NCAConnect: An R Shiny Application for Efficient Processing of Pharmacokinetic Data

MSD: Yanli Sun

Lili Ling

2025-05-09

Disclaimer

This presentation does not necessarily reflect MSD's corporate vision. It reflects the authors' insights and diverse experiences working on analysis and reporting using various programming languages across the industry.

Agenda

1. Introduction

- PK Data Analysis
- Dose Escalation
- Importance of PK Parameters
- Purpose for App Development

2. Traditional Method vs Shiny App

- Traditional Process
- The Shortcomings of Traditional Methods
- App Process
- The Advantages of Using App

3. App Overview and Demo

- Overview
- Demo

4. Conclusion

Agenda

1. Introduction

- PK Data Analysis
- Dose Escalation
- Importance of PK Parameters
- Purpose for App Development

2. Traditional Method vs Shiny App

- Traditional Process
- The Shortcomings of Traditional Methods
- App Process
- The Advantages of Using App

3. App Overview and Demo

- Overview
- Demo

4. Conclusion

PK Data Analysis

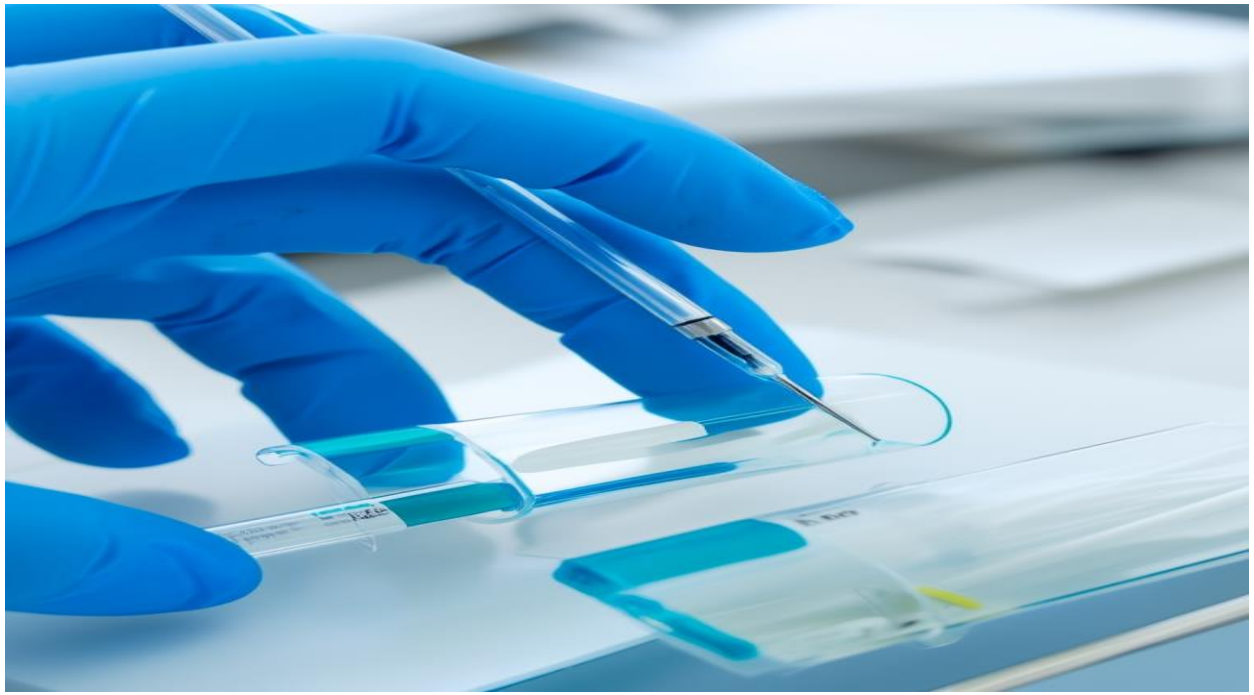
Pharmacokinetics (PK) is the study of the effect of drugs on the body: the time process of absorption, distribution, metabolism and excretion of drugs in the body.

NCA(Non-compartmental Analysis) provides a convenient method for extracting PK parameters from drug concentration time data, which is widely used in drug development and clinical research. The main advantage of NCA lies in its simplicity and intuitiveness, which enables rapid extraction of key pharmacokinetic parameters from experimental data without the need to assume complex drug distribution models.

Dose Escalation

Dose escalation is a fundamental aspect of clinical pharmacology that plays a vital role in drug development. By systematically increasing the dose, researchers can find a balance between maximizing therapeutic benefits and minimizing risks, ultimately determining safer and more effective treatment protocols.

In dose escalation studies, high quality quick turnaround in analytical data for NCA is crucial for faster decision-making.



Importance of PK Parameters

Competitive Edge

PK parameters provide multiple advantages for pharmaceut ical companies and medical institutions in a competitive env ironment, including accelerating drug development, improvi ng market competitiveness, and reducing research and deve lopment risks.

Resource Allocation

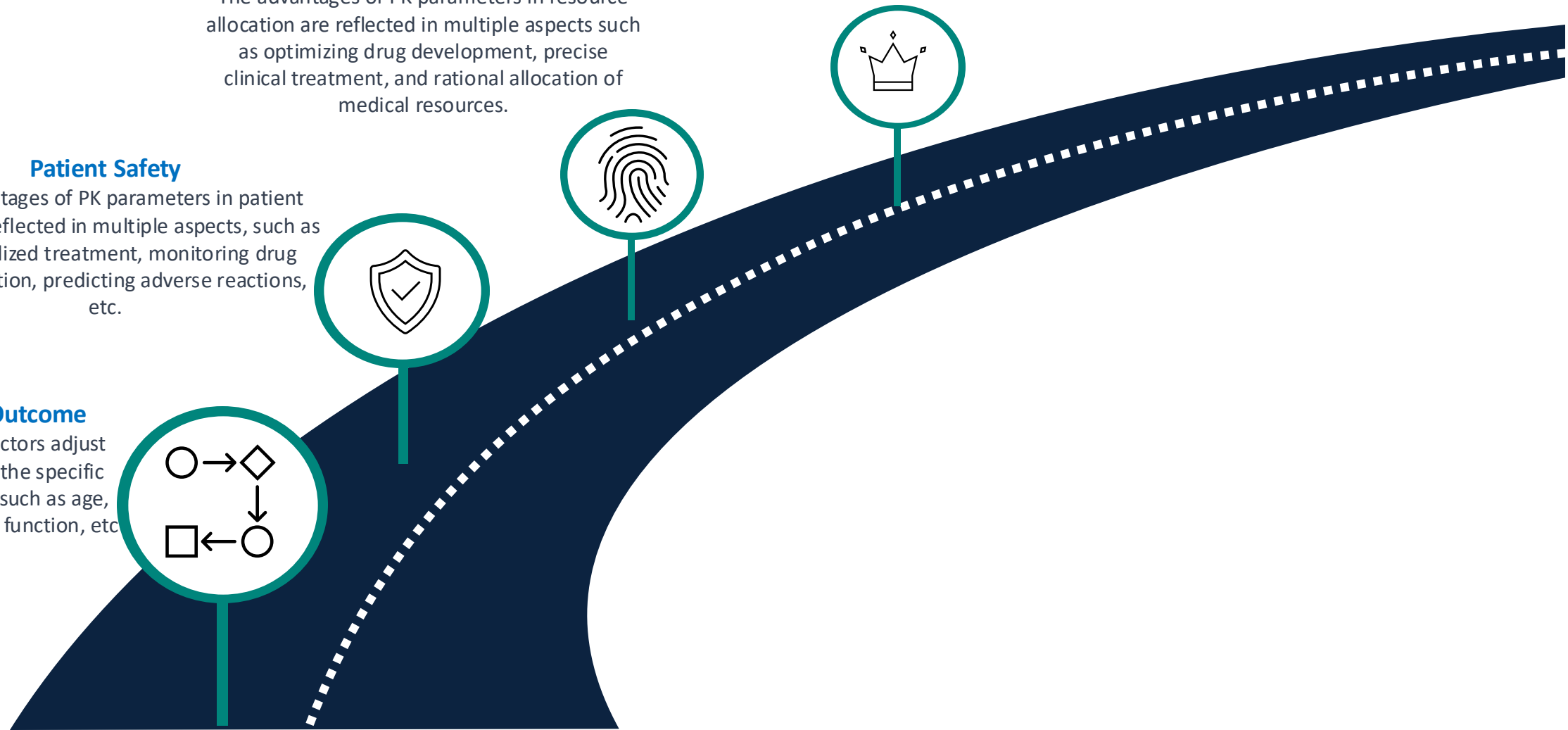
The advantages of PK parameters in resource allocation are reflected in multiple aspects such as optimizing drug development, precise clinical treatment, and rational allocation of medical resources.

Patient Safety

The advantages of PK parameters in patient safety are reflected in multiple aspects, such as individualized treatment, monitoring drug concentration, predicting adverse reactions, etc.

Optimize Trial's Outcome

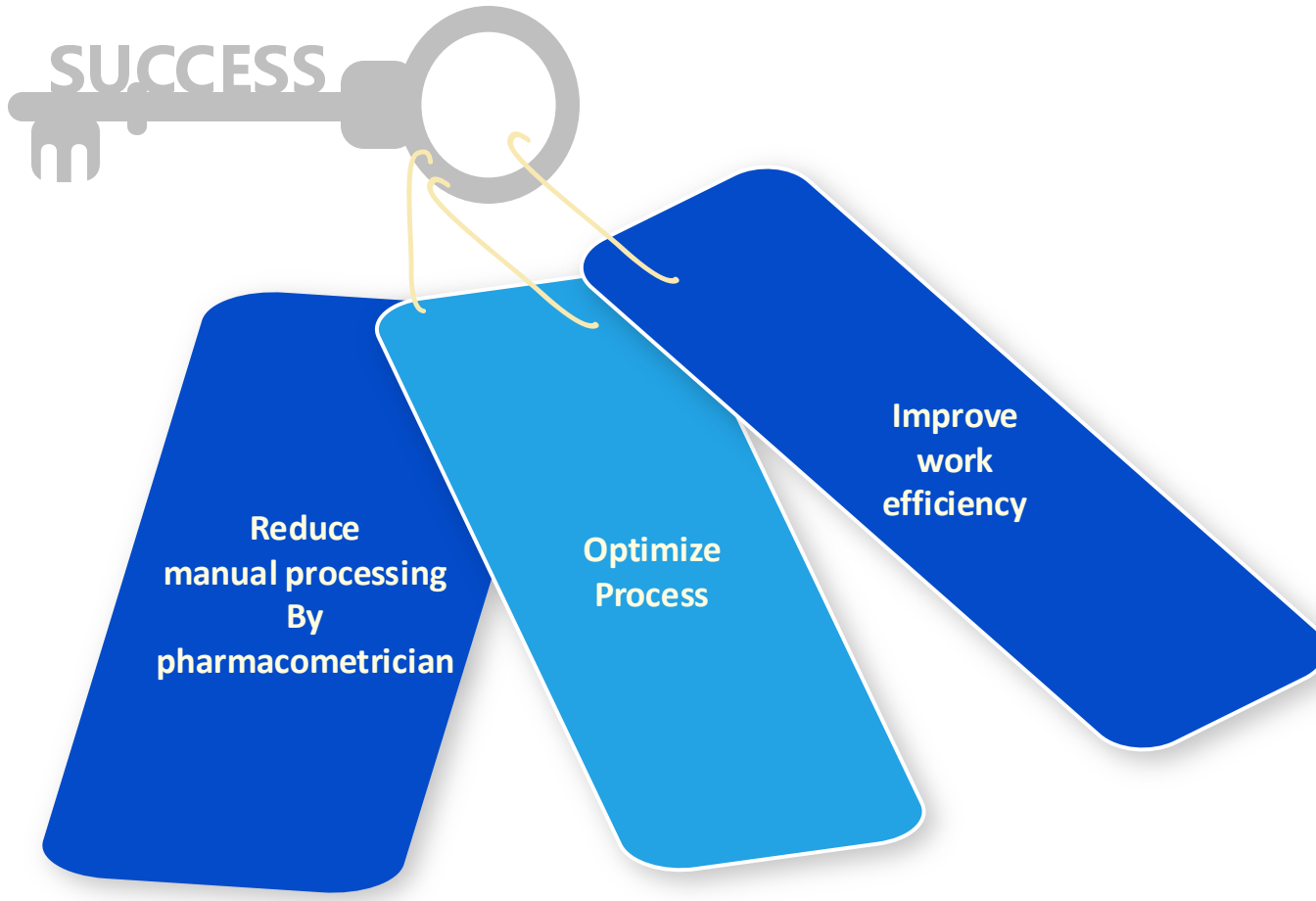
PK parameters help doctors adjust drug dosage based on the specific conditions of patients, such as age, weight, liver and kidney function, etc



Purpose for App Development

Why?

In the evolving landscape of clinical research, the need for efficient and accurate data processing is critical, particularly in dose escalation studies where early dose selection can significantly impact trial progression and outcomes. To meet this demand, we developed an R Shiny application that offers an intuitive interface for uploading, managing, and analyzing data.



Agenda

1. Introduction

- PK Data Analysis
- Dose Escalation
- Importance of PK Parameters
- Purpose for App Development

2. Traditional Method vs Shiny App

- Traditional Process
- The Shortcomings of Traditional Methods
- App Process
- The Advantages of Using App

3. App Overview and Demo

- Overview
- Demo

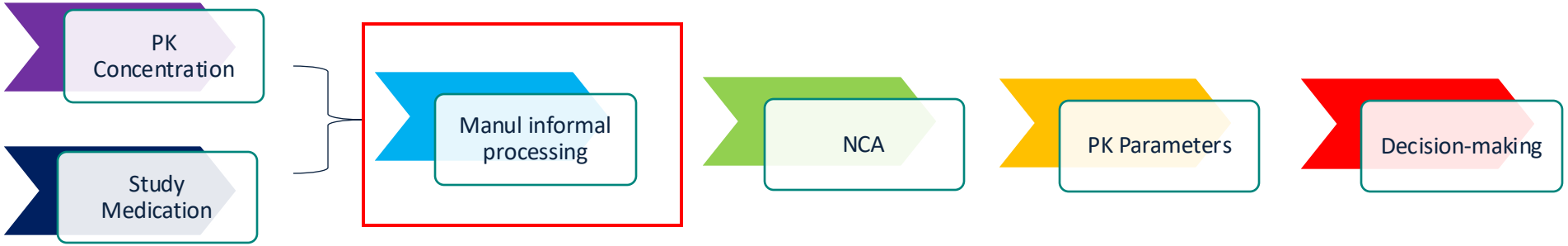
4. Conclusion

Traditional Process

1. Internal standard process on PK analysis：

2. Accelerated Workaround Process:

Due to the specificity of early oncology projects, safety and pharmacokinetic (PK) data must be collected at each dose escalation stage to inform dose escalation decisions.
The generation of ADPC requires collaboration across multiple departments and has long time cycle, Therefore, without the need for formal ADPC analysis, it is necessary to calculate the data used for NCA (Non-Compartmental Analysis) independently:



The Shortcomings of Traditional Methods

Challenges in files integration

- Manually merge multiple documents.
- integration process is time-consuming and laborious.

Inefficiency of manual processing

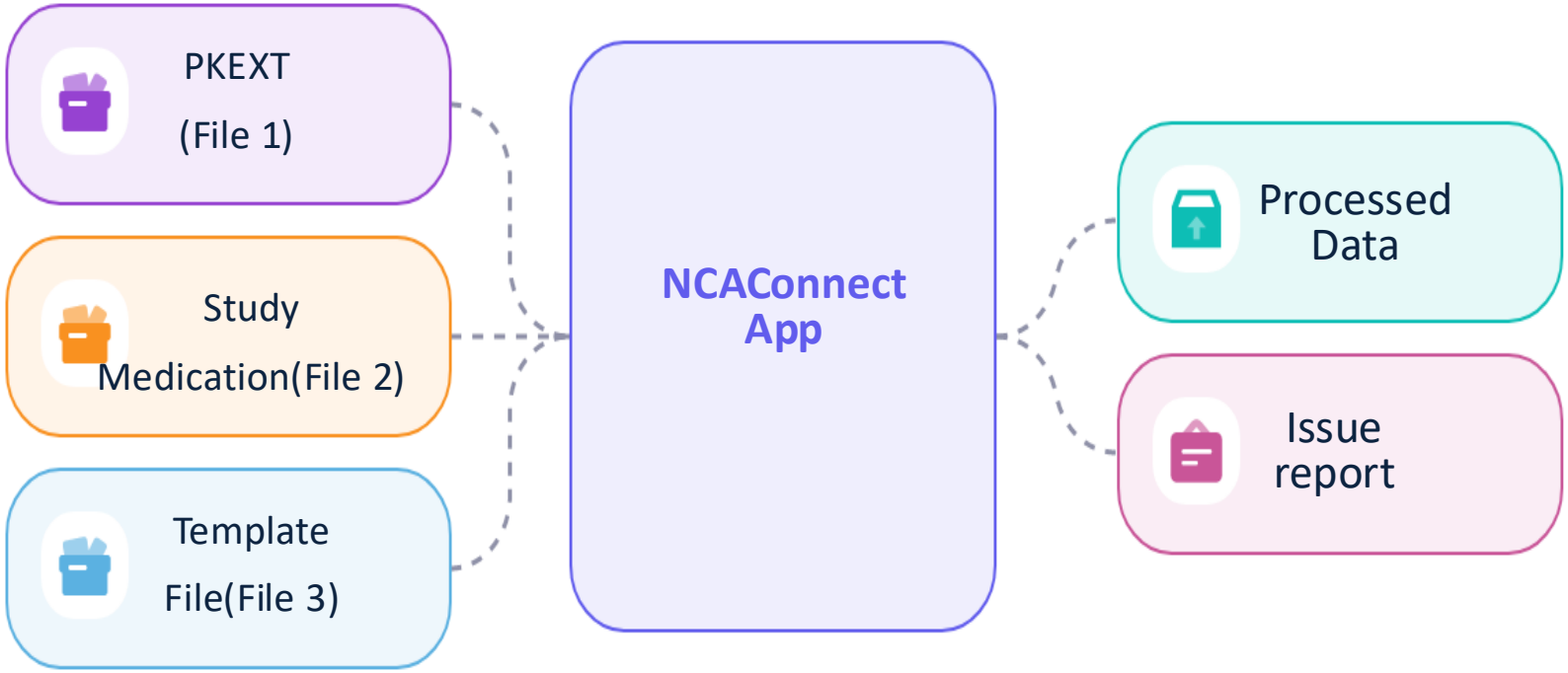
- Manually calculate the required variables.
- Same variable with different logic easy to make mistakes and low efficiency.

Labor-Intensive data verification

- Manual data verification can be time-consuming and labor-intensive.
- Data comparison are relatively cumbersome and inconvenient.

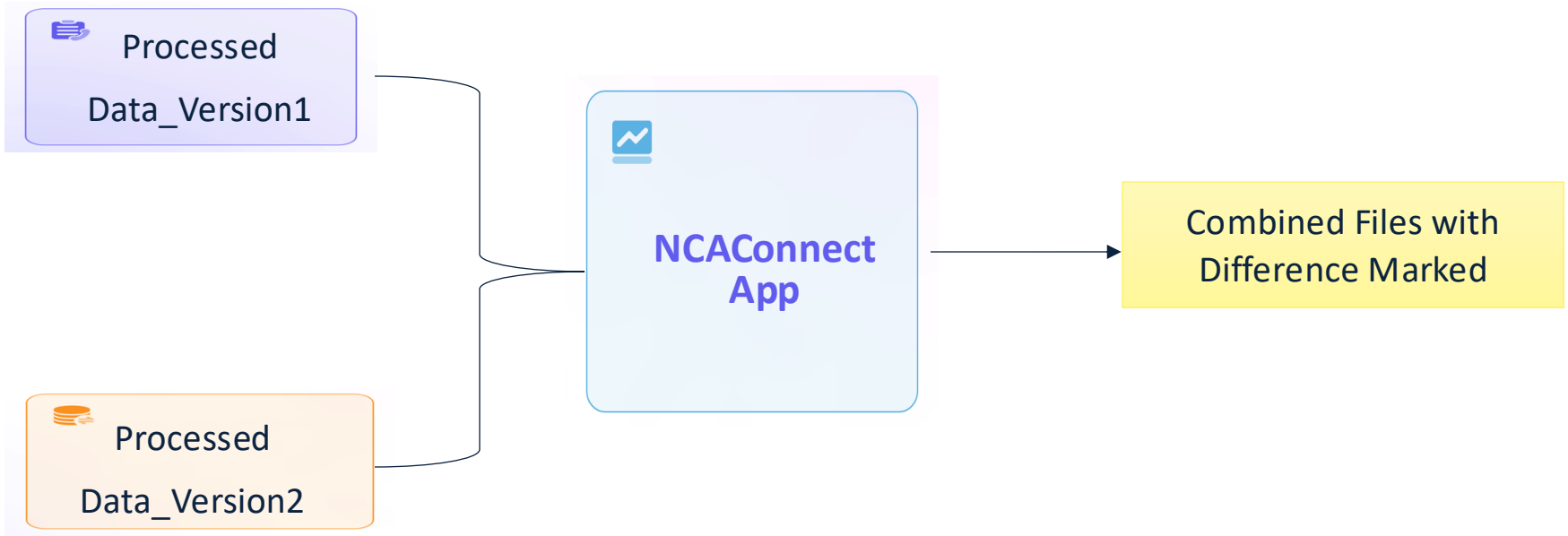
App Process

Function 1- Calculation Usage Process



App Process

Function 2- Data Verification Process



The Advantages of Current Methods (Using App)

Simplified Interface

Through a simple user interface, simply upload multiple source files for quick integration.

Efficiency Improvement

Significantly improved the efficiency of generating data for calculating pharmacokinetic parameters.

Convenient Data Verification

By exporting the issue report, can intuitively and clearly obtain data verification.

Reduce Repetitive Work

By retaining the content of the previous version, we can reduce repetitive work and quickly obtain comparative data.

Agenda

1. Introduction

- PK Data Analysis
- Dose Escalation
- Importance of PK Parameters
- Purpose for App Development

2. Traditional Method vs Shiny App

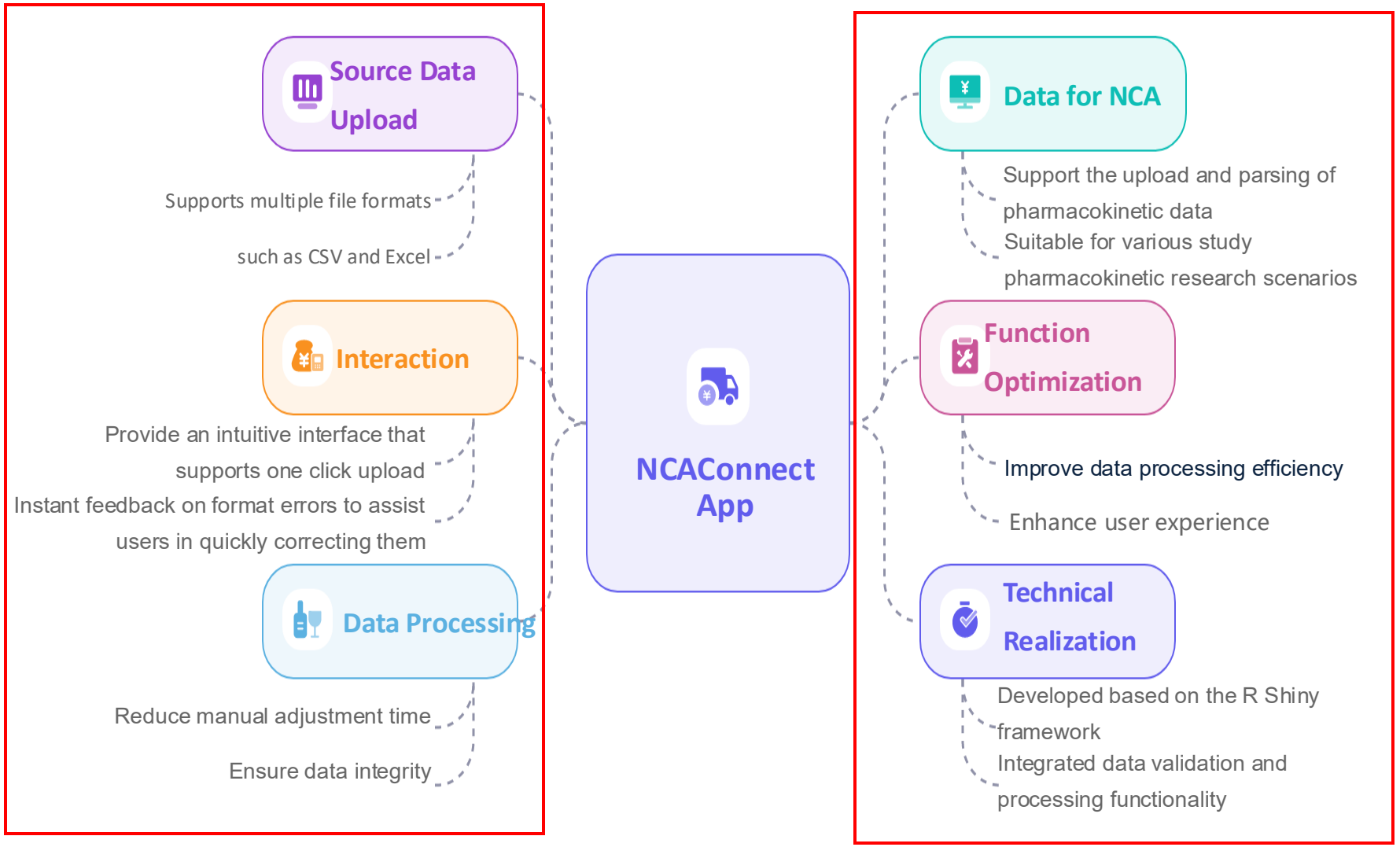
- Traditional Process
- The Shortcomings of Traditional Methods
- App Process
- The Advantages of Using App

3. App Overview and Demo

- Overview
- Demo

4. Conclusion

Overview


Source Data Upload
Supports multiple file formats
such as CSV and Excel
Interaction
Provide an intuitive interface that supports one click upload
Instant feedback on format errors to assist users in quickly correcting them
Data Processing
Reduce manual adjustment time
Ensure data integrity
NCAConnect App
Data for NCA
Support the upload and parsing of pharmacokinetic data
Suitable for various study pharmacokinetic research scenarios
Function Optimization
Improve data processing efficiency
Enhance user experience
Technical Realization
Developed based on the R Shiny framework
Integrated data validation and processing functionality

Overview- I

1. Checking for Missing Variables: A function check_missing_vars is defined to check whether the given data frame contains all the required variables.
2. Error Handling: The tryCatch structure is used to capture and handle errors, ensuring that the program does not stop even if a data frame is missing variables.

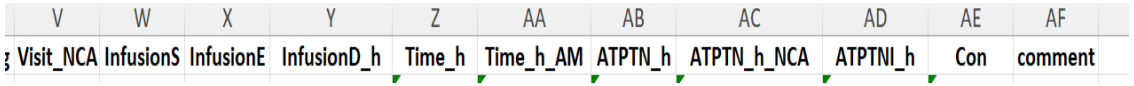
```r
# Define function to check missing variables
check_missing_vars <- function(data_frame, required_vars, df_name) {

  missing_vars <- setdiff(required_vars, names(data_frame))
  if (length(missing_vars) > 0) {
    stop(paste("Missing Variable in", df_name, ":", paste(missing_vars, collapse = ", ")))
  }
}
```

```r
error_occurred <- FALSE

tryCatch({
  check_missing_vars(file1, required_vars_file1, "PKEXT_EDT file")
}, error = function(e) {
  message(e$message)
  error_occurred <<- TRUE
})

tryCatch({
  check_missing_vars(file2, required_vars_file2, "Visit_dose sheet in PKEXT_EDT_template file")
}, error = function(e) {
  message(e$message)
  error_occurred <<- TRUE
})

tryCatch({
  check_missing_vars(file3, required_vars_file3, "Study_Medication file")
}, error = function(e) {
  message(e$message)
  error_occurred <<- TRUE
})

tryCatch({
  check_missing_vars(file4, required_vars_file4, "Study_information sheet in PKEXT_EDT_template file")
}, error = function(e) {
  message(e$message)
  error_occurred <<- TRUE
})

if (!error_occurred) {
  print("All required variable exist")
```

Overview- II

In the data calculation phase, the tidyverse package is commonly used to perform logical calculations and manipulate variables efficiently. The tidyverse is a collection of R packages designed for data science, and it includes several key packages such as dplyr, ggplot2, tidyr, and others.

Key Variable:



V	W	X	Y	Z	AA	AB	AC	AD	AE	AF
Visit_NCA	InfusionS	InfusionE	InfusionD_h	Time_h	Time_h_AM	ATPTN_h	ATPTN_h_NCA	ATPTNI_h	Con	comment

```
#InfusionS/InfusionE
file_com$InfusionS<-str_extract(file_com$`Time ~ Start Date Time`,"\\d+:\\d+:\\d+")
file_com$InfusionE<-str_extract(file_com$`Time ~ Stop Date Time`,"\\d+:\\d+:\\d+")
#InfusionD_h
file_com$InfusionD_h<-round(as.numeric(difftime(as.POSIXct(file_com$InfusionE,format="%H:%M:%S"),as.POSIXct(file_com$InfusionS,format="
#Derive ATPTN_h variable
file_com$ATPTN_h<-as.character(ifelse(file_com$Time_h!=-1,file_com$Time_h1+file_com$InfusionD_h,-1))

#Count the different values of visit1 in file_com according to PKEXTSUBJID and StudyPeriod, and generate variable named Visit2
file_com$Visit2<-ave(file_com$Visit1,file_com$PKEXTSUBJID,file_com$StudyPeriod,FUN=function(x) seq_along(x))
file_com1<-file_com %>%group_by(PKEXTSUBJID,Visit1) %>% slice(n())
file_com1$Visit3<-file_com1$Visit2-lag(file_com1$Visit2)
##If the absolute value of Visit3 is greater than or equal to the Intensity level, the visit value is equal to the previous visit value
file_com1$Visit<-ifelse(file_com1$Visit3<= -file4$`Intensive level`,lag(file_com1$Visit),file_com1$Visit)

#Keep record that less than Intensive level and set up PKEXTRT equal to 'Predose'
file_com1<-file_com1 %>% filter(Visit3<= -file4$`Intensive level`)%>% mutate(PKEXTTPT='Predose')
file_com1$Visit_NCA <-file_com1$Visit
```

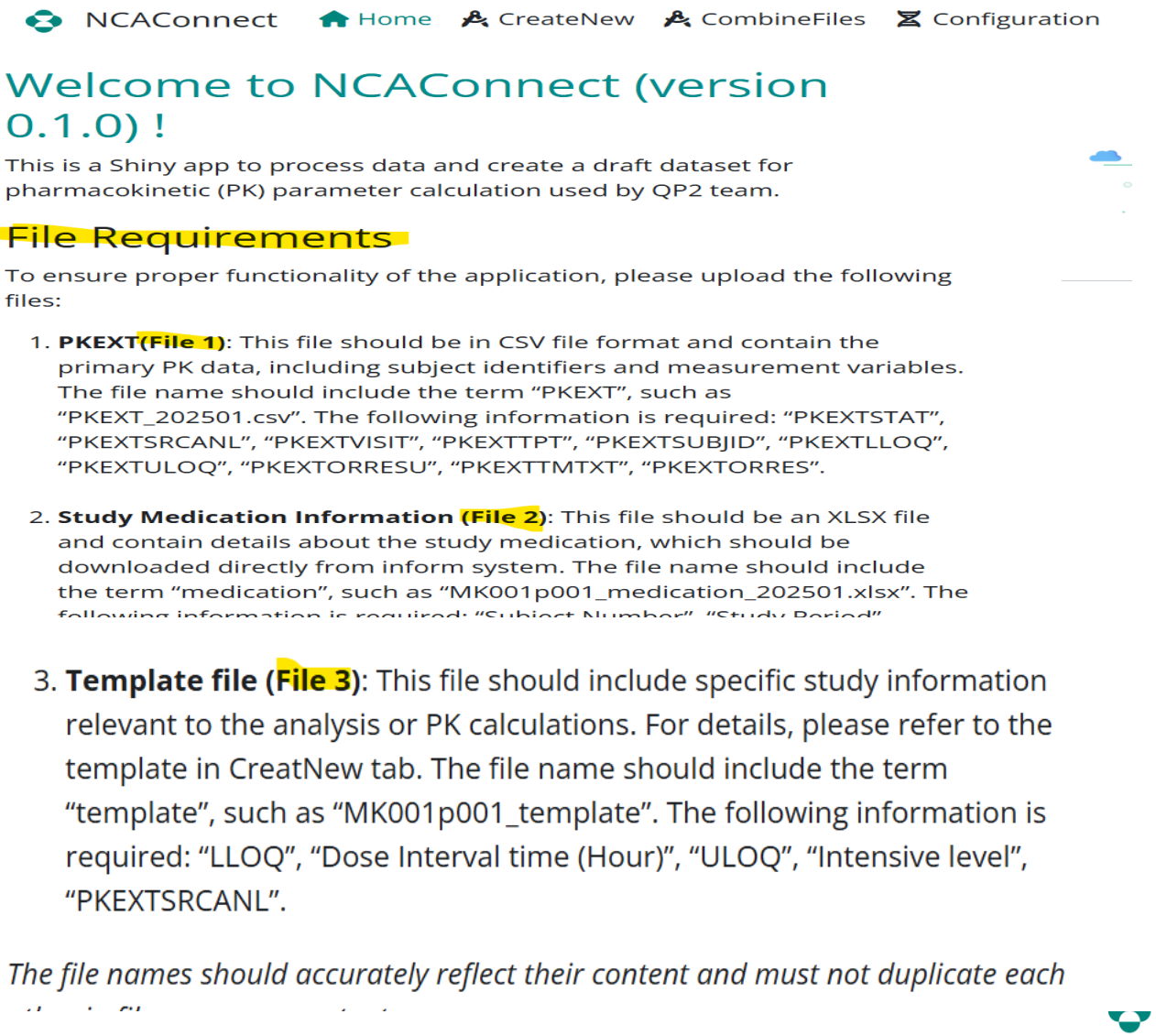
Detail - I

The homepage clearly states the format of the required files, as well as the meaning and introduction of each file.

PKEXT(File 1): PK Concentration Received from Vendor.

Study Medication Information(File 2): This file recorded study medication information and download from Inform/EDC Database.

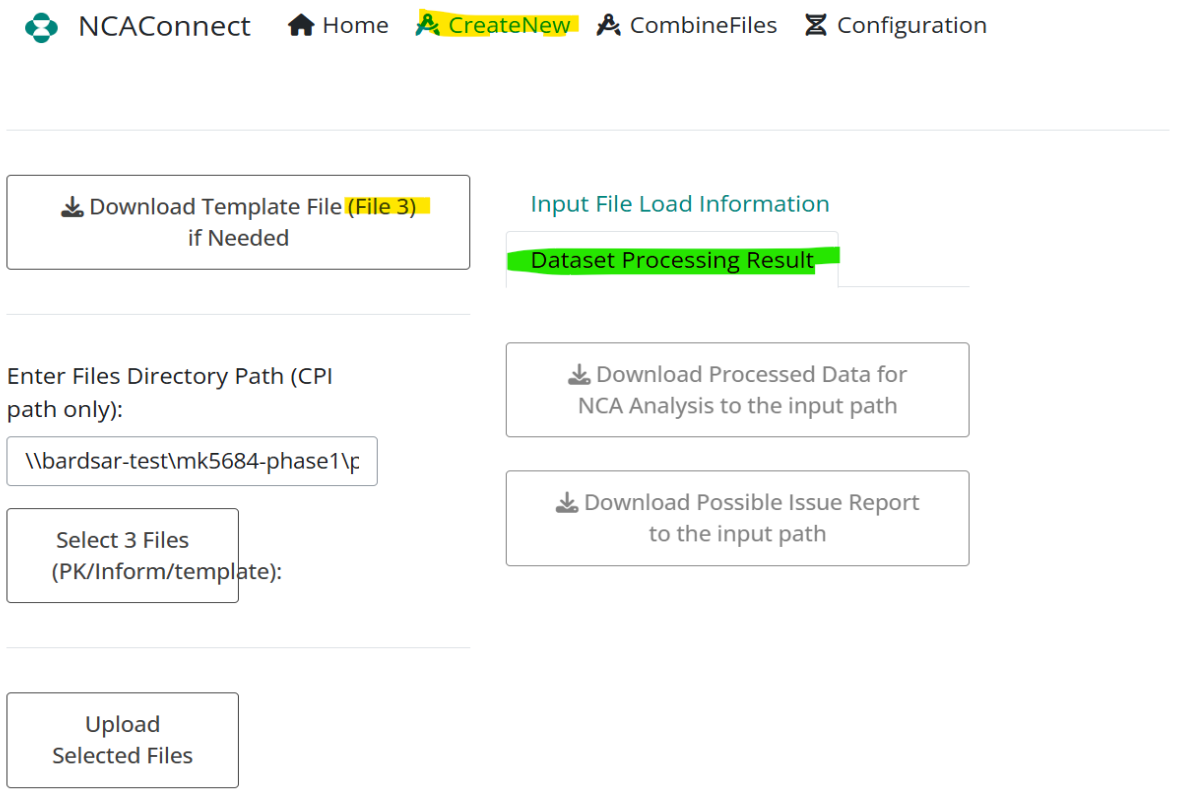
Template file(File 3): This file recorded the study specific information.



Detail - II

The first function of this app: CreateNew is to generate data for NCA.

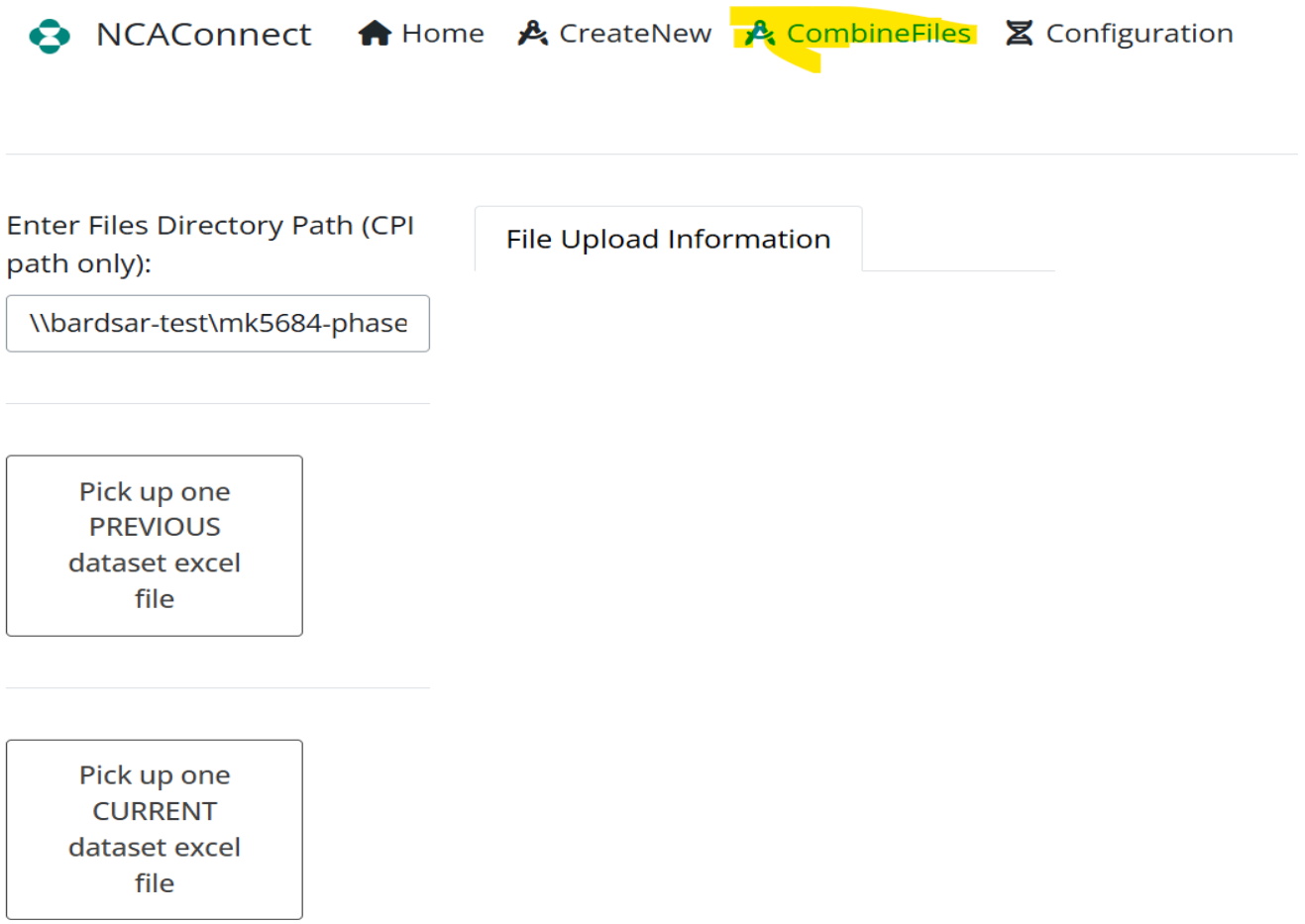
1. For a new study, We can selectively download file 3 for the first time to pharmacometrician for recording study specific information.
2. Select the file path and choose the three files that need to be uploaded.
3. We can download the processed data and issue report to the initial path by yourself.



Detail - III

The Second function of this app: CombineFiles is to compare two different files and generate combined file with difference highlighted.

1. We can upload the previous processed data and current processed data in the path.
2. Automatically download the compare file to the corresponding path.
3. The combined file includes all data from previous file with yellow highlighted, and any difference in the same cell is marked as red. Reduce repetitive work.



Detail - III

```r
# Generate new workbook
wb <- createWorkbook()


addWorksheet(wb, "Comparison")

writeData(wb, "Comparison", fileA)

# Setup background color
addStyle(wb, "Comparison", style = createStyle(fgFill = "#FFFF00"),
         rows = 1:(nrow(fileA) + 1), cols = 1:ncol(fileA), gridExpand = TRUE)

# Compare value for file A and file B
for (i in 1:ncol(fileA)) {
  for (j in 1:nrow(fileA)) {
    if (j <= nrow(fileB)) {
      valueA <- fileA[j, i]
      valueB <- fileB[j, i]
      # Check value for NA
      if (!is.na(valueA) && !is.na(valueB) && valueA != valueB) {
        addStyle(wb, "Comparison", style = createStyle(fgFill = "#FF0000"), rows = j+1, cols = i)
      }
    }
  }
}

# Get the extra lines in file B
extraRows <- fileB[-(1:nrow(fileA)), ]

# Add the extra lines into new file
if (nrow(extraRows) > 0) {
  writeData(wb, "Comparison", extraRows, startRow = nrow(fileA) + 2, colNames = FALSE)
}
```

In an R program, follow two steps to create a new workbook, read in the original file for marking, compare values row by row and column by column with the new file for marking, and then write any remaining new content into the new workbook.

The content is distinguished by different colors, with yellow marking the original content and red marking the different content of the two files, making it visually distinguishable.

DEMO

NCAConnect Home CreateNew CombineFiles Configuration

Welcome to NCAConnect (version 0.1.0) !

This is a Shiny app to process data and create a draft dataset for pharmacokinetic (PK) parameter calculation used by QP2 team.

File Requirements

To ensure proper functionality of the application, please upload the following files:

1. **PKEXT(File 1)**: This file should be in CSV file format and contain the primary PK data, including subject identifiers and measurement variables. The file name should include the term “PKEXT”, such as “PKEXT_202501.csv”. The following information is required: “PKEXTSTAT”, “PKEXTSRCANL”, “PKEXTVISIT”, “PKEXTTPT”, “PKEXTSUBJID”, “PKEXTLLOQ”, “PKEXTULOQ”, “PKEXTORRESU”, “PKEXTTMTXT”, “PKEXTORRES”.

2. **Study Medication Information (File 2)**: This file should be an XLSX file and contain details about the study medication, which should be downloaded directly from inform system. The file name should include the term “medication”, such as “MK001p001_medication_202501.xlsx”. The following information is required: “Subject Number”, “Study Period”, “Protocol Dose Level”, “Dose Amount”, “Time ~ Start Date Time”, “Time ~

Agenda

1. Introduction

- PK Data Analysis
- Dose Escalation
- Importance of PK Parameters
- Purpose for App Development

2. Traditional Method vs Shiny App

- Traditional Process
- The Shortcomings of Traditional Methods
- App Process
- The Advantages of Using App

3. App Overview and Demo

- Overview
- Demo

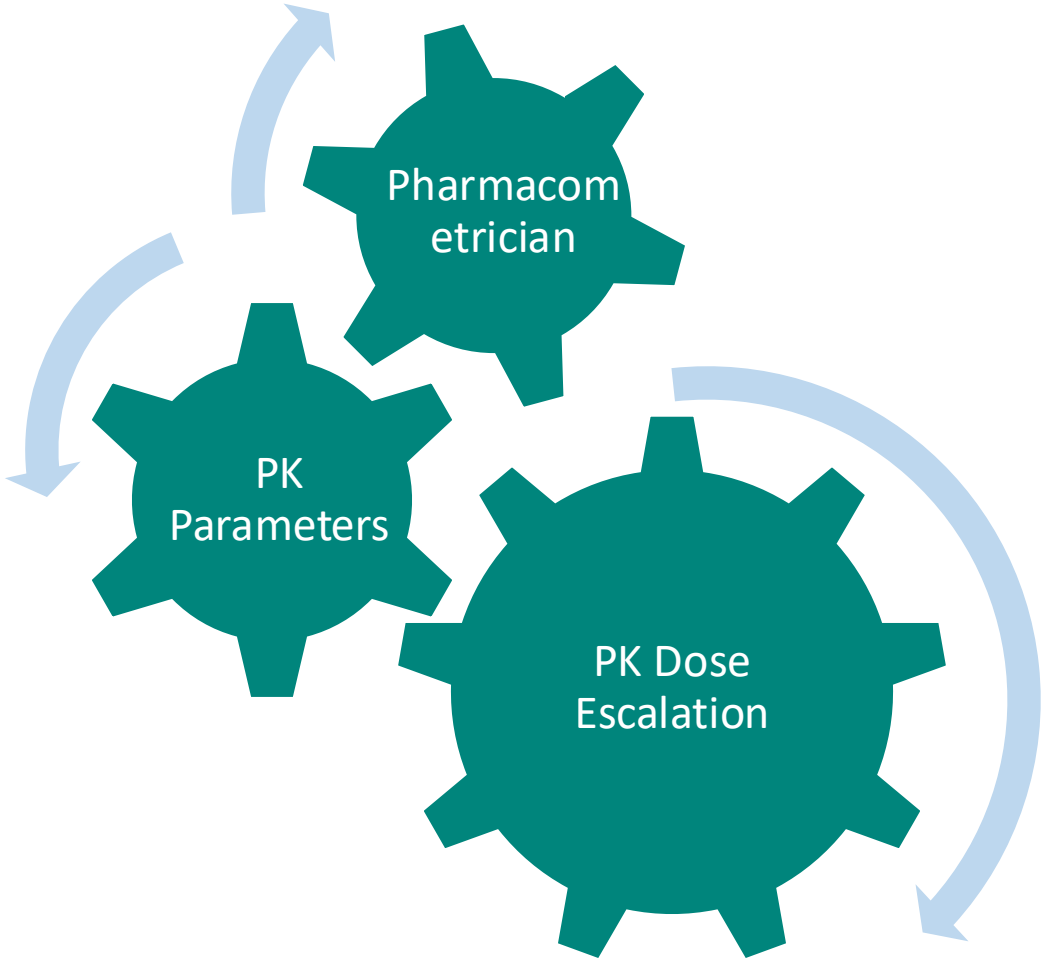
4. Conclusion

Conclusion

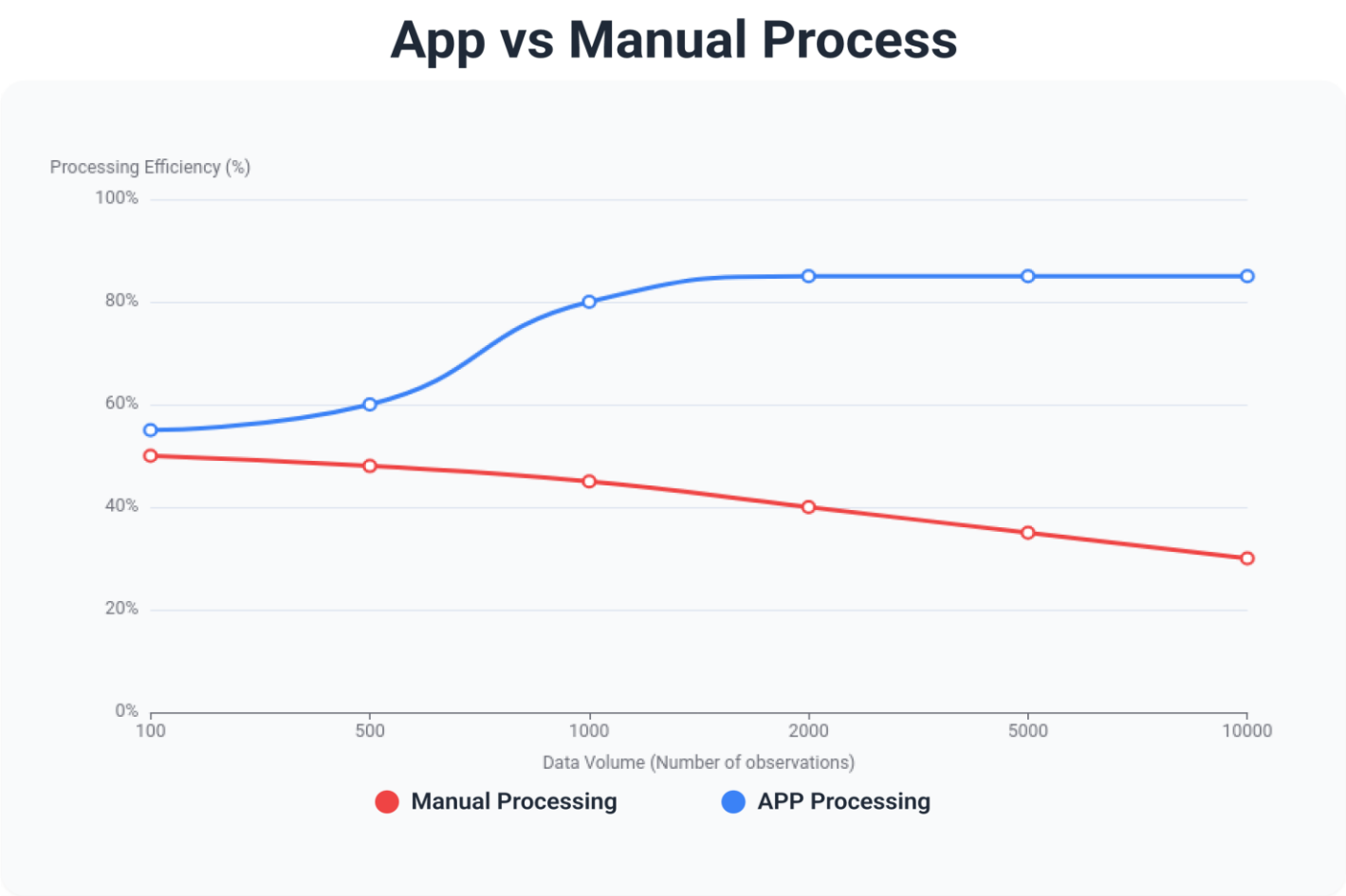
This application:

It is primarily designed for Pharmacometrician to generate the data required for calculating pharmacokinetic (PK) parameters. It accelerates the process of generating PK parameters and facilitates the collection of safety and PK data, which is essential for making dose escalation decisions.

By streamlining these processes and improved data quality contribute to a stronger basis for subsequent experimental phases, ultimately enhancing the overall research outcomes.



Conclusion



The processing efficiency is mainly reflected in the following aspects:

Reduction of Human Error: The use of an app helps to minimize the risk of errors that can occur during manual data handling, leading to more accurate results.

Decreased Workload and Time: Automated processing significantly reduces the amount of manual work required, allowing personnel to focus on more critical tasks and improving overall productivity.

Enhanced Control Over Data Validation: As the dataset grows, the ability to validate data becomes more manageable and systematic when using an app, ensuring that the integrity of the data is maintained.

Acknowledgments

I Would like to thank my team member Lily Ling for her expertise and co-authorship on this R shiny application development and presentation.

I would like to thank China Clinical Pharmacology Team in MSD for their business inputs and collaboration on application UAT and feedback.

Special thanks to MSD Statistical Programming (SP) Function Head Amy Gillespie, MSD SP A&R Group 4 Head Frédéric Coppin and my manager Wang Zhang for their guidance and supports.

Q&A

Thank you for your listening

Email: Yan.li.sun@merck.com