

OS02: Navigating and Building Bridges: A Framework for Clinical Programming using Open-Source Tools

Yash Ankurbhai Shah, Boehringer Ingelheim Pharma GmbH & Co. KG, Biberach, Germany

Kavitha Allala, Boehringer Ingelheim Pharmaceuticals, Inc., Ridgefield, USA

Disclaimer



The views and opinions expressed in the following PowerPoint slides are those of the individual presenter(s) and should not be attributed to any organization with which the presenter is employed or affiliated



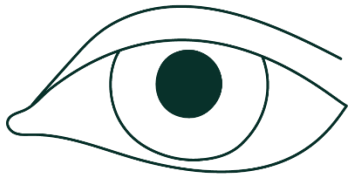
These PowerPoint slides are the intellectual property of the individual presenter(s) and are protected under the copyright laws of the United States of America and other countries. Used by permission. All rights reserved

Agenda

- 01** ▶ Overview of Open-Source Landscape
- 02** ▶ CQM (Clinical Quality Monitoring)
- 03** ▶ Challenges
- 04** ▶ Framework
- 05** ▶ Adapting framework
- 06** ▶ Conclusion
- 07** ▶ Questions??



Overview (open-source landscape)



Adoption of Open-Source Tools: Gradual adoption in the pharmaceutical industry, with increasing contributions to R packages and collaborations



Challenges in Transition: Upskilling requirement, inevitable learning curve, and lack of defined coding standards



Navigating Open-Source Options: Difficulty in direct application of tools and abundance of packages with frequent updates



Resource-Intensive Reporting: Clinical trial reporting is already demanding, making it hard to keep track of updates



Need for Defined Pathway: A stable and defined pathway is needed for programmers to focus on programming tables in R, using familiar tools

What is CQM

(Clinical Quality Monitoring)?

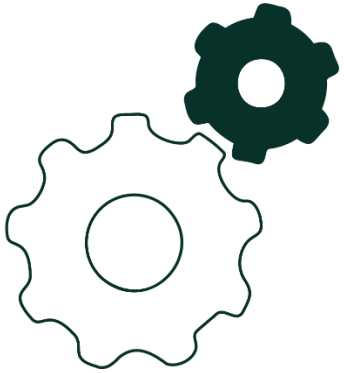
- The CQM process is a quarterly cross-functional review of aggregated clinical data (Tables, Listings, Figures) to ensure data integrity and accuracy.
- Trial team members, including clinical leaders, study physicians, statisticians, data managers, programmers, and safety physicians, participate in this process.

Why CQM?

- The CQM outputs are generated in SAS® as PDF files, which are typically thousands of pages long. For instance, in a trial with 100 treated patients, there are 1000 pages to review for the 1st CQM.
- The PDF files are reviewed in quarterly meetings that last approximately 1-2 hours.
- BDS(Biostatistics and Data Sciences) colleagues will need to query the data and respond via email for any questions on these outputs. Example questions include:
 - Can you give me a list of patients who have <specify criteria>?
 - Can you filter the data using a different criteria?
 - Can you sort the data in a different order?
 - Can you arrange the columns in a different way?
 - Can you send me the list in an Excel file?



Challenges



Time consuming review of the PDF static outputs



Dependency between the functions for desired outputs



Time lapse between queries and resolution during review process

Framework

Workflow

Programmer develops tables

```
# Table ADSL Summary
```

```
# Create summary table object
table1 <- tplyr_table(adsl, ARM) %>%
  add_layer(
    group_desc(AGE, by = "Age (years)")
  ) %>%
  add_layer(
    group_count(EOSSTT, by = "End of Study status (%)")
  ) %>%
  add_layer(
    group_count(vars(DCDECOD, DCSREAS))
  )
```

```
# Once the object is created, build the table.
```

```
results1 <- build(table1, metadata = TRUE) %>%
  dplyr::mutate(row_label2 = ifelse(row_label2 == row_label1, "Total (%)",
                                   row_label2
  )) %>%
  apply_row_masks(row_breaks = TRUE)
```

```
# Table ADAE
```

```
tab2 <- tplyr_table(adae, TRTA) %>%
  set_pop_data(adsl) %>%
  set_pop_treat_var(TRT01A) %>%
  add_layer(
    group_count("All subject") %>%
    set_distinct_by(USUBJID) %>%
    set_format_strings(f_str(
      "xx",
      distinct_total
    ))
  ) %>%
  add_layer(
    group_count("Subjects with adverse events") %>%
    set_distinct_by(USUBJID) %>%
    set_format_strings(f_str(
      "xx (xx %)",
      distinct_n, distinct_pct
    ))
  ) %>%
  add_layer(
    group_count(AESEV, by = "Adverse event severity") %>%
    set_distinct_by(USUBJID) %>%
    set_format_strings(f_str(
      "xx (xx %)",
      distinct_n, distinct_pct
    ))
  ) %>%
  add_layer(
    group_count("Subjects with severe AE", where = AESER == "Y") %>%
    set_distinct_by(USUBJID) %>%
    set_format_strings(f_str(
      "xx (xx %)",
      distinct_n, distinct_pct
    ))
  )
results2 <- build(tab2, metadata = TRUE) %>% apply_row_masks(row_breaks = TRUE)
```

Workflow (cont'd)

Using framework, put them into desired output

```
# Define server logic
server <- function(input, output) {

  customizedlistingServer("listing1", mock_adsl, "listthree.csv", showAllcols = TRUE)
  reactableServer("res1", results1, table1, "adslownloadfile.csv", showAllcols = TRUE)
  reactableServer("res2", results2, tab2, "adaedownloadfile.csv", showAllcols = list("STUDYID"),
    minimumcolwidth = 50)
  reactableServer("res3", results1, table1, "adslownloadfile.csv", showAllcols = list("AGE", "EOSSTT"),
    minimumcolwidth = 1)

}

# Run the application
shinyApp(ui = ui, server = server)
```

```
# Define UI for application
ui <- fluidPage(

  # Application title
  titlePanel("demo CQM app"),

  # Main panel with tabs
  mainPanel(
    tabsetPanel(

      tabPanel("Demographics",
        reactableUI("res1")),
      tabPanel("Adverse Events",
        reactableUI("res2")),
      tabPanel("Demographics",
        reactableUI("res3")),
      tabPanel("Demographics listing",
        customizedlistingUI("listing1")),

    )
  )
)
```

Workflow (cont'd)

See the end results and share with their end users

		Placebo	Xanomeline High Dose	Xanomeline Low Dose
All subject		86	84	84
Subjects with adverse events		32 (37 %)	46 (55 %)	50 (60 %)
Adverse event severity	MILD	31 (36 %)	41 (49 %)	37 (44 %)
	MODERATE	7 (8 %)	24 (29 %)	26 (31 %)
	SEVERE	0 (0 %)	2 (2 %)	7 (8 %)
Subjects with severe AE		0 (0 %)	0 (0 %)	0 (0 %)
Select columns to display:				
TRTAN [Actual Treatment (N)], AGEGR1 [I ▲]				
TRTAN Actual Treatment (N) ↑	AGEGR1 Pooled Age Group 1 ↑	AESEV Severity/Intensity ↑		
81	65-80	MILD		
81	65-80	MILD		
81	65-80	MILD		

Workflow: Summary

Programmer develops
tables



Using framework, put them into
desired output



See the end results and share
with their end users

```
# Table ADSL Summary
# Create summary table object
table1 <- tplyr_table(adsl, ARM) %>%
  add_layer{
    group_desc(AGE, by = "Age (years)")
  } %>%
  add_layer{
    group_count(EOSSTT, by = "End of Study status (%)")
  } %>%
  add_layer{
    group_count(vars(DCDECOD, DCSREAS))
  }
# Once the object is created, build the table.
results1 <- build(table1, metadata = TRUE) %>%
  dplyr::mutate(row_label2 = ifelse(row_label2 == row_label1, "Total (%)",
    row_label2
  )) %>%
  apply_row_masks(row_breaks = TRUE)

# Table ADAE
tab2 <- tplyr_table(adsl, TRTA) %>%
  set_pop_data(adsl) %>%
  set_pop_treat_var(TRT01A) %>%
  add_layer{
    group_count("All subject") %>%
    set_distinct_by(USUBJID) %>%
    set_format_strings(f_str{
      "xx (xx %)",
      distinct_n, distinct_pct
    })
  } %>%
  add_layer{
    group_count("Subjects with adverse events") %>%
    set_distinct_by(USUBJID) %>%
    set_format_strings(f_str{
      "xx (xx %)",
      distinct_n, distinct_pct
    })
  } %>%
  add_layer{
    group_count(AESEV, by = "Adverse event severity") %>%
    set_distinct_by(USUBJID) %>%
    set_format_strings(f_str{
      "xx (xx %)",
      distinct_n, distinct_pct
    })
  } %>%
  add_layer{
    group_count("Subjects with severe AE", where = AESER == "Y") %>%
    set_distinct_by(USUBJID) %>%
    set_format_strings(f_str{
      "xx (xx %)",
      distinct_n, distinct_pct
    })
  }
results2 <- build(tab2, metadata = TRUE) %>% apply_row_masks(row_breaks = TRUE)
```

```
# Define UI for application
ui <- fluidPage(

  # Application title
  titlePanel("demo CQM app"),

  # Main panel with tabs
  mainPanel(
    tabsetPanel(

      tabPanel("Demographics",
        reactableUI("res1")),
      tabPanel("Adverse Events",
        reactableUI("res2")),
      tabPanel("Demographics",
        reactableUI("res3")),
      tabPanel("Demographics listing",
        customizedlistingUI("listing1")),

    )
  )

# Define server logic
server <- function(input, output) {

  customizedlistingServer("listing1", mock_adsl, "listthree.csv", showAllcols = TRUE)
  reactableServer("res1", results1, table1, "adsldownloadfile.csv", showAllcols = TRUE)
  reactableServer("res2", results2, tab2, "adaedownloadfile.csv", showAllcols = list("STUDYID"),
    minimumcolwidth = 50)
  reactableServer("res3", results1, table1, "adsldownloadfile.csv", showAllcols = list("AGE", "EOSSTT"),
    minimumcolwidth = 1)

}

# Run the application
shinyApp(ui = ui, server = server)
```

	Placebo	Xanomeline High Dose	Xanomeline Low Dose
All Subject	86	84	84
Subjects with adverse events	32 (37 %)	46 (55 %)	50 (60 %)
Adverse event severity			
MILD	31 (36 %)	41 (49 %)	37 (44 %)
MODERATE	7 (8 %)	24 (29 %)	26 (31 %)
SEVERE	0 (0 %)	2 (2 %)	7 (8 %)
Subjects with severe AE	0 (0 %)	0 (0 %)	0 (0 %)

Select columns to display:

TRTAN [Actual Treatment (N)]. AGEGR1 [I ←

TRTAN Actual Treatment (N)	AGEGR1 Pooled Age Group 1	AESV Severity/Intensity
81	65-80	MILD
81	65-80	MILD
81	65-80	MILD

Framework Features

		Placebo	Xanomeline High Dose	Xanomeline Low Dose
Age (years)	n	86	84	84
	Mean (SD)	75.2 (8.59)	74.4 (7.89)	75.7 (8.29)
	Median	76.0	76.0	77.5
	Q1, Q3	69.2, 81.8	70.8, 80.0	71.0, 82.0
	Min, Max	52, 89	56, 88	51, 88
	Missing	0	0	0
End of Study status (%)	COMPLETED	58 (67.4%)	27 (32.1%)	25 (29.8%)
	DISCONTINUED	28 (32.6%)	57 (67.9%)	59 (70.2%)
ADVERSE EVENT	Total (%)	8 (9.3%)	40 (47.6%)	44 (52.4%)
1–11 of 30 rows		Previous 1 2 3 Next		
		Placebo	Xanomeline High Dose	Xanomeline Low Dose
All subject		86	84	84
Subjects with adverse events		32 (37 %)	46 (55 %)	50 (60 %)
Adverse event severity	MILD	31 (36 %)	41 (49 %)	37 (44 %)
	MODERATE	7 (8 %)	24 (29 %)	26 (31 %)
	SEVERE	0 (0 %)	2 (2 %)	7 (8 %)
Subjects with severe AE		0 (0 %)	0 (0 %)	0 (0 %)

Different Outputs

Customized shiny app

Table 5.1 Disposition of Patients - Screened Set

Table 7.2 Last digit preference: weight by site - Treated set

Not Treated				9 (34.6%)
Treated				17 (65.4%)
Permanently discontinued trial medication	6 (100.0%)	9 (100.0%)	1 (33.3%)	16 (94.1%)
Completed planned treatment period	6 (100.0%)	8 (88.9%)	0 (0.0%)	14 (82.4%)
Withdrawal By Subject	0 (0.0%)	1 (11.1%)	0 (0.0%)	1 (5.9%)
Other	0 (0.0%)	0 (0.0%)	1 (33.3%)	1 (5.9%)
Permanently discontinued from the trial	6 (100.0%)	9 (100.0%)	0 (0.0%)	15 (88.2%)
Completed planned observation time	6 (100.0%)	8 (88.9%)	0 (0.0%)	14 (82.4%)
Withdrawal By Subject	0 (0.0%)	1 (11.1%)	0 (0.0%)	1 (5.9%)
On treatment at time of reporting	0 (0.0%)	0 (0.0%)	0 (0.0%)	0 (0.0%)

1-11 of 12 rows Show 10 ▾

Previous 1 2 Next

Select columns to display:

DCTREAS [No Label], DCSREAS [No Label] ▾

DCTREAS ▴ ▾ No Label	DCSREAS ▴ ▾ No Label	TRTFL ▴ ▾ No Label	ENRFL ▴ ▾ No Label	ONTRTREP ▴ ▾ No Label	INTRLREP ▴ ▾ No Label
Completed planned treatment period	Completed planned observation time	Y	Y	N	N

Different Outputs (cont'd)

Quarto

Adverse events			Table of contents	
	Placebo	Xanomeline High Dose	DEMOGRAPHICS	Xanomeline Low Dose
	86	84	Adverse events	
			Stand alone listing	84
	32 (37 %)	46 (55 %)		50 (60 %)
MILD	31 (36 %)	41 (49 %)		37 (44 %)
MODERATE	7 (8 %)	24 (29 %)		26 (31 %)
SEVERE	0 (0 %)	2 (2 %)		7 (8 %)
	0 (0 %)	0 (0 %)		0 (0 %)
SITEID				
Study Site Identifier				
		701		
		701		
		701		

Different Outputs (cont'd)

Internal custom framework

×

total entries selected

ive Filters

Clear All Filters

CQM Module

Lab Line Plot

EDISH

DV Table

Demographics

Adverse events

		Placebo
Age (years)	n	86
	Mean (SD)	75.2 (8.59)
	Median	76.0
	Q1, Q3	69.2, 81.8
	Min, Max	52, 89
	Missing	0
Gender Distribution	F	53 (61.6%)
	M	33 (38.4%)
F	Total (%)	53 (61.6%)

1–11 of 21 rows Show 10

Select columns to display:

AGE [Age], SEX [Sex], RACE [Race]

AGE :
Age

SEX :
Sex

63	F
64	M
85	F
52	M

Adapting Framework



R Learning

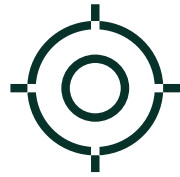


**Challenges with
transition from
templates to R**



**Stakeholder
alignment**

Conclusion



This initiative aims to progress CQM to make this more efficient



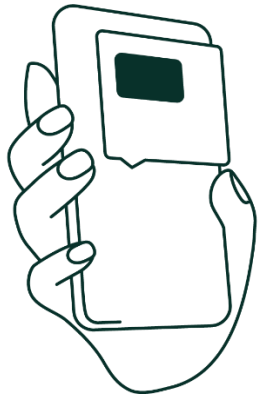
The challenges we have encountered have provided valuable guidance in fine-tuning our future path

**Thank you
for your attention!**

**Any questions are
welcome.**



Contact details



Yash Ankurbhai Shah

yash_ankurbhai.shah@boehringer-ingelheim.com



Kavitha Allala

kavitha.allala@boehringer-ingelheim.com