

HUD_WORKFLOW_DEMO

January 30, 2026 at 07:16 PM

HUD System Workflow Demo

This document demonstrates the full HUD (Heads-Up Display) system in zzvim-R, showcasing RStudio-inspired workspace visibility tools for terminal-based R development.

Prerequisites

- Kitty terminal (or Ghostty/WezTerm)
- R with `jsonlite` package installed
- zzvim-R plugin loaded

Part 1: Setting Up the Demo Environment

1.1 Create a Demo R Script

Create a file `hud_demo.R`:

```
# =====
# HUD System Demo Script
# =====
# This script creates various R objects to demonstrate the HUD features

# Load packages
library(dplyr)
library(ggplot2)

# -----
# Create diverse workspace objects for Memory HUD
# -----


# Small objects
x <- 1:100
y <- rnorm(100)
greeting <- "Hello, HUD!"

# Medium objects
mtcars_copy <- mtcars
```

```

iris_subset <- iris |> filter(Species == "setosa")

# Large object
big_matrix <- matrix(rnorm(1e6), nrow = 1000, ncol = 1000)

# List object
config <- list(
  name = "demo",
  settings = list(a = 1, b = 2, c = 3),
  data = head(mtcars)
)

# -----
# Create data frames for DataFrames HUD
# -----

# Patient data
patients <- data.frame(
  id = 1:50,
  age = sample(20:80, 50, replace = TRUE),
  treatment = sample(c("A", "B", "Control"), 50, replace = TRUE),
  outcome = rnorm(50, mean = 100, sd = 15)
)

# Time series data
daily_sales <- data.frame(
  date = seq.Date(as.Date("2024-01-01"), by = "day", length.out = 365),
  revenue = cumsum(rnorm(365, mean = 1000, sd = 200)),
  units = sample(50:150, 365, replace = TRUE)
)

# Summary statistics
summary_stats <- patients |>
  group_by(treatment) |>
  summarise(
    n = n(),
    mean_age = mean(age),
    mean_outcome = mean(outcome),
    .groups = "drop"
  )

# -----
# Create plots for Plot HUD
# -----

```

```

# Plot 1: Scatter plot
zzplot(mtcars$wt, mtcars$mpg,
       main = "Weight vs MPG",
       xlab = "Weight (1000 lbs)",
       ylab = "Miles per Gallon",
       pch = 19, col = "steelblue",
       .name = "scatter_wt_mpg")

# Plot 2: Histogram
zzplot(hist(patients$age, breaks = 15,
            main = "Patient Age Distribution",
            xlab = "Age", col = "coral"),
       .name = "histogram_age")

# Plot 3: Box plot
zzplot(boxplot(outcome ~ treatment, data = patients,
               main = "Outcome by Treatment",
               col = c("lightblue", "lightgreen", "lightyellow")),
       .name = "boxplot_treatment")

# Plot 4: ggplot scatter with regression
p1 <- ggplot(mtcars, aes(x = hp, y = mpg, color = factor(cyl))) +
  geom_point(size = 3) +
  geom_smooth(method = "lm", se = FALSE) +
  labs(title = "Horsepower vs MPG by Cylinder",
       x = "Horsepower", y = "MPG", color = "Cylinders") +
  theme_minimal()
zzgplot(p1, .name = "ggplot_hp_mpg")

# Plot 5: Time series
p2 <- ggplot(daily_sales, aes(x = date, y = revenue)) +
  geom_line(color = "darkgreen") +
  geom_smooth(method = "loess", color = "red", se = TRUE) +
  labs(title = "Daily Revenue Trend",
       x = "Date", y = "Cumulative Revenue ($)") +
  theme_minimal()
zzgplot(p2, .name = "timeseries_revenue")

# Plot 6: Faceted plot
p3 <- ggplot(iris, aes(x = Sepal.Length, y = Sepal.Width)) +
  geom_point(aes(color = Species)) +
  facet_wrap(~Species) +
  labs(title = "Iris Sepal Dimensions") +
  theme_bw()
zzgplot(p3, .name = "faceted_iris")

```

```
cat("\n✓ Demo environment ready!\n")
cat(" Objects: ", length(ls()), "\n")
cat(" Data frames: patients, daily_sales, summary_stats\n")
cat(" Plots: 6 plots in history\n")
```

1.2 Start R Session

In Vim, open `hud_demo.R` and start R:

```
<LocalLeader>r      " Start Docker R (or <LocalLeader>rr for host R)
```

1.3 Source the Demo Script

```
<CR>          " On first line, or select all and <CR>
```

Wait for the script to complete. You should see plots appearing in the kitty pane as they are created.

Part 2: Individual HUD Demonstrations

2.1 Memory HUD (<LocalLeader>m)

Press `<LocalLeader>m` to see memory usage of all workspace objects:

```
== Memory Usage ==
big_matrix     : 7.63 MB
daily_sales    : 0.01 MB
patients       : 0.00 MB
config         : 0.00 MB
...
TOTAL          : 7.68 MB
=====
```

Key insight: Immediately identify memory-heavy objects (`big_matrix`).

2.2 Data Frames HUD (<LocalLeader>e)

Press `<LocalLeader>e` to see all data frames:

```
== Data Frames ==
patients       : 50 rows x 4 cols
daily_sales    : 365 rows x 3 cols
summary_stats  : 3 rows x 4 cols
mtcars_copy    : 32 rows x 11 cols
iris_subset    : 50 rows x 5 cols
=====

```

Key insight: Quick overview of dataset dimensions without typing `dim()`.

2.3 Packages HUD (<LocalLeader>z)

Press <LocalLeader>z to see loaded packages:

```
==== Package Status ===
```

```
Loaded packages:
```

```
ggplot2
```

```
dplyr
```

```
stats
```

```
graphics
```

```
...
```

```
Total loaded: 12 packages
```

Key insight: Verify required packages are loaded.

2.4 Data Viewer (<LocalLeader>v)

Position cursor on patients and press <LocalLeader>v:

```
id  age  treatment  outcome
1   45    A          102.3
2   67    B          98.7
3   34  Control     115.2
...

```

Key insight: RStudio-style data inspection without leaving Vim.

2.5 Environment HUD (<LocalLeader>x)

Press <LocalLeader>x to see environment variables:

Variable	Value
R_HOME	/usr/lib/R
R_LIBS_USER	~/R/x86_64-pc-linux-gnu-library/4.4
PATH	/usr/local/bin:/usr/bin:...
...	

Key insight: Debug R configuration and path issues.

2.6 Options HUD (<LocalLeader>a)

Press <LocalLeader>a to see R session options:

Option	Value
digits	7
warn	0
width	80
stringsAsFactors	FALSE
...	

Key insight: Verify session configuration affecting output.

Part 3: Plot HUD Demonstration

3.1 Open Plot HUD (<LocalLeader>P)

Press <LocalLeader>P to open the Plot HUD:

```
Plot History [HUD]
=====
Enter=display | z=zoom PDF | s=save | d=delete | q=close | /=search

#   Name           Created          Code
-----
> [1] scatter_wt_mpg    2026-01-30T14:30  zzplot(mtcars$wt, mt...
[2] histogram_age       2026-01-30T14:30  zzplot(hist(patients...
[3] boxplot_treatment   2026-01-30T14:30  zzplot(boxplot(outco...
[4] ggplot_hp_mpg        2026-01-30T14:31  zzggplot(p1, .name =...
[5] timeseries_revenue   2026-01-30T14:31  zzggplot(p2, .name =...
[6] faceted_iris         2026-01-30T14:31  zzggplot(p3, .name =...

Total: 6 plots | Current: 6
```

3.2 Navigate and Display Plots

```
j/k      " Move cursor up/down
3        " Quick jump to plot 3 (boxplot)
Enter    " Display selected plot in kitty pane
```

The kitty pane updates to show the selected plot.

3.3 Zoom Plot (Open PDF)

```
z      " Open PDF of selected plot in Preview
```

The PDF opens in your system viewer - vector graphics, infinite zoom, publication-ready.

3.4 Save Plot

```
s      " Prompt appears: "Save as: /path/to/plot.pdf"
```

Type filename and press Enter. Supports .pdf and .png.

3.5 Delete Plot

```
d      " Prompt: Delete "boxplot_treatment"? (y/n)
y      " Confirm deletion
```

Plot is removed from history, HUD refreshes.

3.6 Search Plots

```
/revenue      " Standard Vim search  
n            " Next match
```

Find plots by name or code snippet.

3.7 Quick Navigation (Outside HUD)

From any R file:

```
<LocalLeader><      " Previous plot  
<LocalLeader>>      " Next plot  
<LocalLeader>]      " Zoom current plot (open PDF)
```

Part 4: HUD Dashboard

4.1 Open Full Dashboard (<LocalLeader>0)

Press <LocalLeader>0 to open all 6 HUDs in tabs:

[Memory] [DataFrames] [Packages] [Environment] [Options] [Plots]

4.2 Navigate Tabs

```
gt          " Next tab  
gT          " Previous tab  
1gt         " Go to tab 1 (Memory)  
6gt         " Go to tab 6 (Plots)
```

4.3 Refresh Dashboard

```
<LocalLeader>0      " Refresh all HUD tabs with current data
```

4.4 Close Dashboard

```
q          " Close current HUD tab  
:tabclose  " Close current tab  
:tabonly   " Close all tabs except current
```

Part 5: Integrated Workflow Example

Scenario: Exploratory Data Analysis

1. Start R session
<LocalLeader>r
2. Load data and create initial plots
source("analysis.R")

```

3. Check workspace state
<LocalLeader>0      " Open dashboard
gt                  " Navigate to DataFrames tab
                     " Verify all datasets loaded correctly

4. Check memory usage
gT gT gT gT          " Back to Memory tab
                     " Identify any unexpectedly large objects

5. Review plots
6gt                  " Go to Plots tab
j j Enter             " Select and display a specific plot
z                   " Zoom to inspect details

6. Export final plot
s                  " Save as publication-ready PDF

7. Continue analysis
q                  " Close HUD
                     " Back to editing code

```

Key Mappings Summary

Key	Context	Action
<LocalLeader>m	R file	Memory HUD (inline)
<LocalLeader>e	R file	Data Frames HUD (inline)
<LocalLeader>z	R file	Packages HUD (inline)
<LocalLeader>v	R file	Data Viewer (on word under cursor)
<LocalLeader>x	R file	Environment HUD (split)
<LocalLeader>a	R file	Options HUD (split)
<LocalLeader>P	R file	Plot HUD (split)
<LocalLeader>0	R file	Full Dashboard (6 tabs)
<LocalLeader>]	R file	Zoom current plot
<LocalLeader><	R file	Previous plot
<LocalLeader>>	R file	Next plot
Enter	HUD buffer	Select/inspect item
z	Plot HUD	Zoom (open PDF)
s	Plot HUD	Save plot
d	Plot HUD	Delete plot
r	Plot HUD	Refresh
q	Any HUD	Close

Comparison with RStudio

RStudio Pane	zzvim-R Equivalent	Key
Environment	Memory HUD + DataFrames HUD	<LocalLeader>m, <Local- Leader>e
Plots	Plot HUD + Kitty pane	<LocalLeader>p
Packages	Packages HUD	<LocalLeader>z
Files	(use Vim file navigation)	:e, :Ex
Help	K on function name	K
Viewer	Data Viewer	<LocalLeader>v
Console	Vim terminal	<LocalLeader>r

The HUD system brings RStudio's workspace visibility to terminal Vim while maintaining keyboard-driven efficiency.