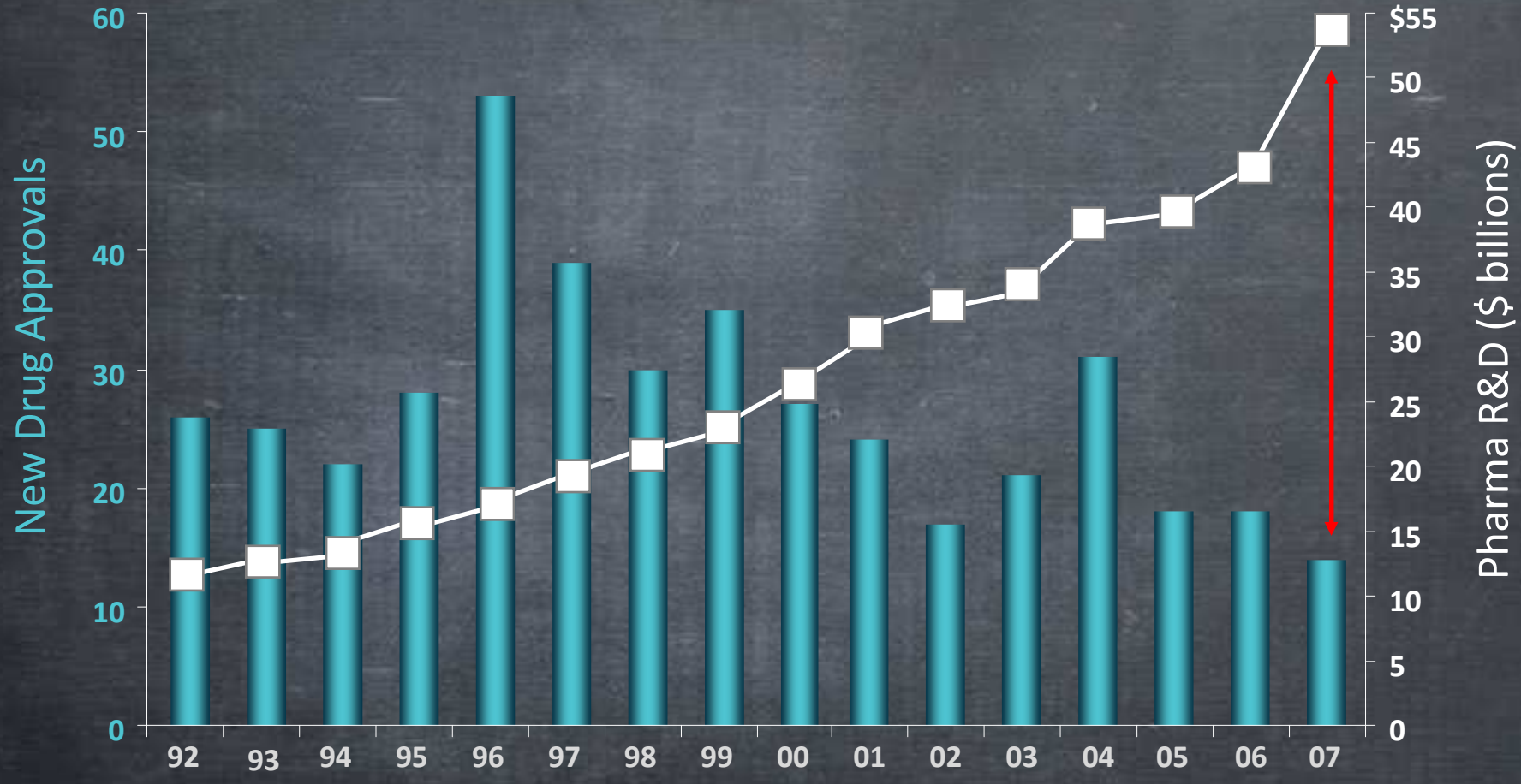
Richard Kaye, Natallia Kokash & Nick Holford

On behalf of the DDMoRe consortium

imi efpia


Therapeutic Innovation

The Productivity Gap in Pharma R&D



Source: Burrill & Company; US Food and Drug Administration.

Therapeutic innovation is challenging

Drug development decision makers **rely** on the best rational to select drug candidates, targets, trial designs, dose regimens, patient populations and suitable endpoint measures

Therapeutic innovation is challenging

A lack of **informed decision making** can have detrimental effects to patients, choice of the wrong development program, and potentially induce huge financial losses

Therapeutic innovation is challenging

Society pushes for more **transparency** and clinical trial **data sharing**, opening the door for aggressive competition

Integrative tools

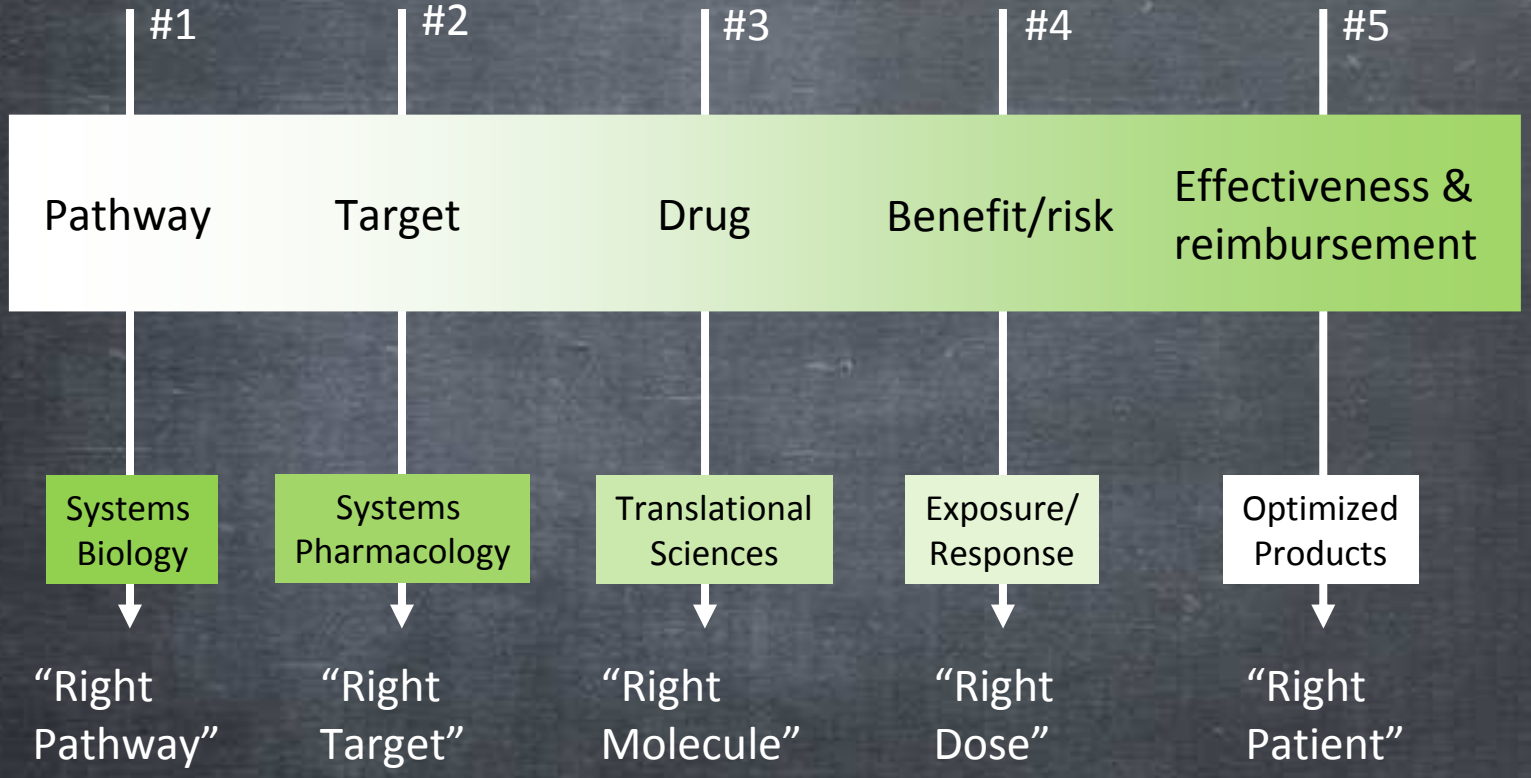
Modelling and Simulation (M&S)

as one of the core technology of pharmaceutical industry, is essential in performing such knowledge integration, providing through inference the quantitative basis for better informed decision making

Model-based drug development

Integrative tools

Model-based drug development



Model-Based Drug Development: A Rational Approach to Efficiently Accelerate Drug Development
CPT 2013, Milligan PA, et al

The Problem

The Problem

How do you find disease or drug models that might be useful for your analysis?

- How can you be sure that published models are relevant?
- How can you be sure that published models work (are reproducible)?

If code is available in software X, but you use software Y, how do you translate the model and ensure that the result is equivalent?

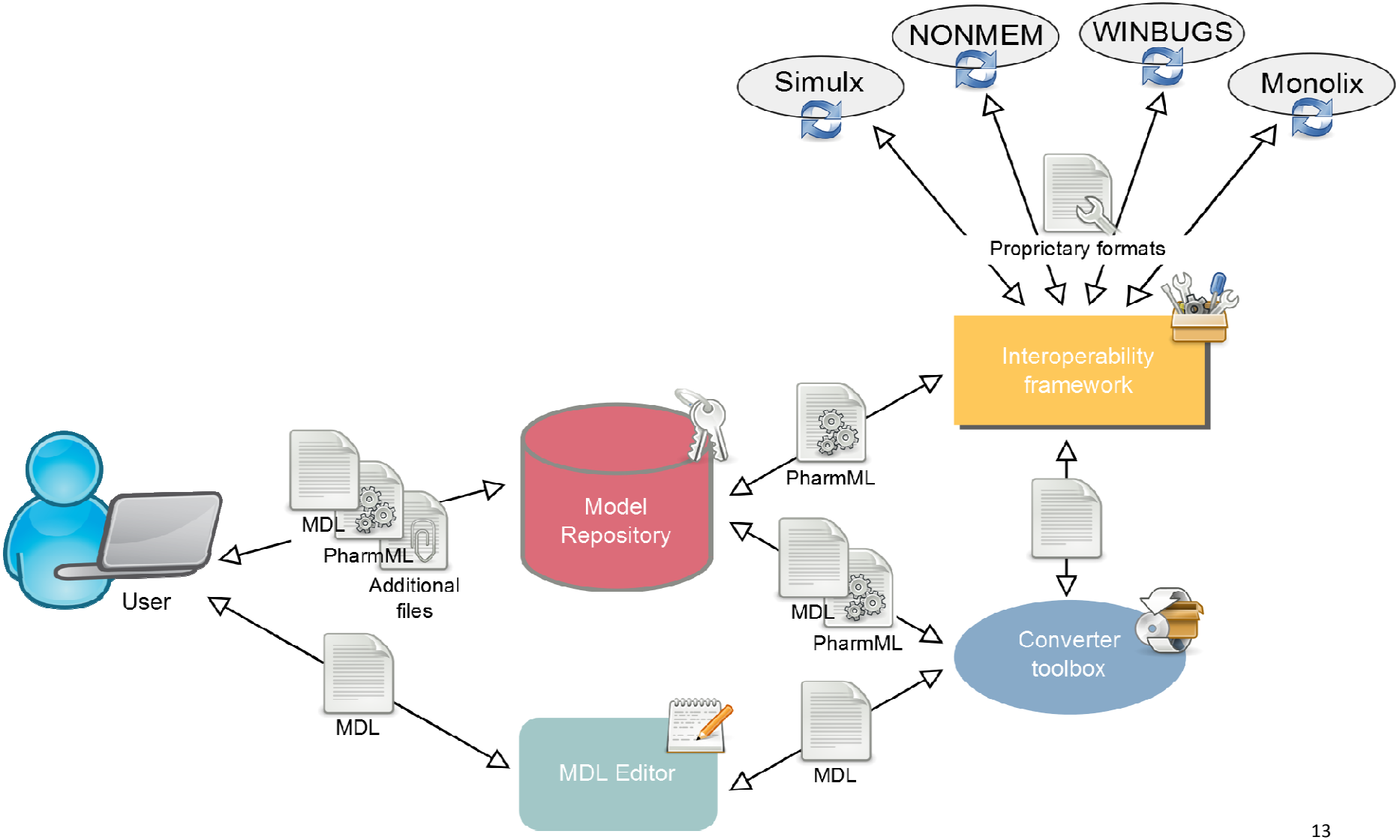
How much of your time is lost by working in a non-integrated environment?

March 2011 – February 2016

The Solution

- *Benjamin Ribba:* Clinical problem, Tumour Growth Inhibition model (**TGI model**)
- *Maciej Swat:* How to store published models and associated details (**Repository**)
- *Nick Holford :* How to create, review, extend and combine published models using a common Modelling Description Language (**MDL**)
- *Natallia Kokash*: Construct models, check syntax, perform tasks within an Integrated Development Environment (**MDL-IDE**)
- *Maciej Swat:* Provide standards for model inputs and outputs using a software interchange standard language to allow translation between target modelling tools, Pharmacometrics Markup Language (**PharmML**)
- *Richard Kaye*: Demonstration on how to solve the clinical problem using a possible workflow via the DDMoRe framework (**Framework**)

DDMoRe Framework Overview



Real life example

This DDMoRe satellite Workshop will demonstrate the model repository, languages and tools developed by the Consortium using as an example:

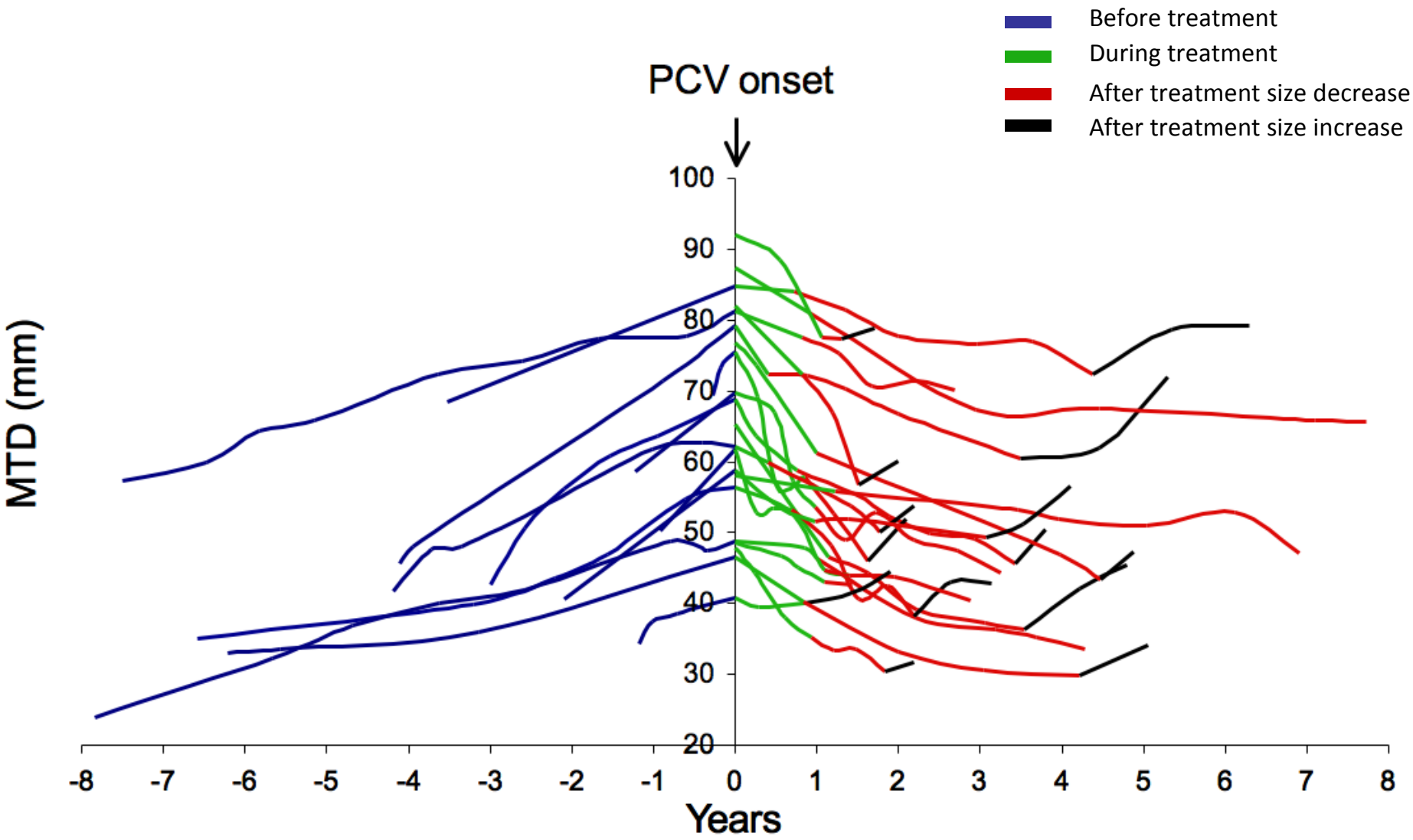
a tumour growth inhibition (TGI) model for the treatment of low-grade glioma using chemotherapy or radiotherapy

Low-grade glioma

- Low-grade gliomas (WHO grade II) are diffusely infiltrative brain tumors affecting young adults
- Annual number of pediatric and adult LGG cases diagnosed in the U.S. 1,800
- Tumor grows slowly: 1-2 mm/year, monitored with periodic MRI
- Patients can be asymptomatic and remained untreated for years
- Standard of care **PCV** (Procarbazine, CCNU, Vincristine)
  - chemotherapy regimen is given every 6 weeks for a maximal duration of 6 cycles
  - Prolonged response in the majority of patients

Data sample (21 patients)

long duration clinical study of patients with glioma



Modelling and Simulation

The use of existing clinical data to model tumors' dynamic response to antitumor treatments is a promising approach toward improving treatment efficacy and accelerating the development of antitumor drugs.

Clinical questions

- Can a model be formulated to propose explanations of the prolonged-response phenomenon?

- Can the PCV scheduling be changed to improve the treatment efficacy?
  - Aim is a further prolonged treatment response

Worked Example

- The study goal is to understand both the biology of tumour growth and how drug treatments modify growth as reflected in tumour size. Description of the data requires a mixed effect modelling approach.
  - The original report of the model was developed using Monolix.

The Story

- A mixed-effect model has been developed to characterize tumor size dynamics in glioma patients treated with PCV chemotherapy
- Herein, we illustrate the potential of the DDMoRe framework to:
  - Browse the DDMoRe Model Repository
  - Estimate parameters of the brain tumor mixed-effect model
  - Simulate (explore) the model to propose an explanation of the prolonged response phenomenon
  - Identify a modified PCV regimen scheduling

Publication

Cancer Therapy: Clinical

Clinical Cancer Research

A Tumor Growth Inhibition Model for Low-Grade Glioma Treated with Chemotherapy or Radiotherapy

Benjamin Ribba[1], Gentian Kaloshi[6], Mathieu Peyre[2], Damien Ricard[7], Vincent Calvez[1], Michel Tod[3,4], Branka Cajavec-Bernard[1], Ahmed Idbaih[6], Dimitri Psimaras[6], Linda Dainese[8], Johan Pallud[9], Stéphanie Cartalat-Carel[2], Jean-Yves Delattre[6], Jérôme Honnorat[2,4,5], Emmanuel Grenier[1], and François Ducray[2,4,5]

Abstract

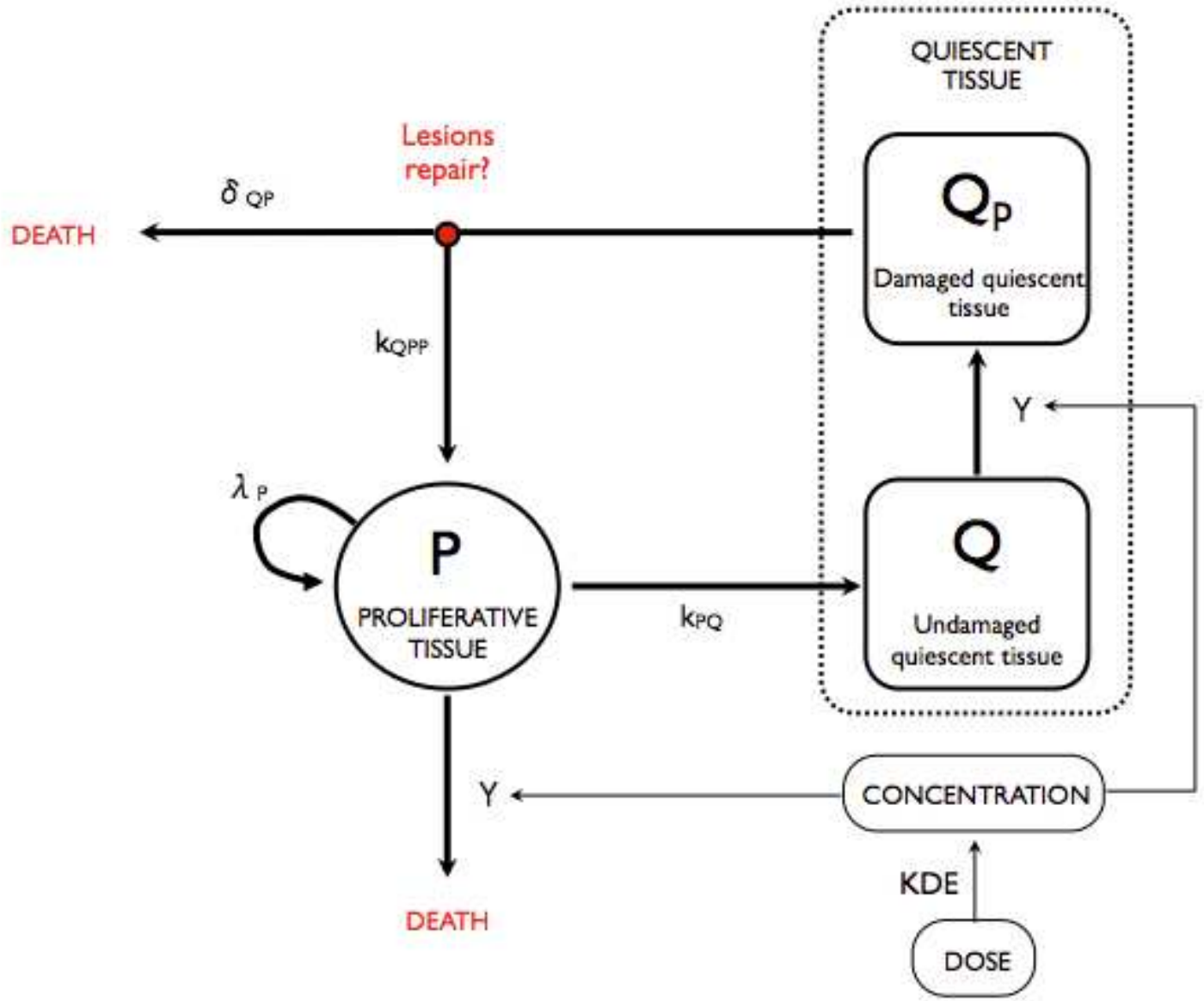
Purpose: To develop a tumor growth inhibition model for adult diffuse low-grade gliomas (LGG) able to describe tumor size evolution in patients treated with chemotherapy or radiotherapy.

Experimental Design: Using longitudinal mean tumor diameter (MTD) data from 21 patients treated with first-line procarbazine, 1-(2-chloroethyl)-3-cyclohexyl-l-nitrosourea, and vincristine (PCV) chemotherapy, we formulated a model consisting of a system of differential equations, incorporating tumor-specific and treatment-related parameters that reflect the response of proliferative and quiescent tumor tissue to treatment. The model was then applied to the analysis of longitudinal tumor size data in 24 patients treated with first-line temozolomide (TMZ) chemotherapy and in 25 patients treated with first-line radiotherapy.

Results: The model successfully described the MTD dynamics of LGG before, during, and after PCV chemotherapy. Using the same model structure, we were also able to successfully describe the MTD dynamics in LGG patients treated with TMZ chemotherapy or radiotherapy. Tumor-specific parameters were found to be consistent across the three treatment modalities. The model is robust to sensitivity analysis, and preliminary results suggest that it can predict treatment response on the basis of pretreatment tumor size data.

Conclusions: Using MTD data, we propose a tumor growth inhibition model able to describe LGG tumor size evolution in patients treated with chemotherapy or radiotherapy. In the future, this model might be used to predict treatment efficacy in LGG patients and could constitute a rational tool to conceive more effective chemotherapy schedules. *Clin Cancer Res; 1–10. ©2012 AACR.*

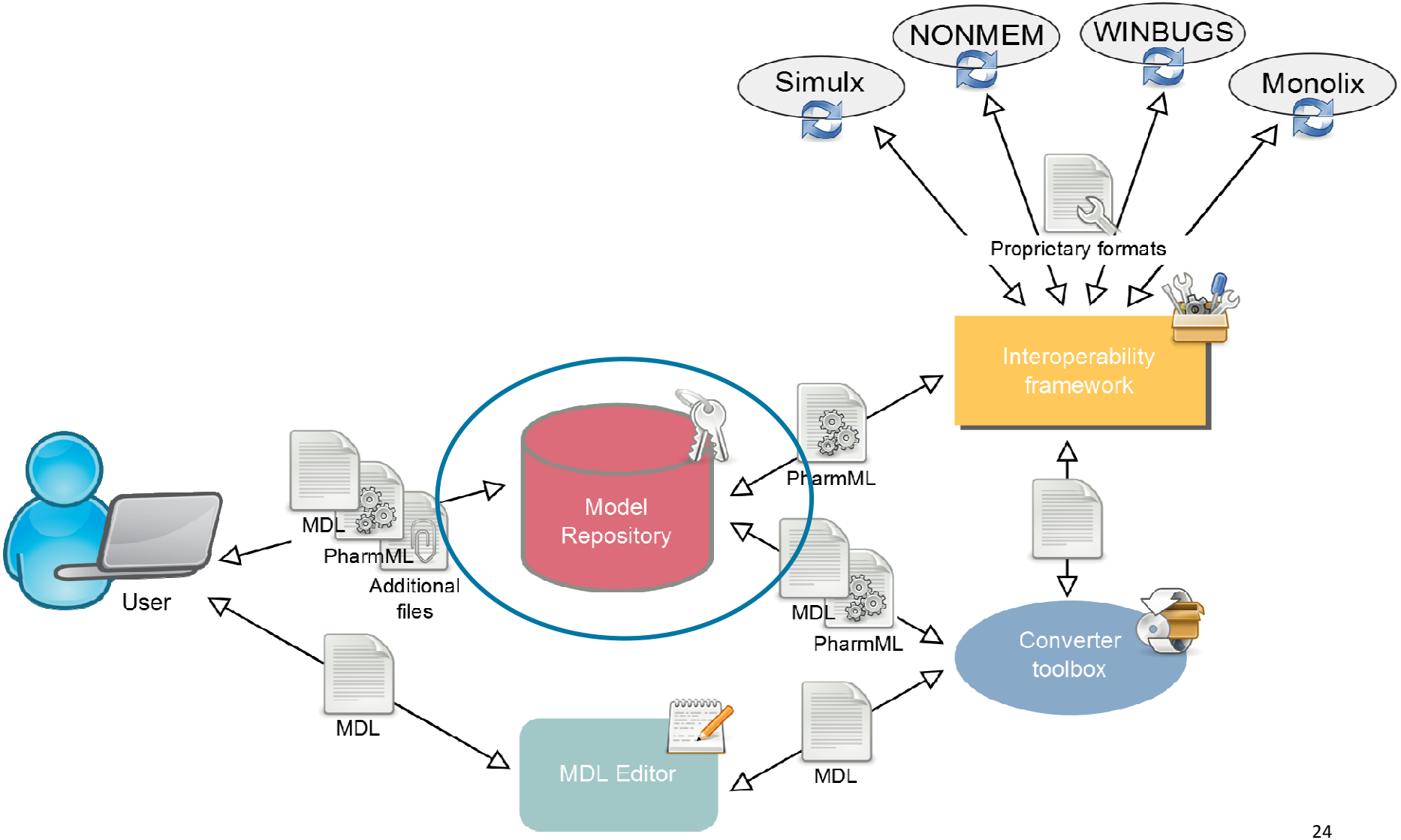
Schematic view



Parameters

- System of 4 compartments written as ordinary differential equations
- 6 parameters and 2 initial conditions (P0 and Q0)
  - λP (growth rate) and kPQ (quiescence rate) only regulates tumor growth in the absence of treatment
- The model was developed within a mixed-effect (population) context
  - 8 fixed parameters and 7 inter-individual variability parameters
- SAEM (with Monolix and NONMEM) was used to estimate parameters

DDMoRe Framework Overview



DDMoRe Model Repository

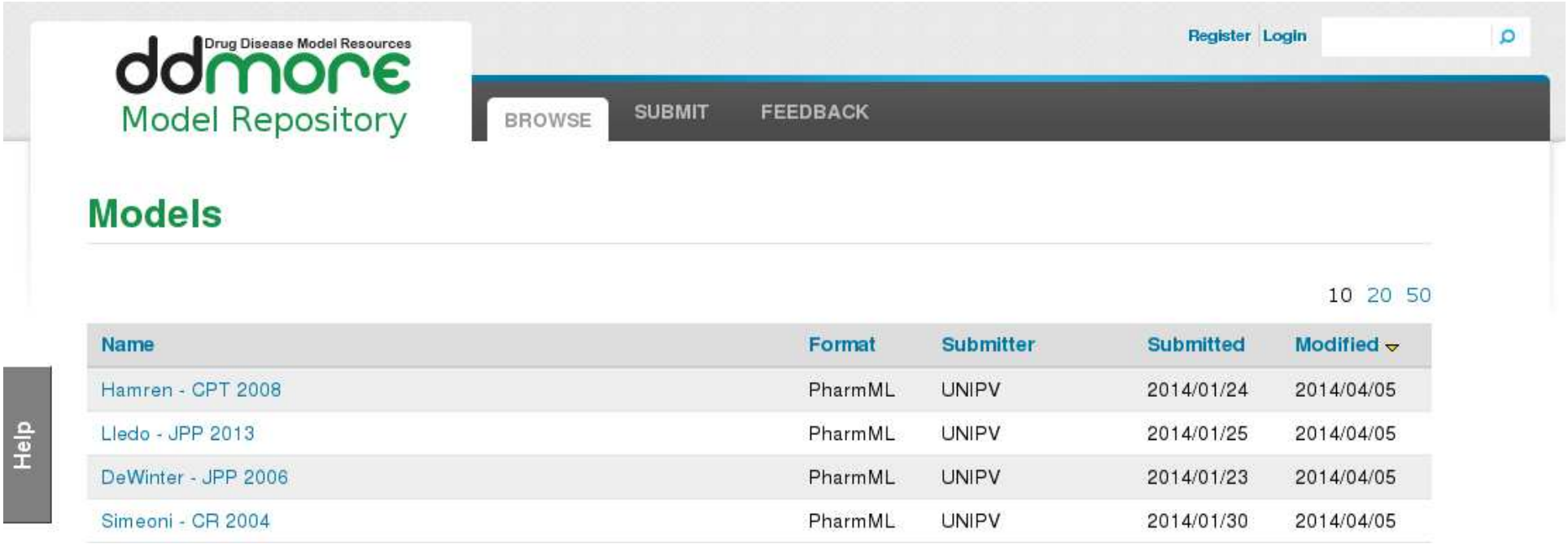
- Provides a collaborative model development platform
- Offers secured and versioned storage of models, data and metadata
- Facilitates dissemination and reuse of models
- Makes content available to **users** and **tools**
- Advocates the use of standard formats

Content – Library of disease models

- Selection of publicly-available models from key therapeutic areas
  - Diabetes
  - Oncology
  - Other diseases e.g. neurodegenerative, infectious

- Available models and data are encoded using standard formats developed within DDMoRe

Browsing content



- Browsing using different classifications
  - Therapeutic areas – e.g. oncology
  - Substance of interest – e.g. drug
- Sorting based on different orders – alphabetical, therapeutical

Building an effective search

- Information about the model is expressed in a knowledge representation format which can be understood by computers
  - e.g. the therapeutic area of this model is oncology
- The semantic context of model entities is expressed using unambiguous external references
  - glioma is a type of cancer

Model overview

Brief, but descriptive title

Ribba - CCR 2012 Tumour Growth Inhibition

Download

Overview | Files | History | Model Definition | Trial Design | Estimation Steps

Model Description:

A Tumor Growth Inhibition Model for Low-Grade Glioma Treated with Chemotherapy or Radiotherapy.

Textual description of the model

Format: PharmML (0.2.1)

Publication: **A tumor growth inhibition model for low-grade glioma treated with chemotherapy or radiotherapy.**
F Ducray, L Dainese, J Honnorat, B Cajavec-Bernard, M Peyre, V Calvez, S Cartalat-Carel, E Grenier, JY Delattre, J Pallud, M Tod, A Idbaih, D Psimaras, B Ribba, G Kaloshi, D Ricard
Clinical cancer research : an official journal of the American Association for Cancer Research, 9/2012, Volume 18, Issue 18, pages: 5071-5080

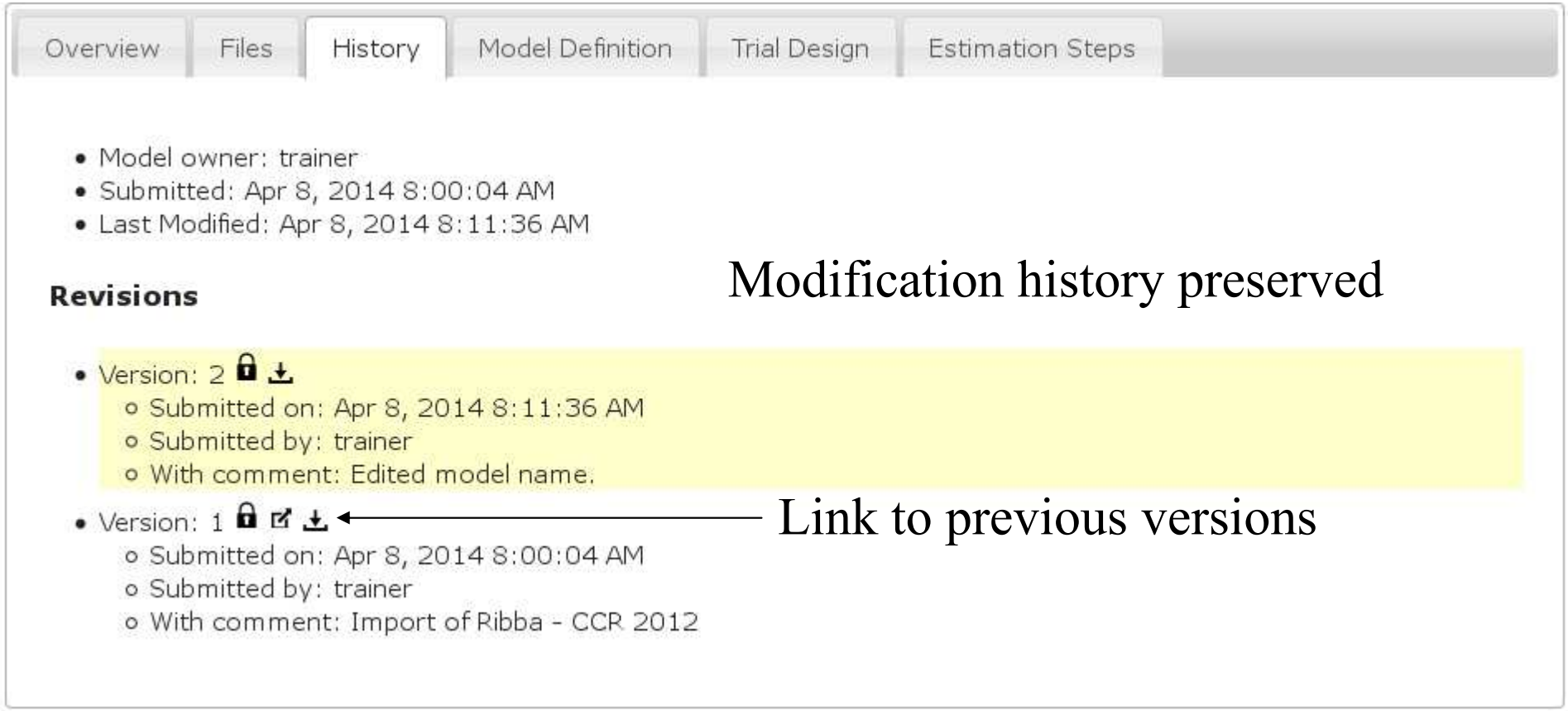
Publication details

Contributors: trainer

Submitter

Model display – files

Model display – history



Display of models in standard formats

Overview | Files | History | Model Definition | Trial Design | Estimation Steps

Independent variable time

Function Definitions

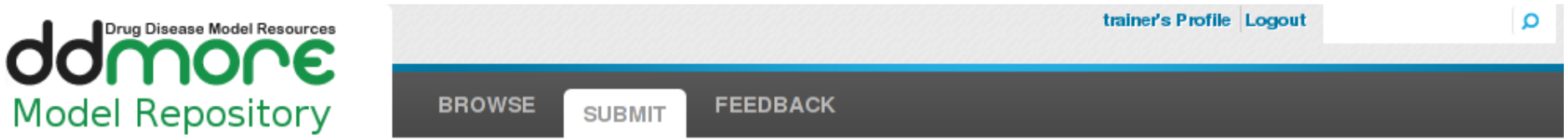
$$\text{constantErrorModel}(a) = a$$

Structural Model *sm1*

Variable definitions

$$\frac{\text{dC}}{\text{dtime}} = \left(-\text{KDE} \times C\right)$$
$$\frac{\text{dPT}}{\text{dtime}} = \left(\left(\text{LAMBDAP} \times \left(\text{PT} \times \left(1 - \frac{\text{PSTAR}}{K}\right)\right)\right) + \left(\left(\text{KQPP} \times \text{QP}\right) - \left(\left(\text{KPQ} \times \text{PT}\right) - \left(\text{GAMMA} \times \left(C \times \left(\text{KDE} \times\right.\right.\right.\right.\right.$$
$$\frac{\text{dQ}}{\text{dtime}} = \left(\left(\text{KPQ} \times \text{PT}\right) - \left(\text{GAMMA} \times \left(C \times \left(\text{KDE} \times Q\right)\right)\right)\right)$$
$$\frac{\text{dQP}}{\text{dtime}} = \left(\left(\left(\text{GAMMA} \times \left(C \times \left(\text{KDE} \times Q\right)\right)\right) - \left(\text{KQPP} \times \text{QP}\right)\right) - \left(\text{DELTAQP} \times \text{QP}\right)\right)$$
$$\text{PSTAR} = (\text{PT} + (Q + \text{QP}))$$

Model submission



Submit a model

Submission Guidelines

You are about to submit a new model to the DDMoRe Model Repository. We accept submissions in PharmML version 0.2.1 and MDL 5.0.8.

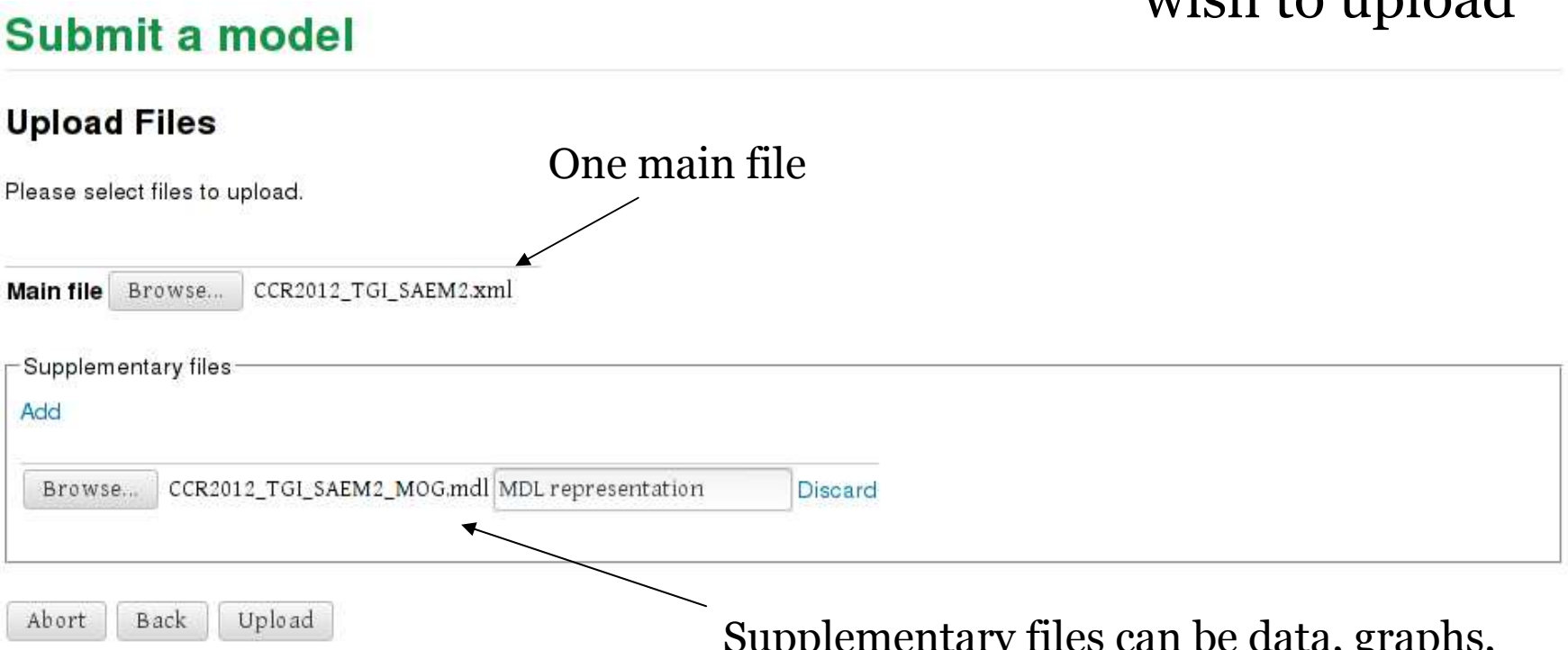
In your submission, you may also choose to provide any additional files considered relevant to the model, which may be labelled with a description.

Abort Continue

Learn about the supported formats

Model submission

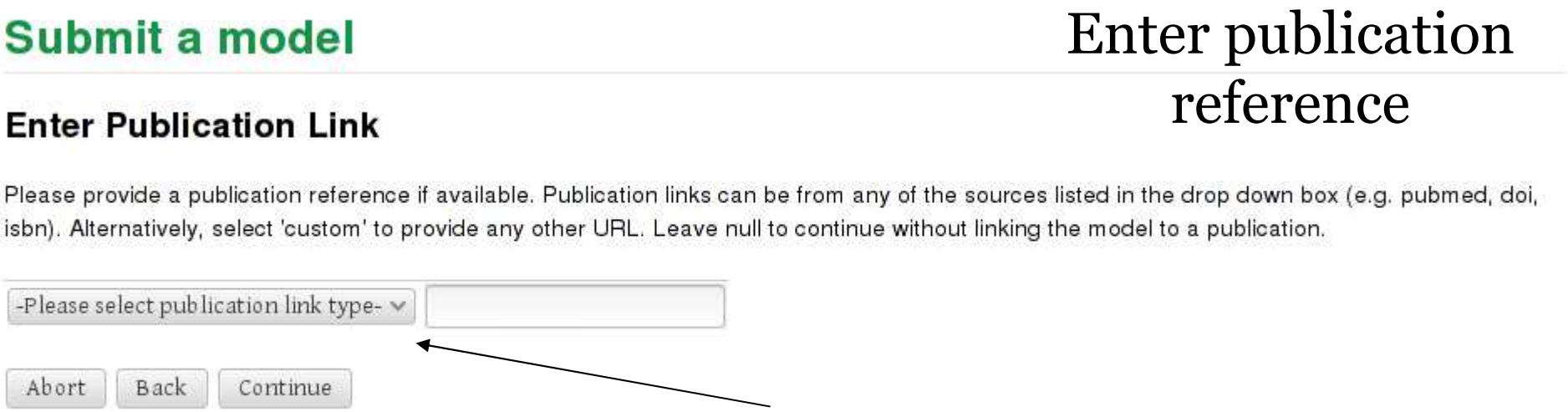
Specify the files you wish to upload



One main file

Supplementary files can be data, graphs, processing scripts, or any information related to this model submission

Model submission

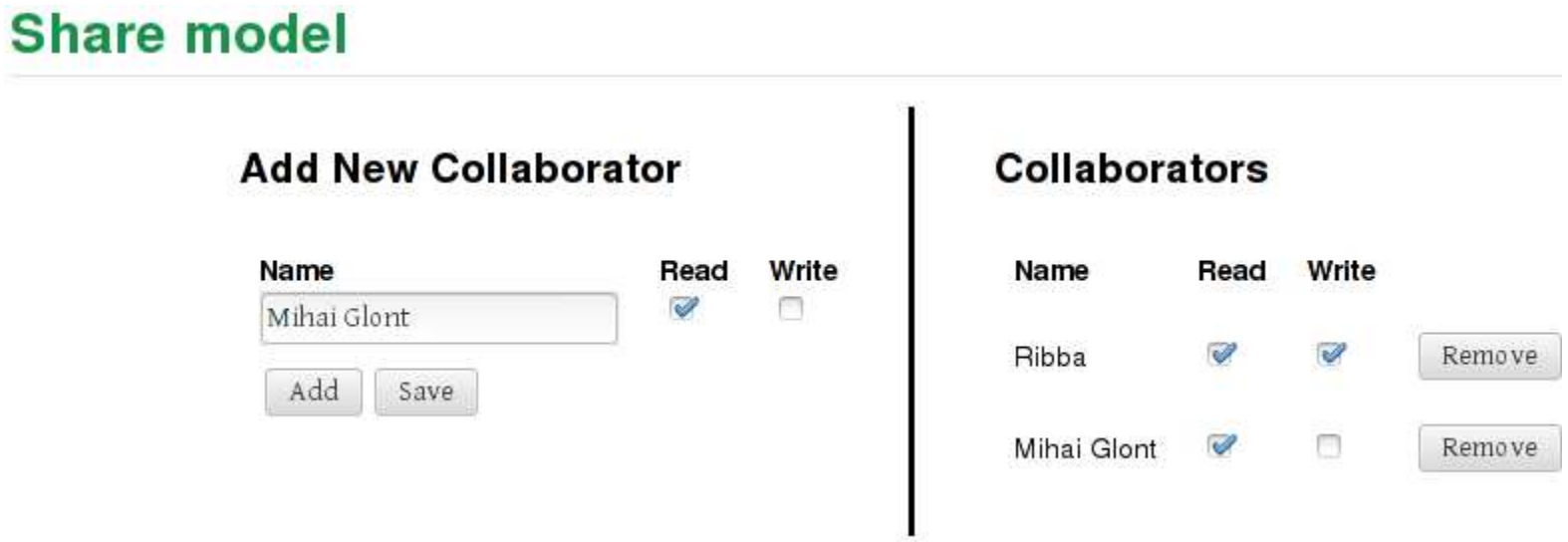


Enter publication reference

Information automatically extracted if PubMed ID is supplied

Sharing

- Models can be shared with other modellers
- Differentiate between
  - collaborators that can update the model, and
  - collaborators that can only see the model



Showing your model to others

- Useful for feedback and review purposes

The model can be viewed and retrieved, but not updated

Ribba - CCR 2012 Tumour Growth Inhibition

Overview | Files | History | Model Definition | Trial Design | Estimation Steps

Model Description:

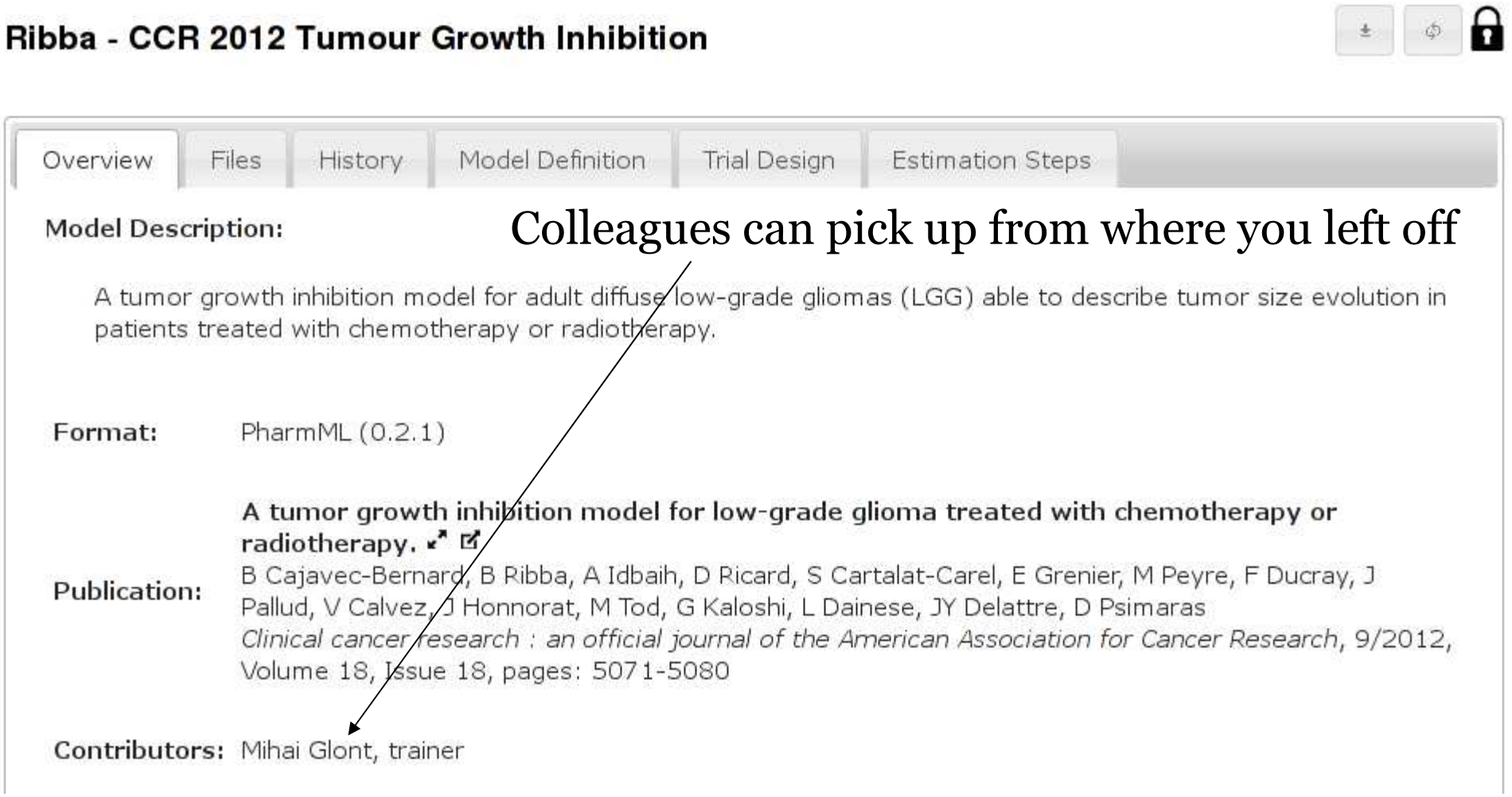
A Tumor Growth Inhibition Model for Low-Grade Glioma Treated with Chemotherapy or Radiotherapy.

Format: PharmML (0.2.1)

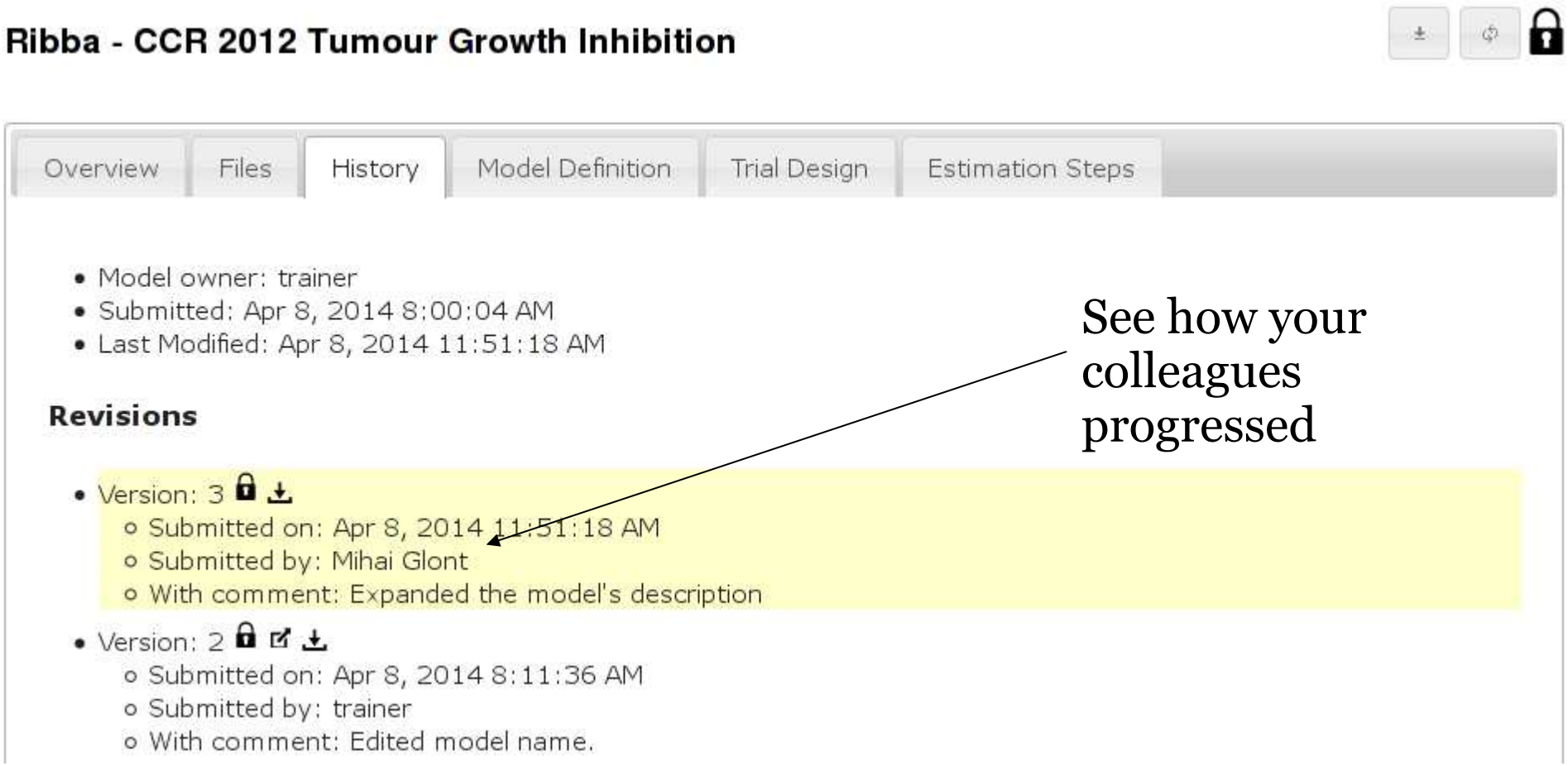
Publication: **A tumor growth inhibition model for low-grade glioma treated with chemotherapy or radiotherapy.**
F Ducray, JY Delattre, V Calvez, D Ricard, L Dainese, B Cajavec-Bernard, J Honnorat, E Grenier, A Idbaih, D Psimaras, G Kaloshi, J Pallud, B Ribba, M Tod, M Peyre, S Cartalat-Carel
Clinical cancer research : an official journal of the American Association for Cancer Research, 9/2012, Volume 18, Issue 18, pages: 5071-5080

Contributors: trainer

Updating models collaboratively



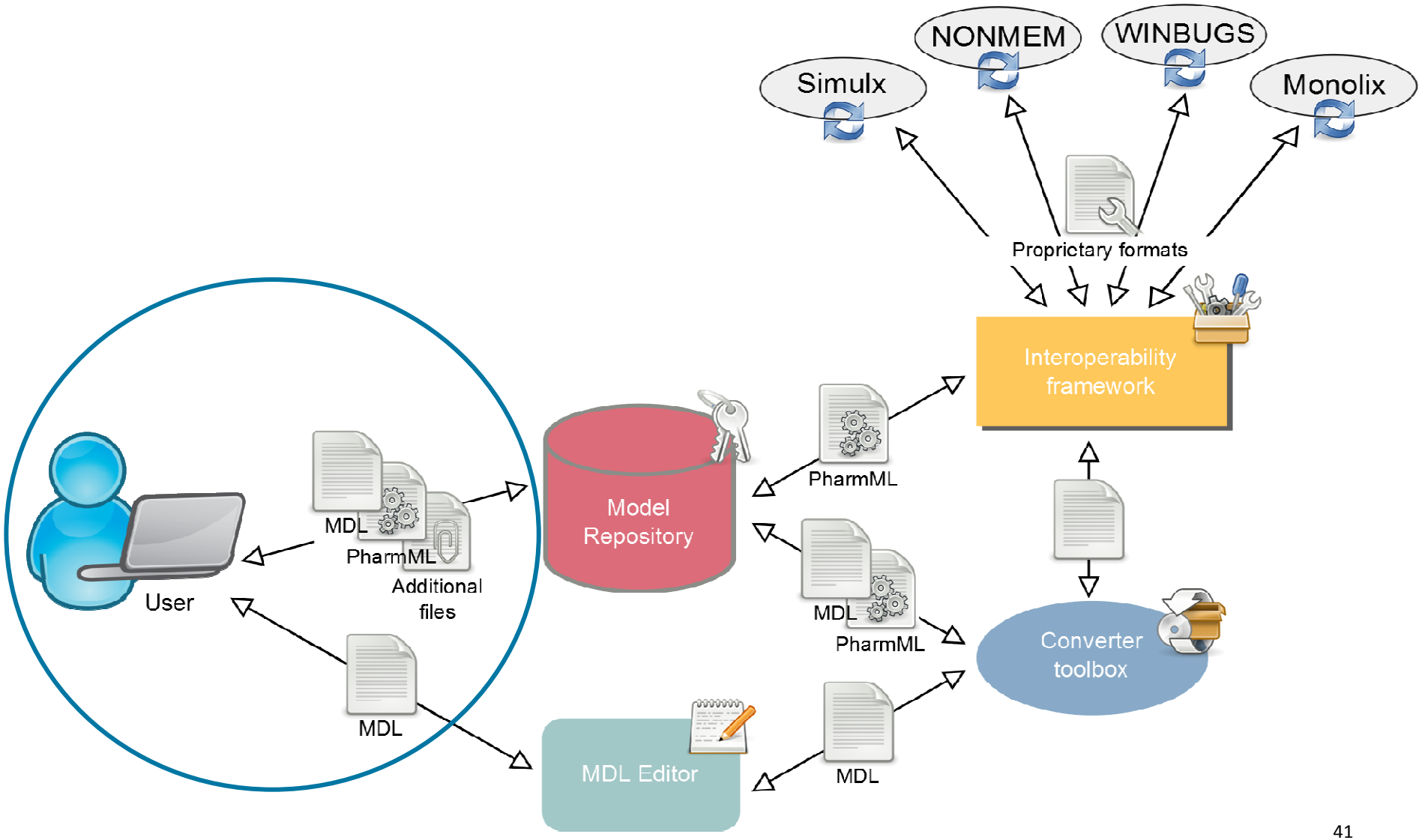
Updating models collaboratively



Model Repository

- The DDMoRe Repository is a public resource for disseminating models encoded in standard formats
- Models submitted to the DDMoRe Repository can be developed collaboratively
- Models from the DDMoRe Repository can be executed using other tools developed in the Consortium
- Models from the DDMoRe Repository can be edited using MDL IDE

DDMoRe Framework Overview



Target Languages

Tumour Growth Inhibition (TGI)

Solution or Problem?

NONMEM

```
$PROB CCR2012 RIBBA TUMOUR GROWTH INHIBITION
$INPUT ID TIME AMT DV MDV EVID
$DATA dataLGG_PCV_nh2.csv
$EST METHOD=SAEM NBURN=3000 NITER=1000 ISAMPLE=1 NSIG=2 SIGL=6
   CTYPE=3 CITER=10 CALPHA=0.05 NOPRIOR=1
   ;LAPLACE NUMERICAL SLOW
   NOABORT PRINT=10
   GRD=TS(9) FILE=CCR2012_TGI_SAEM2.raw
$COV MATRIX=R PRINT=E UNCONDITIONAL SIGL=8

$THETA
;Parameters for PCV chemotherapy study from CCR2012
7.13 ; POP_PT0 mm
41.2 ; POP_Q0 mm
0.121 ; POP_LAMBDAP mo-1
0.03 ; POP_KPQ mo-1
0.0031 ; POP_KQPP mo-1
0.008 ; POP_DELTAQP mo-1
0.729 ; POP_GAMA
0.24 ; POP_KDE mo-1
(0,1,) ; RUV_SD mm (No estimate reported in CCR2012)
$OMEGA
0.25 ; PPV_PT0
0.25 ; PPV_Q0
0.25 ; PPV_LAMBDAP
0.25 ; PPV_KPQ
0.25 ; PPV_KQPP
0.25 ; PPV_DELTAQP
0.25 ; PPV_GAMA
0.25 FIX ; PPV_KDE
$SIGMA
1 FIX ; EPS1

$SUBR ADVAN13 TOL=6
$MODEL
COMP (C)
COMP (PT)
COMP (Q)
COMP (QP)

$PK

   PT0=THETA(1)*EXP(ETA(1))
   Q0=THETA(2)*EXP(ETA(2))
   LAMBDAP=THETA(3)*EXP(ETA(3))
   KPQ=THETA(4)*EXP(ETA(4))
   KQPP=THETA(5)*EXP(ETA(5))
   DELTAQP=THETA(6)*EXP(ETA(6))
   GAMA=THETA(7)*EXP(ETA(7))
   KDE=THETA(8)*EXP(ETA(8))
   K=100

;Initial conditions
   A_0(2)=PT0
   A_0(3)=Q0

$DES
   C=A(1)
   PT=A(2)
   Q=A(3)
   QP=A(4)

   DPSTAR=PT+Q+QP

   DADT(1) = -KDE*C ; conc in KPD effect compartment
   DADT(2) = LAMBDAP*PT*(1-DPSTAR/K) + KQPP*QP - KPQ*PT - GAMA*PT*KDE*C ; proliferating
cells
   DADT(3) = KPQ*PT - GAMA*Q*KDE*C ; quiescent cells
   DADT(4) = GAMA*Q*KDE*C - KQPP*QP - DELTAQP*QP ; damage quiescent cells

$ERROR

   PSTAR=A(2)+A(3)+A(4)

   Y=PSTAR + THETA(9)*ERR(1)

$TABLE ID TIME Y
ONEHEADER NOPRINT FILE=tgi.fit
```

Monolix

```
DESCRIPTION:
CCR2012_TGI_p5p2.mlxtran

DATA:
  path = "%MLXPROJECT%/../Data/",
  file ="dataLGG_PCV_nh2_nohead.csv",
  headers = {ID,TIME,AMT,Y,MDV,EVID},
  columnDelimiter = ","

INDIVIDUAL:
  DELTAQP = {distribution=logNormal, iiv=yes},
  GAMA = {distribution=logNormal, iiv=yes},
  KDE = {distribution=logNormal, iiv=yes},
  KPQ = {distribution=logNormal, iiv=yes},
  KQPP = {distribution=logNormal, iiv=yes},
  LAMBDAP = {distribution=logNormal, iiv=yes},
  PT0 = {distribution=logNormal, iiv=yes},
  Q0 = {distribution=logNormal, iiv=yes}

STRUCTURAL_MODEL:
  file = "mlxt:TGI",
  path = "%MLXPROJECT%",
  output = {PSTAR}

OBSERVATIONS:
   Pstar = {type=continuous, prediction=PSTAR, error=constant}

TASKS:
  ; settings
  globalSettings={
     withVariance=no,
     settingsAlgorithme="%MLXPROJECT%/p5p2_algorithms.xmlx",
     resultFolder="%MLXPROJECT%/results_p5p2"},
  ; workflow
  estimatePopulationParameters(
     initialValues={
        pop_DELTAQP = 0.008,
        pop_GAMA = 0.729,
        pop_KDE = 0.24,
        pop_KPQ = 0.03,
        pop_KQPP = 0.0031,
        pop_LAMBDAP = 0.121,
        pop_PT0 = 7.13,
        pop_Q0 = 41.2,
        a_Pstar = 2.93,
        omega_DELTAQP = 0.75,
        omega_GAMA = 1.15,
        omega_KDE = 0.5 [method=FIXED],
        omega_KPQ = 0.76,
        omega_KQPP = 0.97,
        omega_LAMBDAP = 0.72,
        omega_PT0 = 0.94,
        omega_Q0 = 0.54
        } ),
  estimateFisherInformationMatrix( method={stochasticApproximation} ),
  estimateLogLikelihood(method={importantSampling, linearization}  ),



INPUT:
parameter={PT0,Q0,LAMBDAP,KPQ,KQPP,DELTAQP,GAMA,KDE}
PK:
compartment(amount=C)
iv()
EQUATION:
t0=0
PT_0=PT0
Q_0 = Q0
K = 100
PSTAR = PT+Q+QP
ddt_C = -KDE*C
ddt_PT = LAMBDAP*PT*(1-PSTAR/K) + KQPP*QP - KPQ*PT - GAMA*PT*KDE*C
ddt_Q = KPQ*PT - GAMA*Q*KDE*C
ddt_QP = GAMA*Q*KDE*C - KQPP*QP - DELTAQP*QP
OUTPUT:
output=PSTAR
```

```
<monolix>
<algorithms seed="123456">
<logLikelihood>
<M_ls value="20000"/>
<optimize value="0"/>
<Mtest_ls value="2000"/>
<tdof_fixed value="5"/>
<tdof_opt value="1,2,5,10,15"/>
<tdof value="5"/>
<MMindiv value="10"/>
<timeEstimator value="auto"/>
</logLikelihood>
<populationParameters>
<kdisp value="50"/>
<K0 value="5"/>
<iop_sa value="1"/>
<coef_sa value="0.95,0.95"/>
<rK2 value="0.1"/>
<nu value="2,0,2,2"/>
<rmcmc value="0.3"/>
<fr_rmcmc value="0.4"/>
<vna value="500,200"/>
<va value="0,1"/>
<niter_mcmc value="20"/>
<iop_Kauto value="1,1"/>
<IK value="100,50"/>
<dU value="5"/>
<nmc value="1"/>
<kde0 value="28"/>
<maxIter_initEta value="50"/>
<no_const_error value="5"/>
<no_autocorr value="50"/>
<optim_iter value="20"/>
<freq_error_c value="20"/>
<rG value="0.5"/>
<DGamma2_phi0_ini value="1"/>
<sign_df value="10"/>
<strategy value="2"/>
<niter_cov value="4"/>
<no_mixture value="100"/>
<em_niter value="10"/>
<sd_alpha value="2"/>
<sd_ratio value="4"/>
  <min_prop value="0.1"/>
  <acceptanceRatio value="0.85"/>
  <nu_times value="2"/>
  <maxIterForTimes value="200"/>
  <iop_nmc value="false"/>
</populationParameters>
<individualParameters>
 <kdisp value="50"/>
 <nu value="2,0,2,2"/>
 <fr_rmcmc value="0.4"/>
 <rmcmc value="0.3"/>
 <l_mcmc value="50"/>
 <r_mcmc value="0.05"/>
 <mixt_iter value="100"/>
 <take_eta_ratio value="0.5"/>
 <nb_max_simulated_eta value="50"/>
 <take_prior_iterations value="1"/>
 <take_prior_ratio value="0.1"/>
 <nb_max_simulated_theta value="1000"/>
 <timeEstimator value="auto"/>
 <tempI_hmm value="10"/>
 <tempF_hmm value="10"/>
 <kiter_hmm value="50"/>
 <niter_hmm value="200"/>
</individualParameters>
<fisherInformationMatrix>
 <timeEstimator value="auto"/>
 <nu value="2,0,2,2"/>
 <fr_rmcmc value="0.4"/>
 <rmcmc value="0.3"/>
 <nburn_iter value="10"/>
 <l_mcmc value="50"/>
 <retries value="5"/>
 <ntest value="1"/>
 <Nsim value="100000"/>
 <reldiff value="0.001"/>
</fisherInformationMatrix>
<linearization>
 <Nsim value="100000"/>
 <reldiff value="0.001"/>
 <mindiff value="1e-010"/>
</linearization>
<results>
 <timeEstimator value="auto"/>
 <max_vectorsize value="Inf"/>
 <Knpde value="500"/>
 <Knpde_ode value="100"/>
 <Kvpc value="100"/>
 <Kvpc_ode value="50"/>
 <nx value="100"/>
</results>
</algorithms>
</monolix>
```

Modelling Description Language

Overview

- MDL – A New Vision for Modelling
  - Same code for multiple execution targets (e.g. NONMEM, Monolix)
  - Independent model, data, parameter, task objects
- Object Groups
  - Modelling Object Group (MOG)
  - TEL Object Group (TOG)
- MDL Example: The TGI model
  - Demonstrate the objects and their components
- Integration of Object Groups with MDL-IDE and Repository

Modelling Description Language
Design Objectives

- MDL
  - Target independent model code
    - Usable with different targets (e.g. NONMEM, Monolix)
  - Independent model, data, parameter, task objects
    - Plug & Play

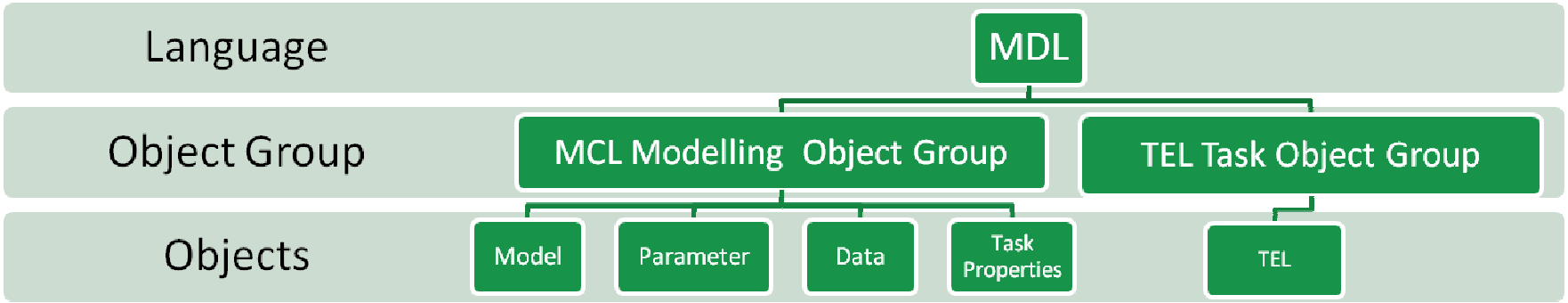
Modelling Description Language

Object Groups

The MDL is divided into two object groups

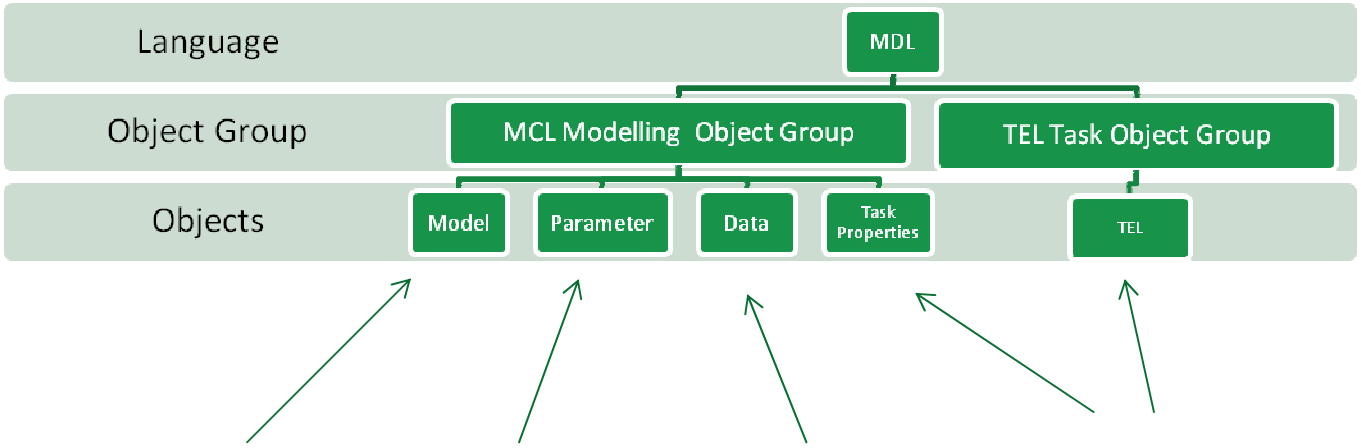
Modelling Object Group: The Model Coding Language (MCL) describes the model, data, parameters and task properties.

Task Object Group: The Task Execution Language (TEL) describes the task execution steps.



Modelling Description Language

Plug & Play Objects



1. Tham 2008
2. Claret 2010
3. Ribba 2012

1. Drug A
2. Drug B
3. Drug X

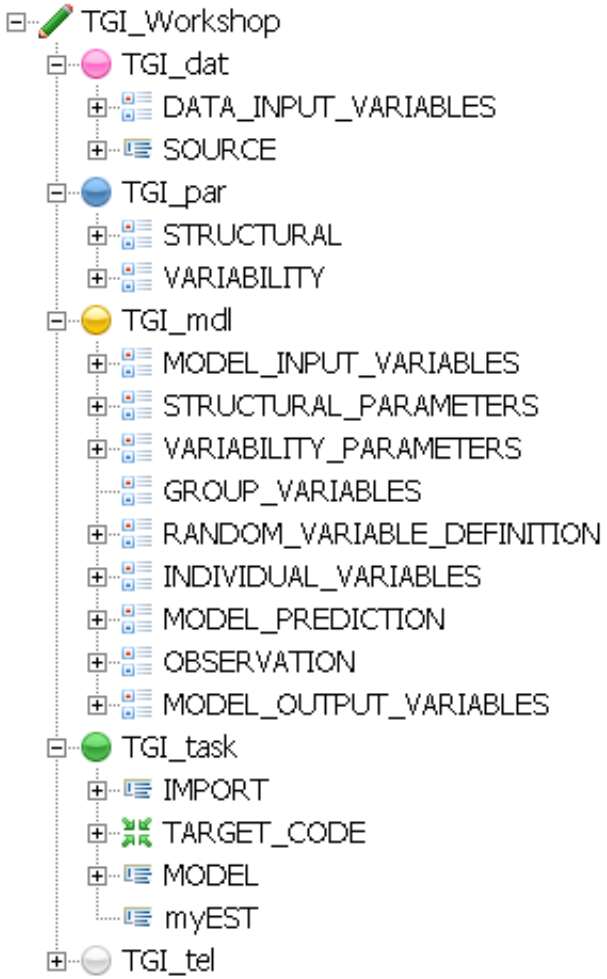
1. Phase 2
2. Phase 3-1
3. Phase 3-2

1. Estimation
2. CTS
3. Optimal Design

Model Coding Language

Objects and Blocks

Outline View



Blocks of data object

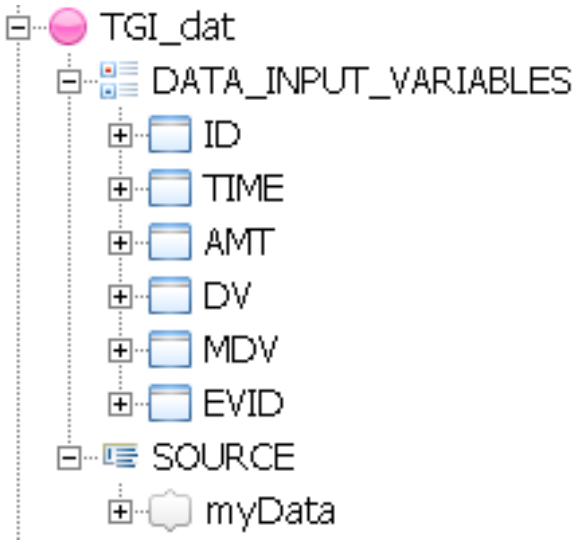
Blocks of parameter object

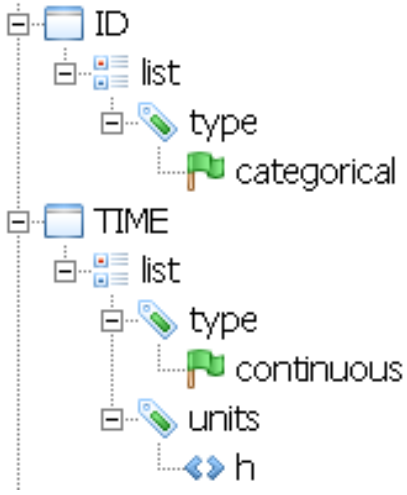
Blocks of model object

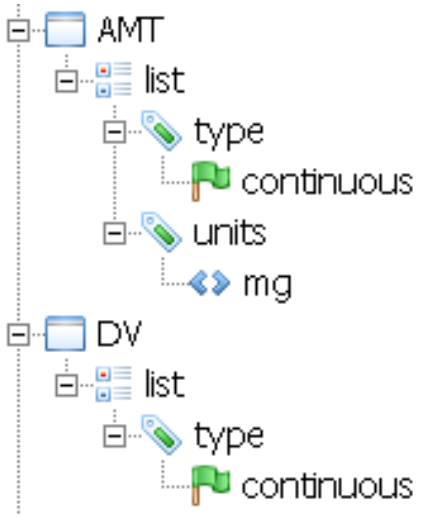
Blocks of task properties object

Model Coding Language
Blocks, Variables, Lists, Attributes
Data Object Outline

The DATA_INPUT_VARIABLES block describes the data variables that may be used by a modelling task. Each variable has a name (e.g. ID) and a list of attributes.
The SOURCE block defines where the data is located (e.g. in a file, from a URL, created by an R script or coded inline)







Model Coding Language
Blocks, Variables, Lists, Attributes
Code defining the Data Object

```
TGI_dat = dataobj{
### Data object
   DATA_INPUT_VARIABLES{
      ID=list(type=categorical)
      TIME=list(type=continuous,units="h")
      AMT=list(type=continuous,units="mg")
      DV=list(type=continuous)
      MDV=list(type=categorical)
      EVID=list(type=categorical)
   }# end DATA_INPUT_VARIABLES

   SOURCE{
      myData=list(
            file="../Data/dataLGG_PCV_nh2.csv",
            inputformat=nonmemFormat,
            ignore="#")
   }# end SOURCE

} # end data object
```

DATA_INPUT_VARIABLES block defines the data items and attributes

SOURCE block defines the source of the data in a file on the local computer

Each data variable may be described with optional type and units. The list keyword is used to group the attributes associated with the data variable. These attributes are used to check data integrity and match the data with model object requirements.

MCL – Flexibility
The model object
Define random effects - Alternative 1

```
GROUP_VARIABLES {
    # No covariates used to explain fixed effects
}# end GROUP_VARIABLES

RANDOM_VARIABLE_DEFINITION{
   eta_PPV_PT0 ~ (type=normal, mean=0, var=PPV_PT0,level=ID)
   eta_PPV_Q0 ~ (type=normal, mean=0, var=PPV_Q0,level=ID)
   eta_PPV_LAMBDAP ~ (type=normal, mean=0, var=PPV_LAMBDAP,level=ID)
   eta_PPV_KPQ ~ (type=normal, mean=0, var=PPV_KPQ,level=ID)
   eta_PPV_KQPP ~ (type=normal, mean=0, var=PPV_KQPP,level=ID)
   eta_PPV_DELTAQP ~ (type=normal, mean=0, var=PPV_DELTAQP,level=ID)
   eta_PPV_GAMA ~ (type=normal, mean=0, var=PPV_GAMA,level=ID)
   eta_PPV_KDE ~ (type=normal, mean=0, var=PPV_KDE,level=ID)
   eps_RUV_EPS1 ~ (type=normal, mean=0, var=RUV_EPS1,level=DV)
}# end RANDOM_VARIABLE_DEFINITION

INDIVIDUAL_VARIABLES{
   PT0=POP_PT0*exp(eta_PPV_PT0)
   Q0=POP_Q0*exp(eta_PPV_Q0)
   LAMBDAP=POP_LAMBDAP*exp(eta_PPV_LAMBDAP)
   KPQ=POP_KPQ*exp(eta_PPV_KPQ)
   KQPP=POP_KQPP*exp(eta_PPV_KQPP)
   DELTAQP=POP_DELTAQP*exp(eta_PPV_DELTAQP)
   GAMA=POP_GAMA*exp(eta_PPV_GAMA)
   KDE=POP_KDE*exp(eta_PPV_KDE)
   K=100
}# end INDIVIDUAL_VARIABLES
```

GROUP_VARIABLES may be used to calculate fixed effects.

RANDOM_VARIABLES_DEFINITION defines the distribution of random effects and the level associated with model input variables.

INDIVIDUAL_VARIABLES block defines how the structural parameters and any group variables are linked to the random effects.

MCL – Flexibility
The model object
Define random variables - Alternative 2

```
 GROUP_VARIABLES {
     # No covariates used to explain fixed effects
 }# end GROUP_VARIABLES

RANDOM_VARIABLE_DEFINITION{
 # No random effect variables defined
}# end RANDOM_VARIABLE_DEFINITION

INDIVIDUAL_VARIABLES{
   PT0 ~ (type=logNormal, median=POP_PT0, var=PPV_PT0,level=ID)
   Q0 ~ (type=logNormal, median=POP_Q0, var=PPV_Q0,level=ID)
   LAMBDAP ~ (type=logNormal, median=POP_LAMBDAP, var=PPV_LAMBDAP,level=ID)
   KPQ ~ (type=logNormal, median=POP_KPQ, var=PPV_KPQ,level=ID)
   KQPP ~ (type=logNormal, median=POP_KQPP, var=PPV_KQPP,level=ID)
   DELTAQP ~ (type=logNormal, median=POP_DELTAQP, var=PPV_DELTAQP,level=ID)
   GAMA ~ (type=logNormal, median=GAMA, var=PPV_GAMA,level=ID)
   lnKDE ~ (type=logNormal, median=KDE, var=PPV_KDE,level=ID)
   K=100
}# end INDIVIDUAL_VARIABLES
```

INDIVIDUAL_VARIABLES Distribution of individual variables is defined without using explicit random effect variables.

MCL – External Functions
The model object
ODEs and Model Libraries

```
MODEL_PREDICTION{
    ODE{
        DPSTAR=PT+Q+QP
        C=ode(deriv= -KDE*C ) # conc in KPD effect compartment
        PT=ode(deriv= LAMBDAP*PT*(1-DPSTAR/K) + KQPP*QP - KPQ*PT - GAMA*PT*KDE*C ,init=PTO)
        Q=ode(deriv= KPQ*PT - GAMA*Q*KDE*C ,init=QO) # quiescent cells
        QP=ode(deriv= GAMA*Q*KDE*C - KQPP*QP - DELTAQP*QP ) # damage quiescent cells
    }
    LIBRARY{
        amount=nmadvan(model=13)
    }
    PSTAR=PT+Q+QP
}# end MODEL_PREDICTION
```

MODEL_PREDICTION is where calculations are defined using group and individual variables ("parameters“)

LIBRARY sub-block supplies predictions from external libraries e.g. NONMEM PREDPP, Monolix PKPD

MCL – Flexibility
The model object
Residual Error

Prediction and Random Effects - Alternative 1

```
OBSERVATION{
   Y = PSTAR+RUV_SD*eps_RUV_EPS1
}# end OBSERVATION
```

OBSERVATION block links individual model prediction (PSTAR) and random effect variables for residual error

Observation Distribution - Alternative 2

```
OBSERVATION{
   Y ~ (type=Normal, mean=PSTAR, variance=RUV_EPS1,level=DV)
}# end OBSERVATION
```

OBSERVATION block defines distribution of observations

MCL – Flexibility
The task properties object
Default Estimation Options – Alternative 1

```
TGI_task = taskobj{
### Task Properties object

IMPORT{nmadvan = list(target = NMTRAN_CODE, name="ADVAN",
    param=list(model=0,trans=0, ncmt=0)
)}


    myEST=function(t,m,p,d) {
        EXECUTE{
            command="call nmgo TGI_Workshop"
        }# end EXECUTE

        ESTIMATE {
            target=t
            model=m
            parameter=p
            data=d

            algo="SAEM"
            cov=true

        }# end ESTIMATE
    }# end of myEST

}# end of task object
```

IMPORT block provides information to the MDL-IDE so that it can recognize the nmadvan library

myEST function optionally specifies a command to EXECUTE the target application and defines task specific options e.g. to ESTIMATE

ESTIMATE block provides default estimation method options for any applicable target.

MCL – Flexibility
The task properties object
Direct control of target – Alternative 2

```
myEST=function(t,m,p,d) {
   EXECUTE{
      command="call nmgo TGI_Workshop"
   }# end EXECUTE

   ESTIMATE {
      target=t
      model=m
      parameter=p
      data=d

TARGET_CODE(target=NMTRAN_CODE,location="$ESTIMATION"){***
$EST METHOD=SAEM NBURN=3000 NITER=1000 ISAMPLE=1 NSIG=2 SIGL=6
CTYPE=3 CITER=10 CALPHA=0.05 NOPRIOR=1
NOABORT PRINT=10
GRD=TS(9) FILE=TGI.raw
$COV MATRIX=R PRINT=E UNCONDITIONAL SIGL=8
***} # end TARGET_CODE

   }# end ESTIMATE
}# end of myEST
```

TARGET_CODE blocks specify statements to be passed unchanged to the target scripting language e.g. NM-TRAN $ESTIMATION record to specify the estimation methods and options.

TEL – Task Execution

The TEL object

Modelling with R

TEL is an R package for DDMoRe that allows modellers to manipulate MCL objects, create MOGs and perform tasks with these MOGs e.g. estimation, simulation, optimal design.

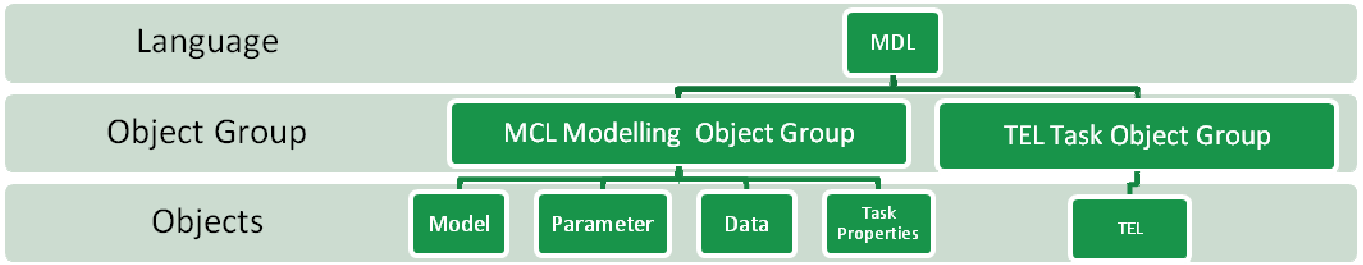
This TEL object uses the myEST function defined in the TGI_task properties object. The arguments to myEST specify the target and define a Modelling Object Group.

```
TGI_tel = telobj{
# Fit model using NONMEM
TGI_fit=TGI_task$myEST(t=NMTRAN_CODE, m=TGI_mdl, p=TGI_par, d=TGI_dat)

} # end of task execution language code
```

The results of the estimation task are returned in a standard output object named TGI_fit. This is an R object that can be passed to other TEL functions or standard R commands.

Standards are being defined for modelling output to allow, for example, use of *Xpose* with BUGS output, or *coda* with NONMEM MCMC output.

MDL – Flexibility
Modelling Object Group and Task Object Group
Links to the Environment



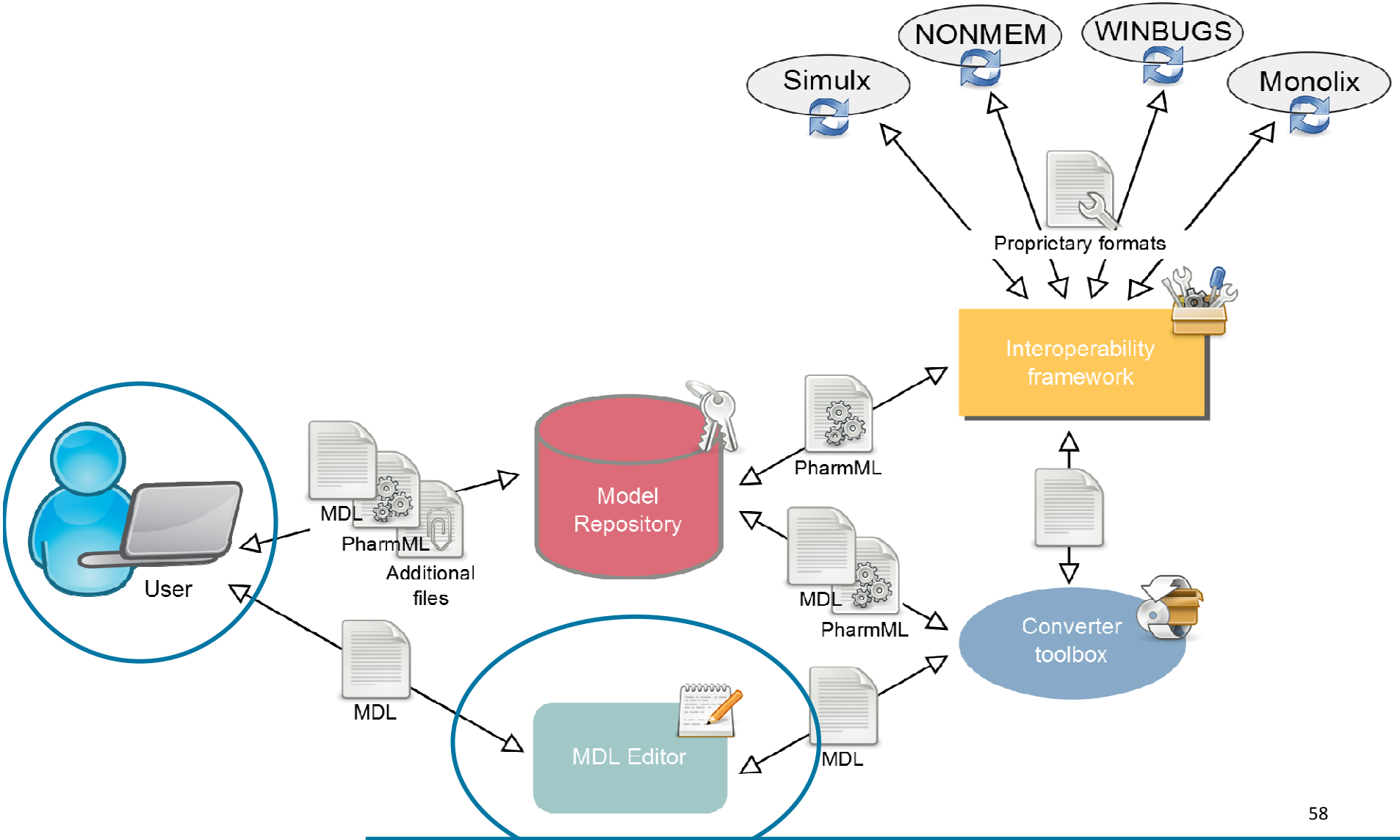
1. Tham 2008
2. Claret 2010
3. Ribba 2012

1. Drug A
2. Drug B
3. Drug X

1. Phase 2
2. Phase 3-1
3. Phase 3-2

1. Estimation
2. CTS
3. Optimal Design

DDMoRe Framework Overview



MDL IDE

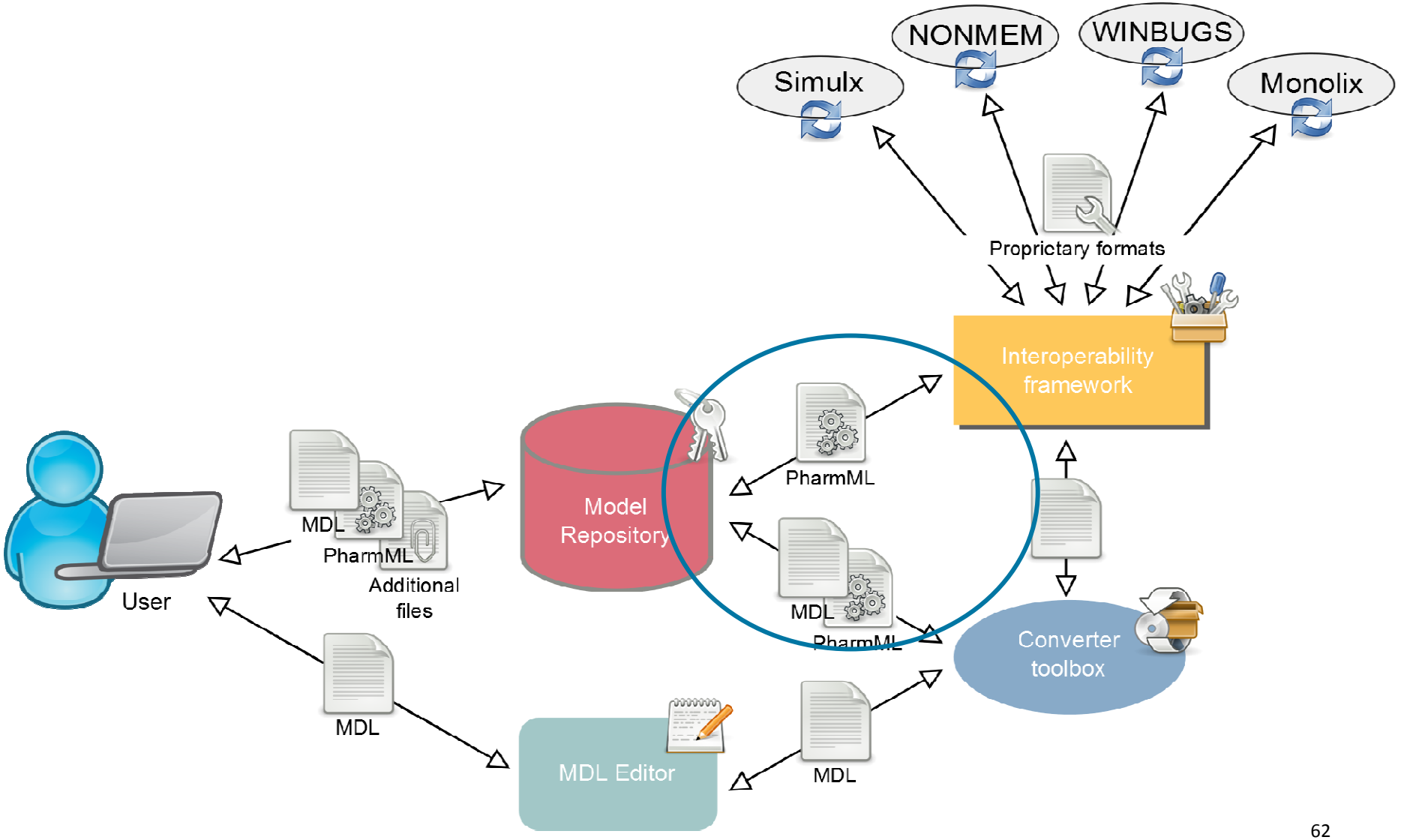
Platform-independent

- MDL implementation
  - Grammar
  - Validation
  - Scoping rules
  - Type checking
  - Automated code formatting
- User interface
  - Content assistant
  - Highlighting of code
  - Show information about errors/warnings
  - Automated fixes
  - Documentation
- Code generator
  - Converter to NONMEM
  - Converter to PharmML
- **Model execution**
  - R
  - NONMEM
  - Simulx

Live demonstration: overview

- Create a new project
- Add/import models and data
- Write code using content assistant
- Navigate using outline view
- Use templates
- Understanding errors and warnings
- Quick fixes
- Convert and execute models

DDMoRe Framework Overview



Introduction to PharmML

A long-standing problem in Pharmacometrics is the lack of a common standard allowing for exchangeability of models between existing software tools, such as Bugs, Monolix, NONMEM and others. PharmML, as part of the DDMoRe interoperability platform, tries to fill this gap.

PharmML stands for 'Pharmacometrics Markup Language'

This new standard provides an encoding platform for approaches currently in use but also attempts to create support for novel elements.

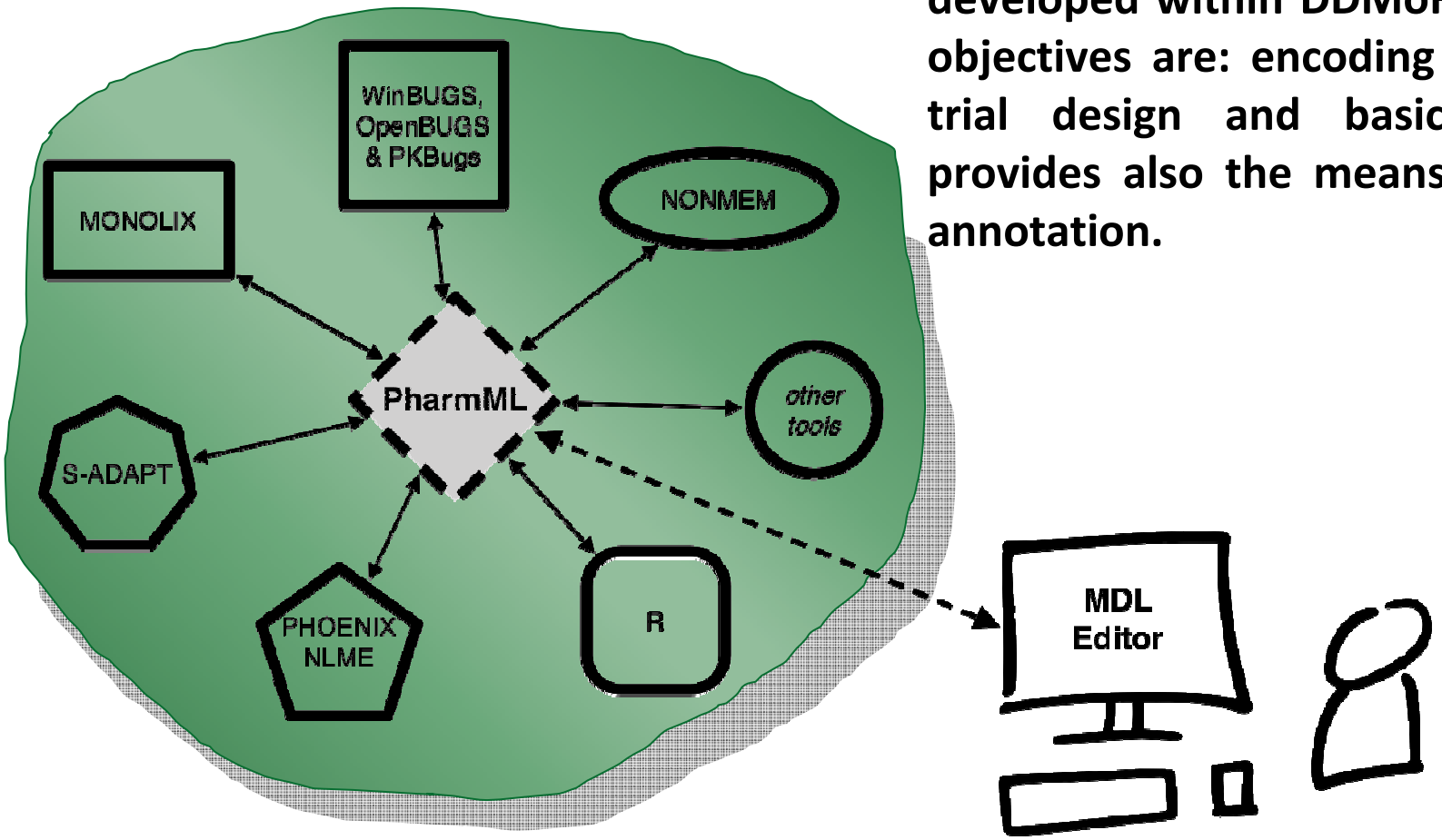
Results

- The current specification supports the exchange of continuous models. Models encoded in this way can be used not only for the standard tasks, such as simulation, estimation but also modelling and exploration.
- The novel clinical trial model provides the modeller with the tools to construct almost arbitrary study designs using only few basic building blocks, such as *Epoch*, *Arm* and *Treatment*.
- Moreover, PharmML is providing a means to annotate an arbitrary element of the model, making effective searching and reasoning on models in the DDMoRe repository possible.

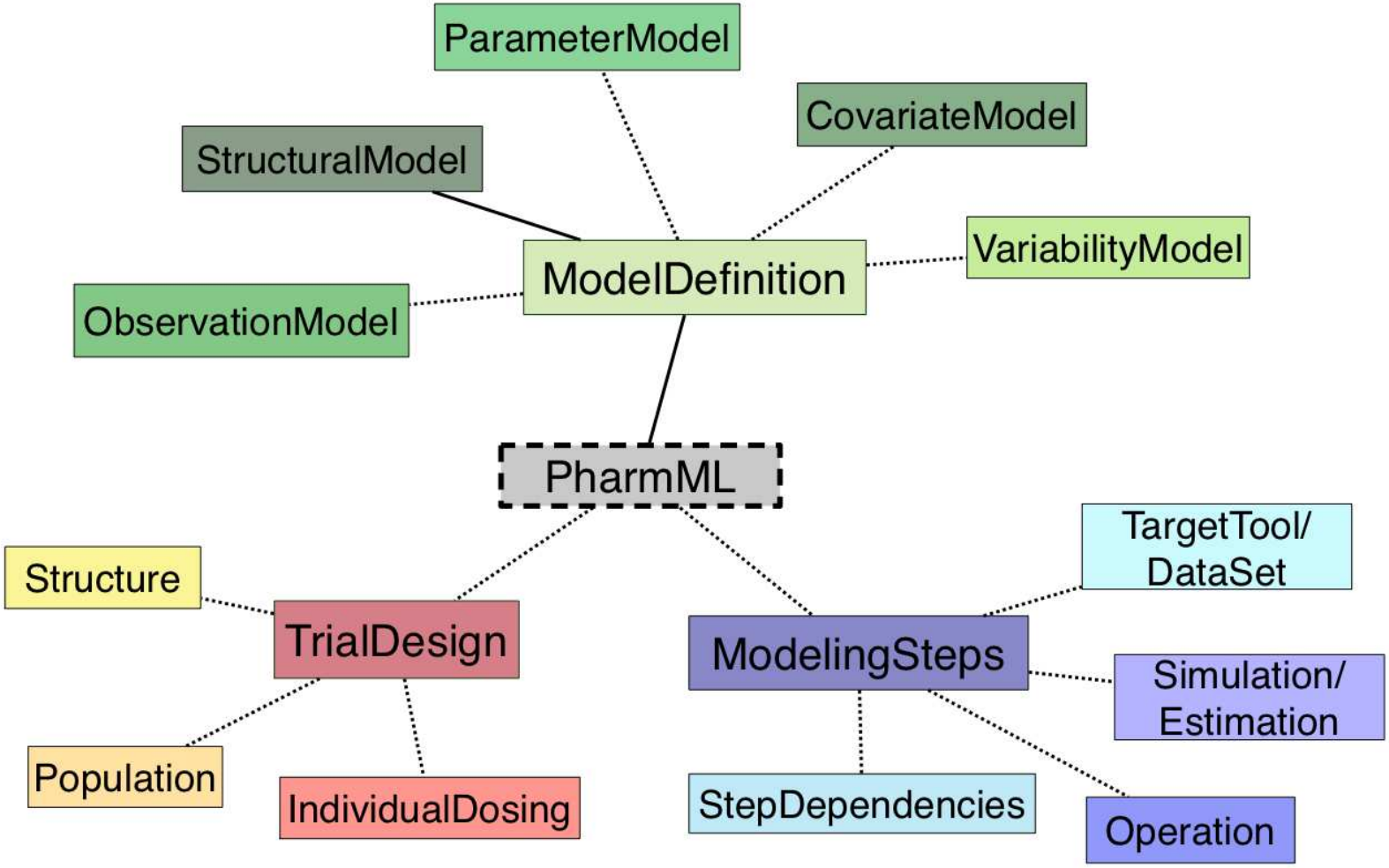
Role of PharmML in the interoperability platform

PharmML is one of the key elements of the interoperability platform developed within DDMoRe. Its main objectives are: encoding of models, trial design and basic tasks. It provides also the means for model annotation.

Machine-to-machine



PharmML language organization



There are three basic blocks in PharmML:
'*ModelDefinition*', '*TrialDesign*' and '*ModellingSteps*'

PharmML model in editor view

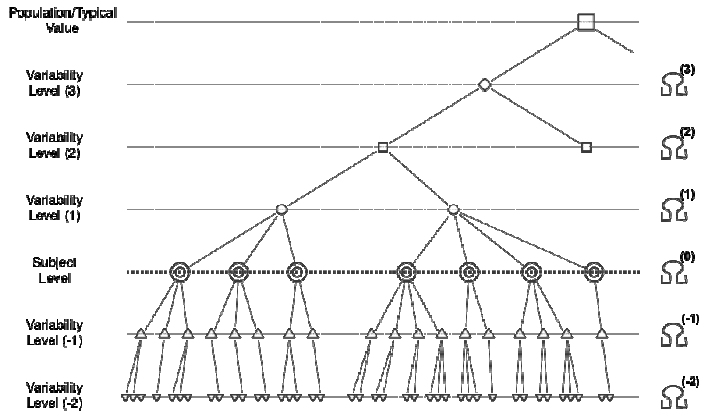
The first three layers of the PharmML hierarchical structure are shown: (a) The root level '*PharmML*', (b) the second level with '*ModelDefinition*', '*TrialDesign*' and '*ModellingSteps*', and (c) the third level within '*ModelDefinition*' and '*TrialDesign*'.

Model Definition

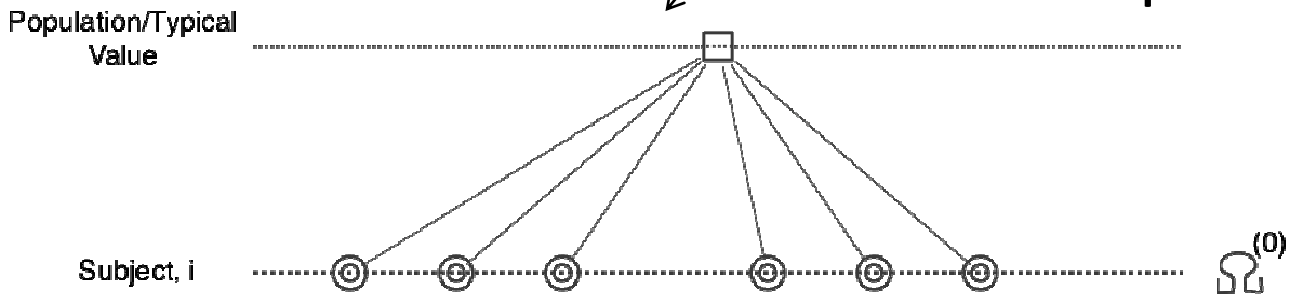
Variability Model – Nested hierarchy

There is only one level of variability
- inter-individual variability (IIV)

$$\lambda_{P_i} = \lambda_{P_{pop}} e^{\eta_{\lambda_P}}; \quad \eta_{\lambda_P} \sim N(0, \omega_{\lambda_P})$$



Complex hierarchy reduces to simple one




```
<!-- VARIABILITY MODEL -->
<VariabilityModel blkId="modelVar" type="parameterVariability">
    <Level symbId="subject"/>
</VariabilityModel>
```

Model Definition

Parameter Model

All parameters are log-normally distributed, e.g.

Random effect

$$\eta_{\lambda_P} \sim N(0, \omega_{\lambda_P})$$

```
<!-- LAMBDAP log-normal distributed -->
<RandomVariable symbId="eta_LAMBDAP">
    <ct:VariabilityReference>
        <ct:SymbRef blkIdRef="modelVar" symbIdRef="indiv"/>
    </ct:VariabilityReference>
    <NormalDistribution xmlns="http://www.uncertml.org/3.0">
        <mean>
            <rVal>0</rVal>
        </mean>
        <stddev>
            <var varId="omega_LAMBDAP"/>
        </stddev>
    </NormalDistribution>
</RandomVariable>
```

Model Definition

Structural Model

ODE for 'C'

$$\frac{dC}{dt} = -KDE \times C$$

Initial condition

$$C(t=0) = C_0$$

```
<!-- STRUCTURAL MODEL - ODEs -->
<StructuralModel blkId="sm1">

    <!-- dC/dt -->
    <ct:DerivativeVariable symbolType="real" symbId="C">
        <ct:Assign>
            <Equation xmlns="http://www.pharmml.org/2013/03/Maths">
                <Binop op="times">
                    <Uniop op="minus">
                        <ct:SymbRef blkIdRef="pm1" symbIdRef="KDE"/>
                    </Uniop>
                    <ct:SymbRef symbIdRef="C"/>
                </Binop>
            </Equation>
        </ct:Assign>
        <ct:IndependentVariable>
            <ct:SymbRef symbIdRef="time"/>
        </ct:IndependentVariable>
        <ct:InitialCondition>
            <ct:InitialValue>
                <ct:Assign>
                    <ct:SymbRef blkIdRef="pm1" symbIdRef="C0"/>
                </ct:Assign>
            </ct:InitialValue>
            <ct:InitialTime>
                <ct:Assign>
                    <ct:Real>0</ct:Real>
                </ct:Assign>
            </ct:InitialTime>
        </ct:InitialCondition>
    </ct:DerivativeVariable>
```

PharmML Trial Design Features

PharmML offers a very flexible structure for the setup of clinical trials – based on a CDISC standard. Using only a few basic elements the modeler can compose virtually any type of design.

The basic building blocks are:

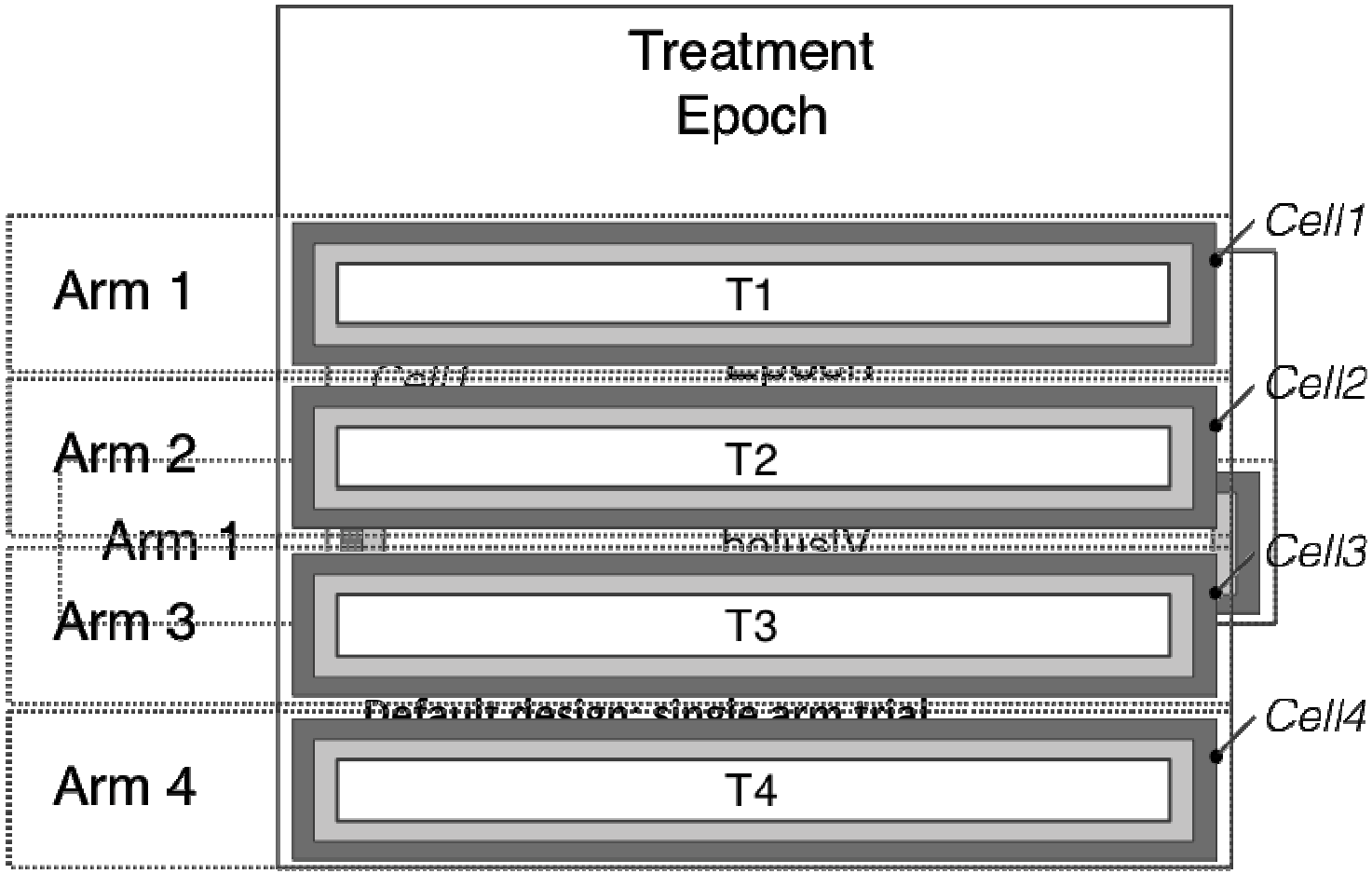
1. ***Epoch*** – basic time interval within a study – for example a treatment or washout.

2. ***Arm*** – represents a path through the study taken by a subject. An arm is composed of a study cell for each epoch in the study.

3. ***Cell*** – describes what is carried out during an epoch in a particular arm. There is only one cell per epoch.

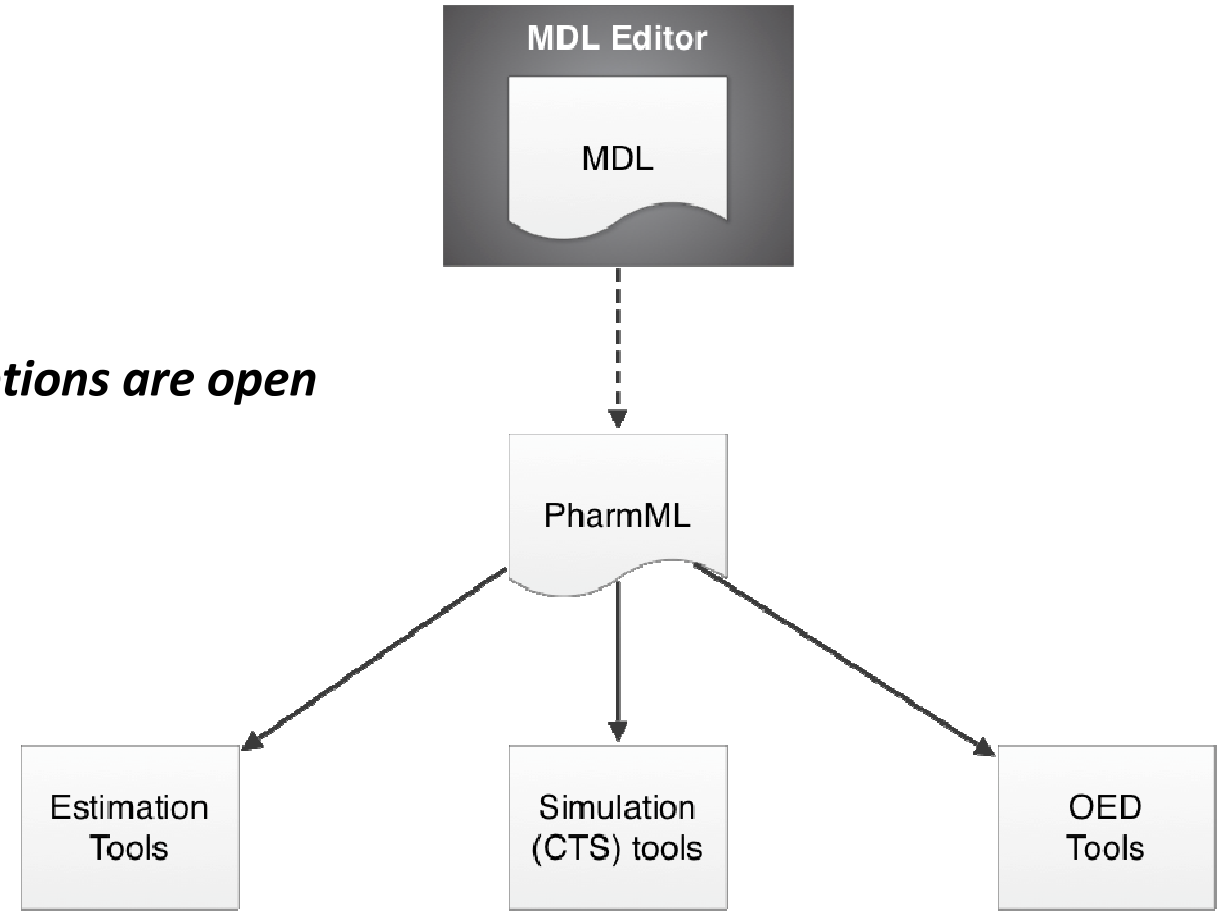
4. ***Activity*** *(Treatment)* – is an action that is taken in the study. Here it is typically a treatment regimen or a washout.

Trial Design – *Ribba et al.* example



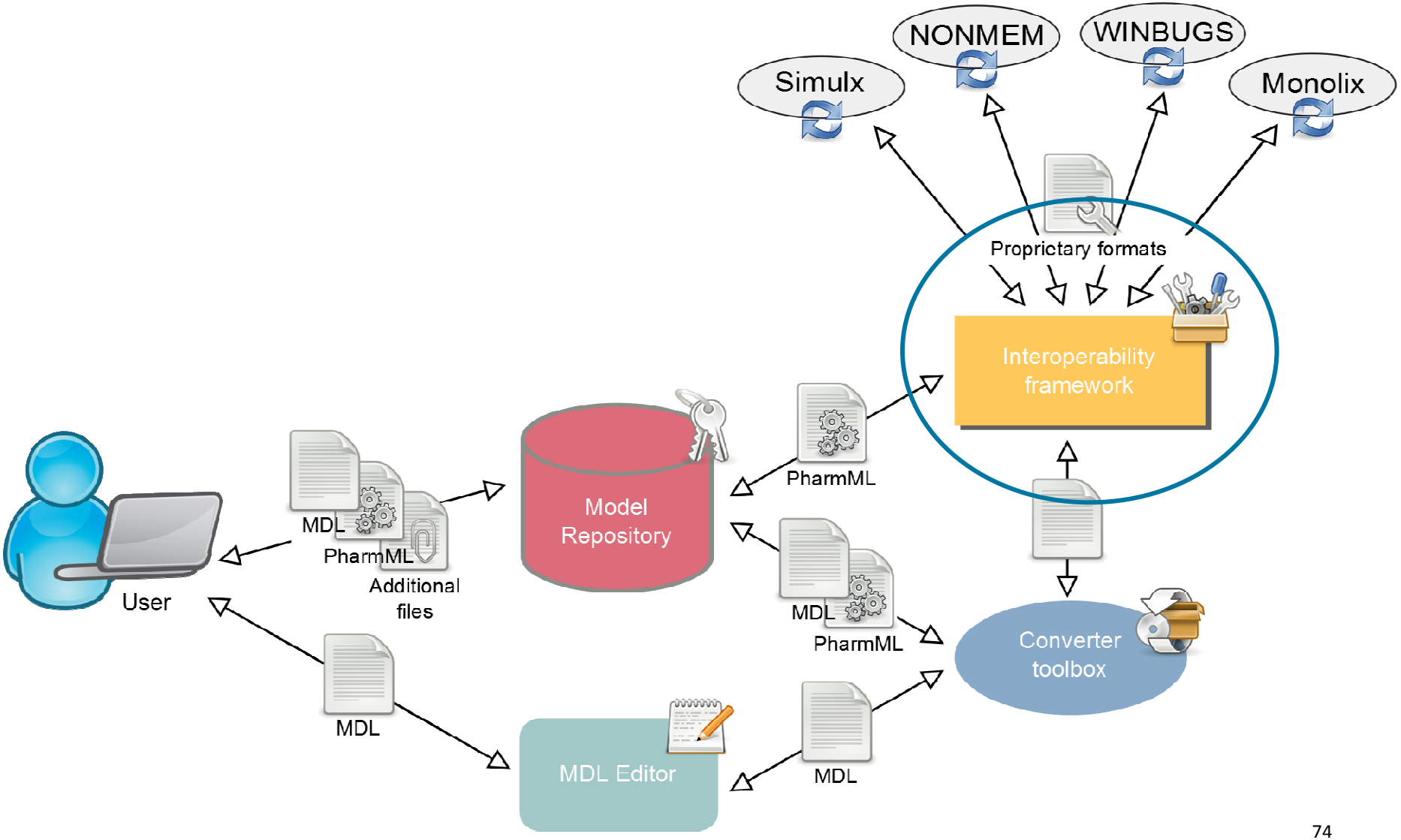
Alternative design, e.g. with four arms, is easily implemented in PharmML – each arm receives a different treatment, T1-T4.

Once you have a model encoded in PharmML…

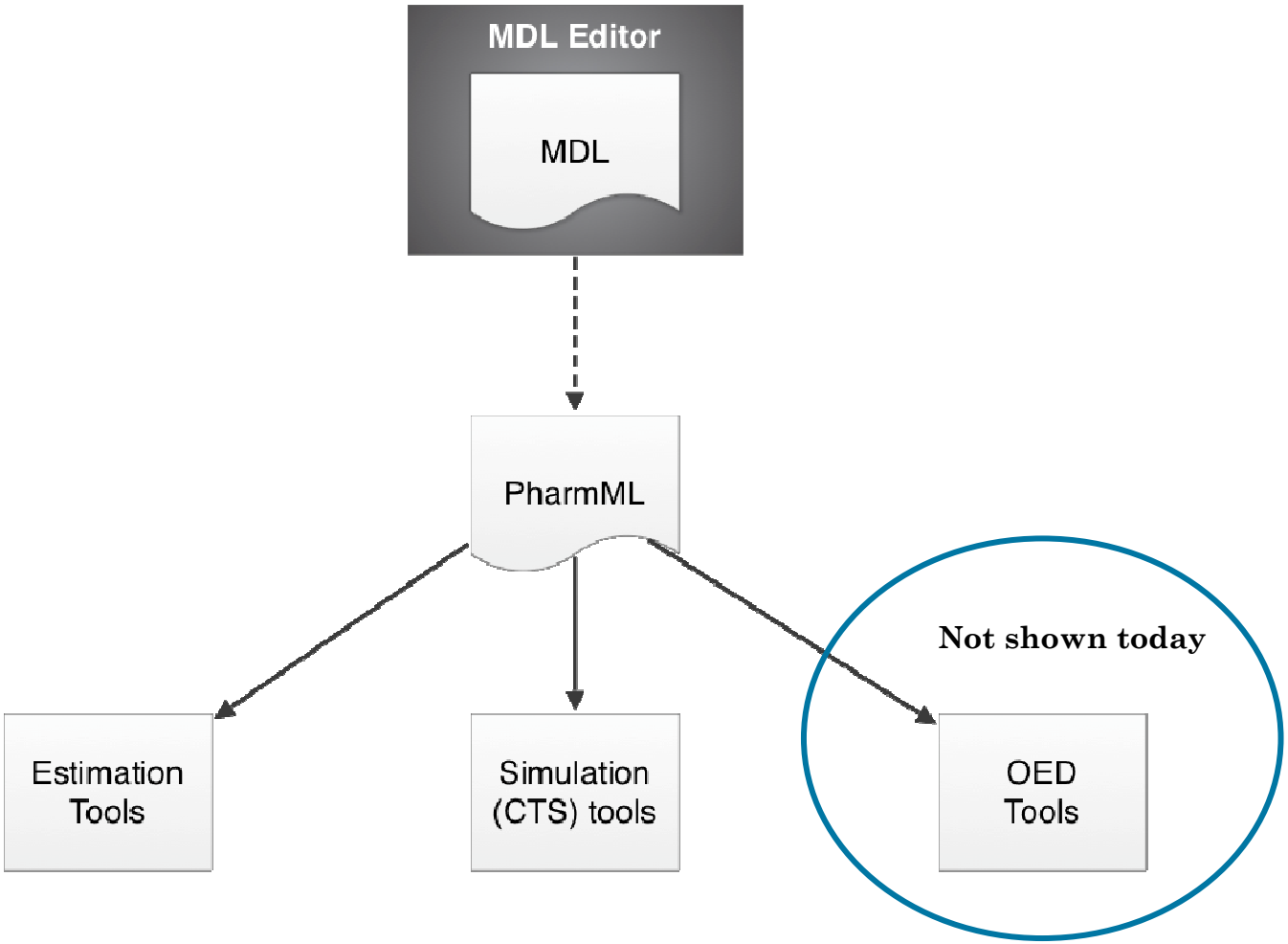


all options are open

DDMoRe Framework Overview



Once you have a model encoded in PharmML…



Estimation using NONMEM

Requires PharmML->NM-TRAN conversion

```
$PROBLEM Example 5 - estimation for growth tumor model (Ribba et al. 2012)

$INPUT ID TIME DV ARM AMT MDV
$DATA example5_data.csv IGNORE=@

$EST METHOD=SAEM INTER NBURN=2000 NITER=1000 ISAMPLE=2 IACCEPT=0.4
$COV

$THETA
(1.0) ; pop_P0
(1.0) ; pop_Q0
(0.0 FIX) ; QP0
(1.0) ; a
(10.0) ; pop_LAMBDAP
(100.0 FIX) ; K
(10.0) ; pop_KDE
(10.0) ; pop_KQPP
(10.0) ; pop_KPQ
(10.0) ; pop_GAMMA
(10.0) ; pop_DELTAQP

$OMEGA
(0.5) ; omega_P0
(0.5) ; omega_Q0
(1.0) ; omega_LAMBDAP
(0.5 FIX) ; omega_KDE
(1.0) ; omega_KQPP
(1.0) ; omega_KPQ
(1.0) ; omega_GAMMA
(1.0) ; omega_DELTAQP
```

Estimation using NONMEM

- NONMEM estimation for the TGI model …
  - … not much fun to watch 'live' ☹

- However we have an estimation we prepared earlier ☺

```
Methods used:
[1] "Stochastic Approximation Expectation-Maximization (No Prior)"

Parameter estimates:
###############
THETAs:
[[1]]
     TH1      TH2      TH3      TH4      TH5      TH6      TH7      TH8      TH9
12.40000 35.60000  0.08190  0.02670  0.00917  0.00242  0.59100  0.08170  2.77000
```

Simulation using Simulx

Requires PharmML->MLXtran conversion

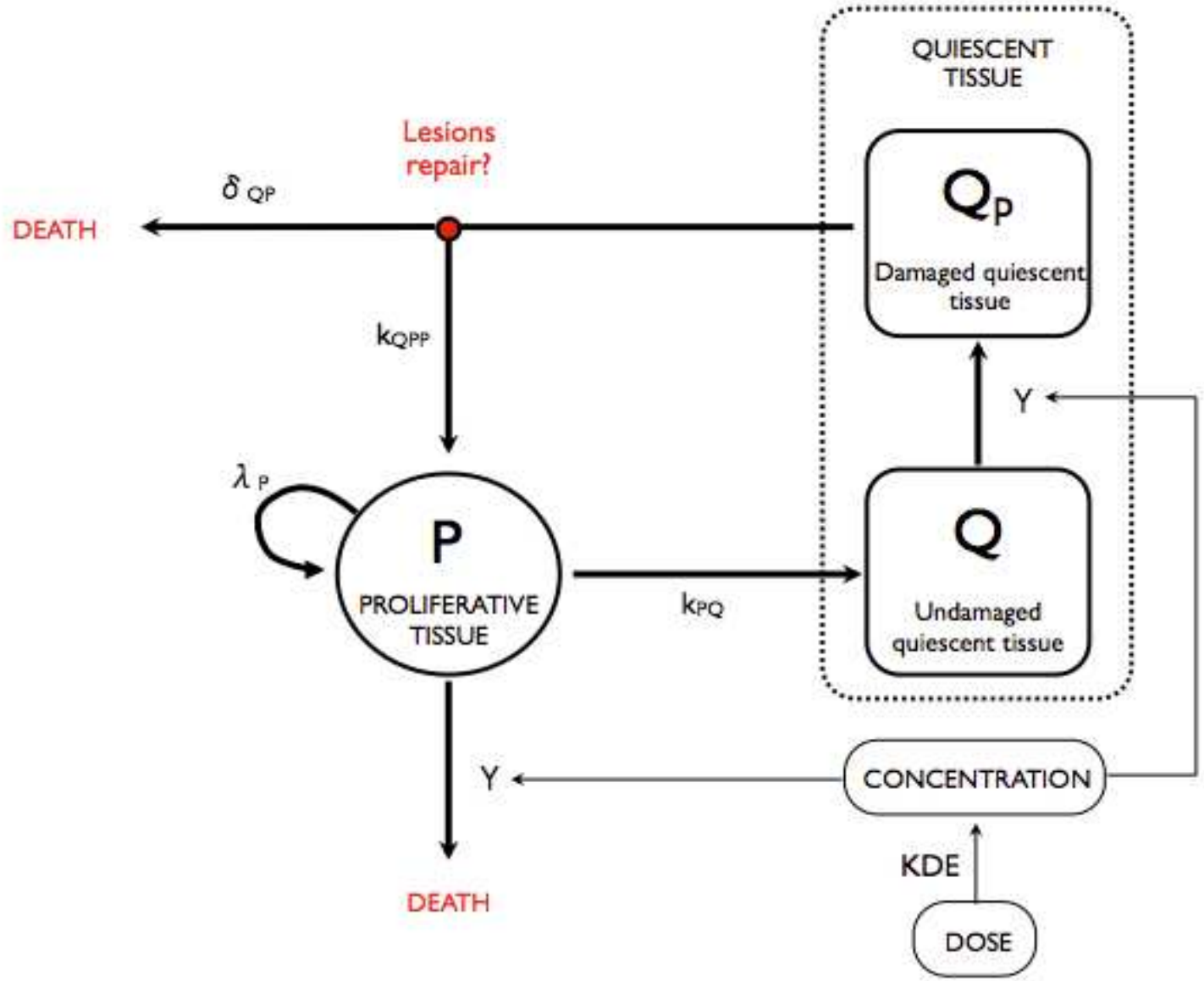
```
[INDIVIDUAL]
input={DELTAQP_pop, omega_DELTAQP, GAMMA_pop, omega_GAMMA, KDE_pop, omega_KDE, KPQ_pop, omega_KPQ, KQPP_pop, omega_KQPP, LAMBDAP_pop, omega_LAMBDAP}

DEFINITION:
DELTAQP = {
     distribution=logNormal,
     reference=DELTAQP_pop,
     standardDeviation=omega_DELTAQP
    }
GAMMA = {
     distribution=logNormal,
     reference=GAMMA_pop,
     standardDeviation=omega_GAMMA
    }
KDE = {
     distribution=logNormal,
     reference=KDE_pop,
     standardDeviation=omega_KDE
    }
KPQ = {
     distribution=logNormal,
     reference=KPQ_pop,
     standardDeviation=omega_KPQ
    }
KQPP = {
     distribution=logNormal,
     reference=KQPP_pop,
     standardDeviation=omega_KQPP
    }
```

Simulation using Simulx

- Simulx is a R function for easily computing predictions and simulating data from both MLXtran and **PharmML** models and calls MLXcompute.

  - This powerful C++ based computation engine allows to solve efficiently complex systems of ordinary differential equations (ODEs) and delayed differential equations (DDEs).

Recap : TGI model schematic view



Recap : Clinical questions

1. Can a model be developed to try to better understand (formulate an hypothesis to explain) the prolonged response phenomenon?

2. Can the model be used to investigate if, given this prolonged response phenomenon, a modification of the therapeutic scheduling can lead to improve the efficacy of the treatment?

Demonstration – Question 1

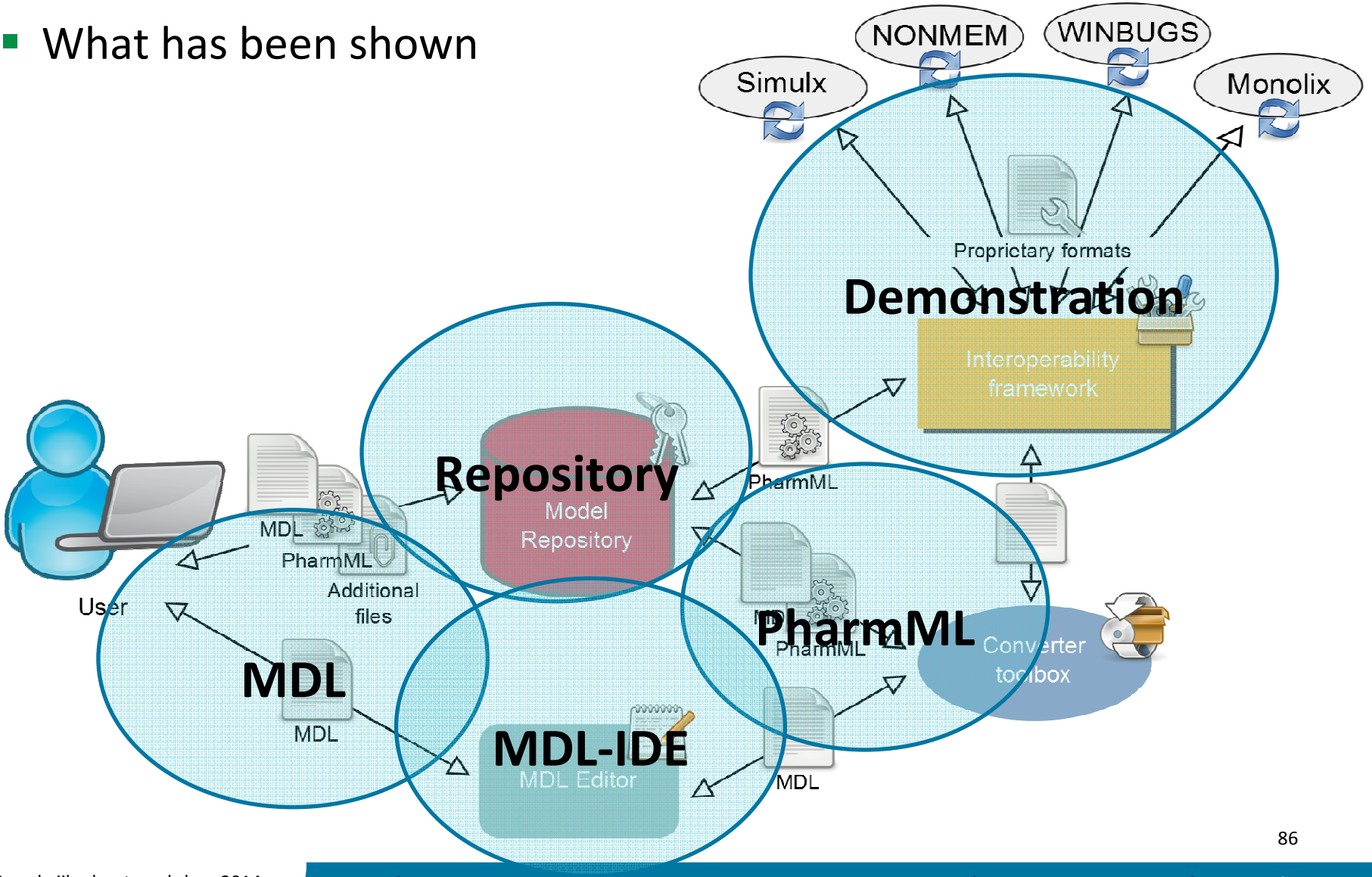
1. Can a model be developed to try to better understand (formulate an hypothesis to explain) the prolonged response phenomenon?
   - Pick the model from the repository
   - Estimate population parameters with NONMEM
   - Use Simulx to simulate a typical individual

Demonstration – Question 2

2. Can the model be used to investigate, given this prolonged response phenomenon, if a modification of the therapeutic scheduling can lead to improve the efficacy of the treatment?
   - Use Simulx to simulate patient's response with a different PCV scheduling

Recap

- What has been shown

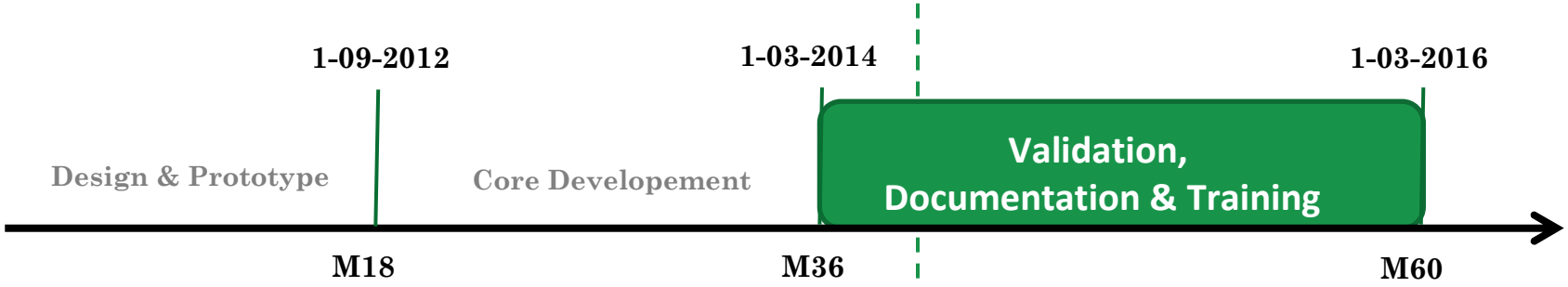


Recap

- What has been shown

- What has not/partly been shown
  - Task Execution Language / “Workflow”
  - Metadata framework / ontologies
  - Prototypes on CTS and MB-AOD
  - Disease area prototype models
  - Plans for a Model Review Group (MRG)

Road map



Public release Nov 2013

- PharmML specification
- MDL specification
- MDL-IDE

Public release Q2 –Q4 2014

- Model Repository
- Simulx
- Framework Demonstrator

Contributors to presentation

- Wendy Aartsen
- Jonathan Chard
- Mihai Glont
- Mats Karlsson
- Niels Rode Kristensen
- Camille Laibe
- Marc Lavielle
- Celine Sarr
- Mike K Smith