# DSS Prototype Analysis

Alvin Murphy

## Table of contents

# 1 Hypothesis

Hypotheses are "innocent until proven guilty." We'll assume that SpaceX and others have proven that DevSecOps tech can meet hard-real-time requirements but nothing available in the body of knowledge documents this.

**Hypothesis:** Modern DevSecOps architectures can be designed to meet hard-real-time latency ($\mu$) requirements using modern computing environments and computing infrastructure.

$H_0 : \mu \leq 500ms$ with jitter within latency bounds
$H_a : \mu > 500ms$ with jitter exceeding latency bounds

*Murphy, Alvin C. and Moreland Jr, James D. 'Integrating AI Microservices into Hard-Real-Time SoS to Ensure Trustworthiness of Digital Enterprise Using Mission Engineering'. 1 Jan. 2021 : 38 – 54.*

```
setwd('/home/jovyan/work/data')
```

# 2 Load Data Files

```
macData <- read.csv('DSS_SpanData-mac-2022-05-02 18_38_26_s10-5-1.csv', header = TRUE)
linpcData <- read.csv('DSS_SpanData-linuxpc-2022-06-06 17_38_29_s10-5-1.csv', header = TRU
rpi4Data <- read.csv('DSS_SpanData-rpi4-2022-06-06 17_52_59_s10-5-1.csv', header = TRUE)
awsEC2Data <- read.csv('DSS_SpanData-aws_ec2-2022-06-07 17_44_08_s10-5-1.csv', header = TR
cci_Data <- read.csv('DSS_SpanData-odu_cci-2022-06-28 17_47_20_s10-5-1.csv', header = TRUE
```

## 2.1 Review and Tag MacBook Air (2017) Data

```
summary(macData)
```

```
   Trace.ID           Trace.name          Start.time          Duration
 Length:100          Length:100          Length:100          Length:100
 Class :character    Class :character    Class :character    Class :character
 Mode  :character    Mode  :character    Mode  :character    Mode  :character
```

```
head(macData[, c(1,2)])
head(macData[, c(3,4)])
```

A data.frame: $6 \times 2$

| | Trace.ID <chr> | Trace.name <chr> |
|---|---|---|
| 1 | 9ee3577fb1b427bc4fc17fecc5154d7d | dss-prototype: /TE |
| 2 | f05ddc4dc13aff5c3098011b2a402401 | dss-prototype: /tracks |
| 3 | 2bd901fbbfc9ee8dfa7c9629d93a1567 | dss-prototype: /IAD |
| 4 | 69a48381a14e79da08aaa2353f7db4b2 | dss-prototype: /RIC |
| 5 | e83037dcb9438c04dc12fba373b5502f | dss-prototype: /WA |
| 6 | 7e381cd880adb670bb9627ca47020938 | dss-prototype: /TE |

A data.frame: $6 \times 2$

| | Start.time <chr> | Duration <chr> |
|---|---|---|
| 1 | 2022-05-02 10:25:01.366 | 36.0 ms |
| 2 | 2022-05-02 10:25:00.309 | 43.3 ms |
| 3 | 2022-05-02 10:24:58.818 | 464 ms |
| 4 | 2022-05-02 10:24:57.307 | 494 ms |
| 5 | 2022-05-02 10:24:56.128 | 139 ms |
| 6 | 2022-05-02 10:24:55.081 | 30.3 ms |

### 2.1.1 Add Source Indicator to MacBook Data

```
macDataPlat <- macData

macDataPlat$platform = "2017-macbook"
macDataPlat$env = 0
```

## 2.2 Tag Linux PC (2012) Data

```
linpcDataPlat <- linpcData

linpcDataPlat$platform = "2012-linpc"
linpcDataPlat$env = 1
```

## 2.3 Tag Raspberry Pi 4 (2020) Data

```
rpi4DataPlat <- linpcData

rpi4DataPlat$platform = "2020-rpi4"
rpi4DataPlat$env = 2
```

## 2.4 Tag AWS EC2 t2.micro (2022) Data

```
awsEC2DataPlat <- awsEC2Data

awsEC2DataPlat$platform = "2022-aws-ec2"
awsEC2DataPlat$env = 3
```

## 2.5 Tag ODU CCI (2022) Data

```
cciDataPlat <- cci_Data

cciDataPlat$platform = "2022-odu-cci"
cciDataPlat$env = 4
```

## 2.6 Merge Data Files

Here we merge data from all platforms.

```
spanData = rbind(macDataPlat, linpcDataPlat, rpi4DataPlat,
                 awsEC2DataPlat, cciDataPlat)

# Mclust components
    # cci = 1
    # mac = 9
    # linpc = 1
    # rpi4 = 1
    # awsEC2 = 9

# summary(spanData)
# head(spanData[, c(1,2,3)])
```

```
# head(spanData[, c(4,5,6)])
# spanData
```

## 3 Convert Data into Useable Metrics

To make the data more usable and easier to understand we apply conversions from text to numeric and add additional columns with supporting information. A **useCase** column is added to identify specific DSS request use cases; e.g. Get Dulles Airport Data. The data also indicates whether the request is managed internally or a connection to an external service is required to provided a response (i.e., https://opensky-network.org). A **numContainers** column is added to indicate the number of containers involved in providing a use case response (e.g. independent variable). An **ext** column is added to indicate whether an API external to the Docker environment is used; e.g., ext = TRUE for OpenSky API calls.

```
## Dictionary for converting data

DSSoperations <- c(
    "dss-prototype: /IAD" = "Get Dulles Airport Data (External)",
    "dss-prototype: /RIC" = "Get Richmond Airport Data (External)",
    "dss-prototype: /tracks" = "Get Stored Local DSS Tracks (Internal)",
    "dss-prototype: /TE" = "Trial Engage (Internal)",
    "dss-prototype: /WA" = "Assess Weapons (Internal)"
)

DSSuseCaseNum <- c(
    "dss-prototype: /IAD" = 4,
    "dss-prototype: /RIC" = 5,
    "dss-prototype: /tracks" = 1,
    "dss-prototype: /TE" = 2,
    "dss-prototype: /WA" = 3
)

DSSexternal <- c(
    "dss-prototype: /IAD" = TRUE,
    "dss-prototype: /RIC" = TRUE,
    "dss-prototype: /tracks" = FALSE,
    "dss-prototype: /TE" = FALSE,
    "dss-prototype: /WA" = FALSE
)
```

```r
DSStraceShortName <- c(
    "dss-prototype: /IAD" = "/IAD",
    "dss-prototype: /RIC" = "/RIC",
    "dss-prototype: /tracks" = "/tracks",
    "dss-prototype: /TE" = "/TE",
    "dss-prototype: /WA" = "/WA"
)

# DSShypothesis <- c(
#     Duration > 500 = FALSE
#     Duration <= 500 = TRUE
# )
```

## 3.1 Add Additional Column Descriptors

```r
spanMetrics <- spanData
```

```r
spanMetrics$useCase <- DSSoperations[spanMetrics$Trace.name]
spanMetrics$useCaseNum <- DSSuseCaseNum[spanMetrics$Trace.name]

spanMetrics$ext = DSSexternal[spanMetrics$Trace.name]
spanMetrics$Trace.name = DSStraceShortName[spanMetrics$Trace.name]
```

```r
# install.packages("tidyverse")
library(tidyverse)
```

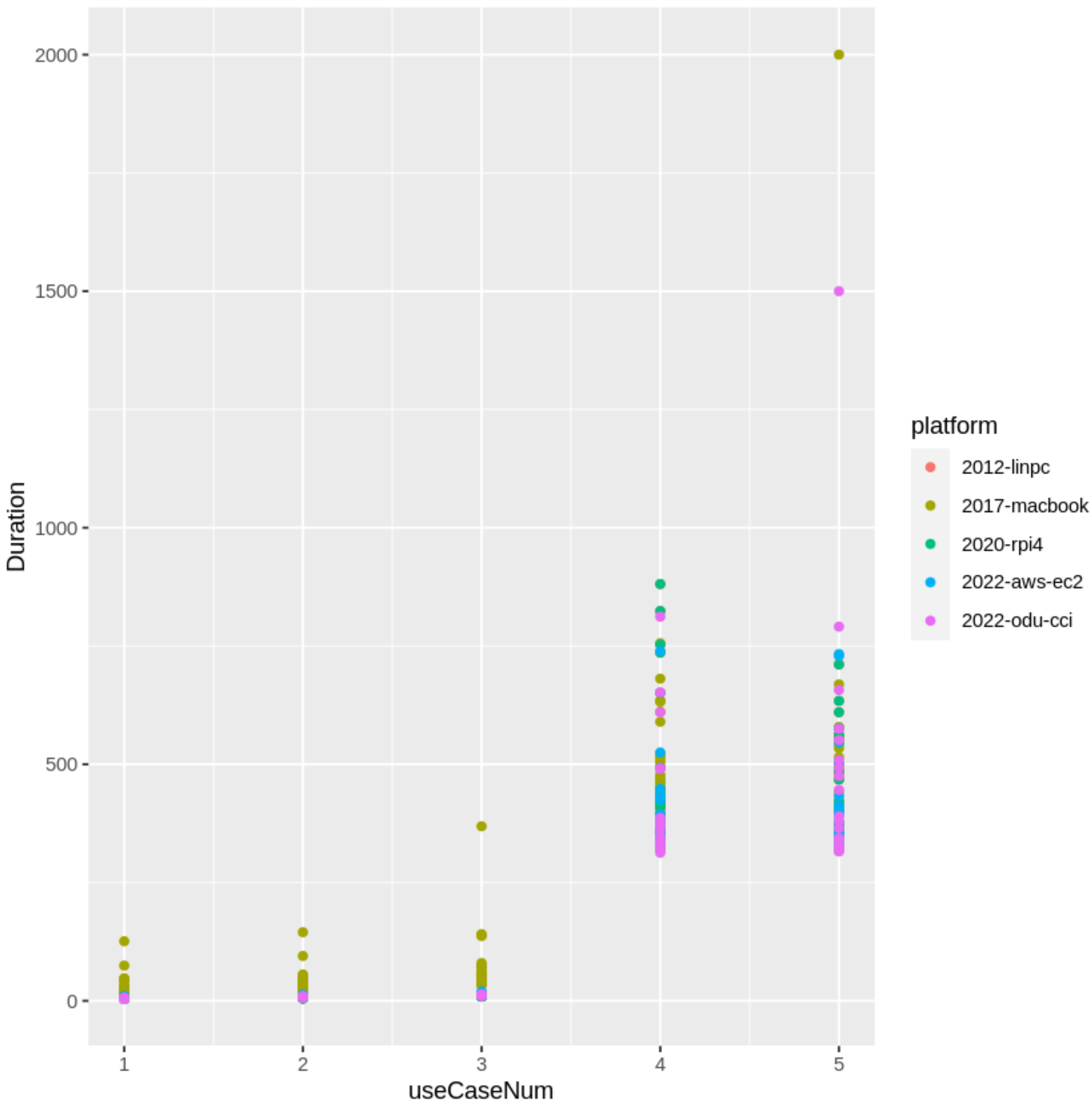
```r
# Convert character data into numeric metrics

for(index in 1:nrow(spanMetrics)) {       # for-loop over rows

    # Convert span duration

    char = spanMetrics[index,4]
    len = str_length(char)                # from tidyverse
    duration = str_sub(char,1,(len-3))
    units = str_sub(char,(len-1),len)
    duration <- as.numeric(duration)
```

```
    # glimpse(duration)
    # print(units)

    if(units == 'ms') {
        duration = duration                # Keep ms
    } else if (units == 'µs') {
        duration = duration * 0.001        # Convert µs to ms
    } else if (units == ' s') {
        duration = duration * 1000         # Convert s to ms
    } else {
        print ('Unable to find specified units')
        print (units)
    }
    spanMetrics$Duration[index] = as.numeric(duration)
}

# Convert Duration columns from char to numeric
spanMetrics$Duration = as.numeric(spanMetrics$Duration)

glimpse(spanMetrics)
```

```
Rows: 500
Columns: 9
$ Trace.ID   <chr> "9ee3577fb1b427bc4fc17fecc5154d7d", "f05ddc4dc13aff5c309801…
$ Trace.name <chr> "/TE", "/tracks", "/IAD", "/RIC", "/WA", "/TE", "/tracks", …
$ Start.time <chr> "2022-05-02 10:25:01.366", "2022-05-02 10:25:00.309", "2022…
$ Duration   <dbl> 36.0, 43.3, 464.0, 494.0, 139.0, 30.3, 30.0, 478.0, 546.0, …
$ platform   <chr> "2017-macbook", "2017-macbook", "2017-macbook", "2017-macbo…
$ env        <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,…
$ useCase    <chr> "Trial Engage (Internal)", "Get Stored Local DSS Tracks (In…
$ useCaseNum <dbl> 2, 1, 4, 5, 3, 2, 1, 4, 5, 3, 2, 1, 4, 5, 3, 2, 1, 4, 5, 3,…
$ ext        <lgl> FALSE, FALSE, TRUE, TRUE, FALSE, FALSE, FALSE, TRUE, TRUE, …
```

## 4 Exploratory Analysis Plots

```
qplot(useCaseNum, Duration, data = spanMetrics, colour = platform)
```



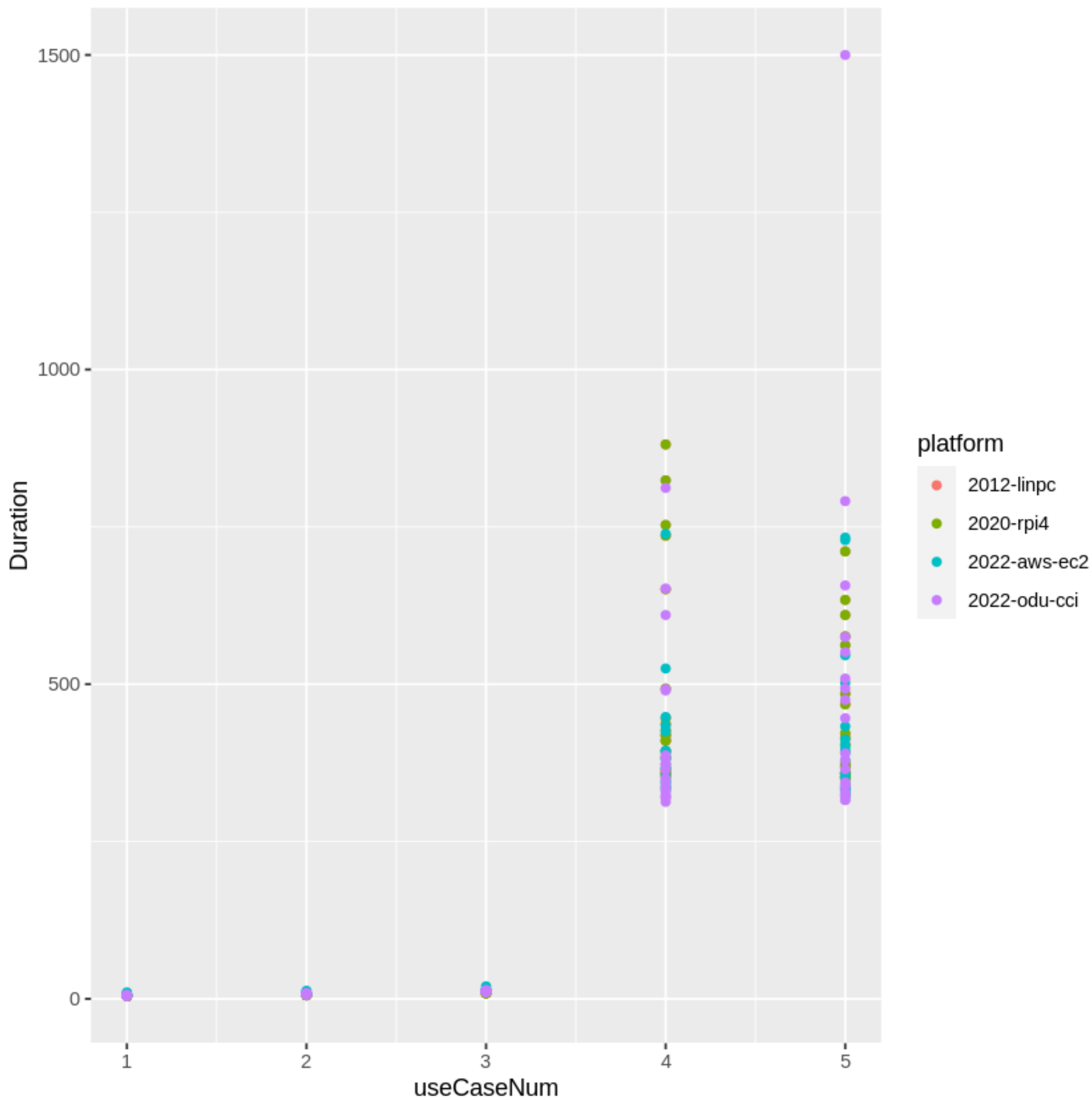

## 4.1 Remove Macbook Data from Development Platform

Here we remove the data from the Macbook development platform. The qplot shows that the **Mac implementation of Docker** adds latency within the Docker environment. In non-linux based plaforms, a Docker desktop running a virtual machine is required to provided that Docker capability that is native to Linux platforms. The Mac is considered to be the development environment and not representative of the integration and production environments.

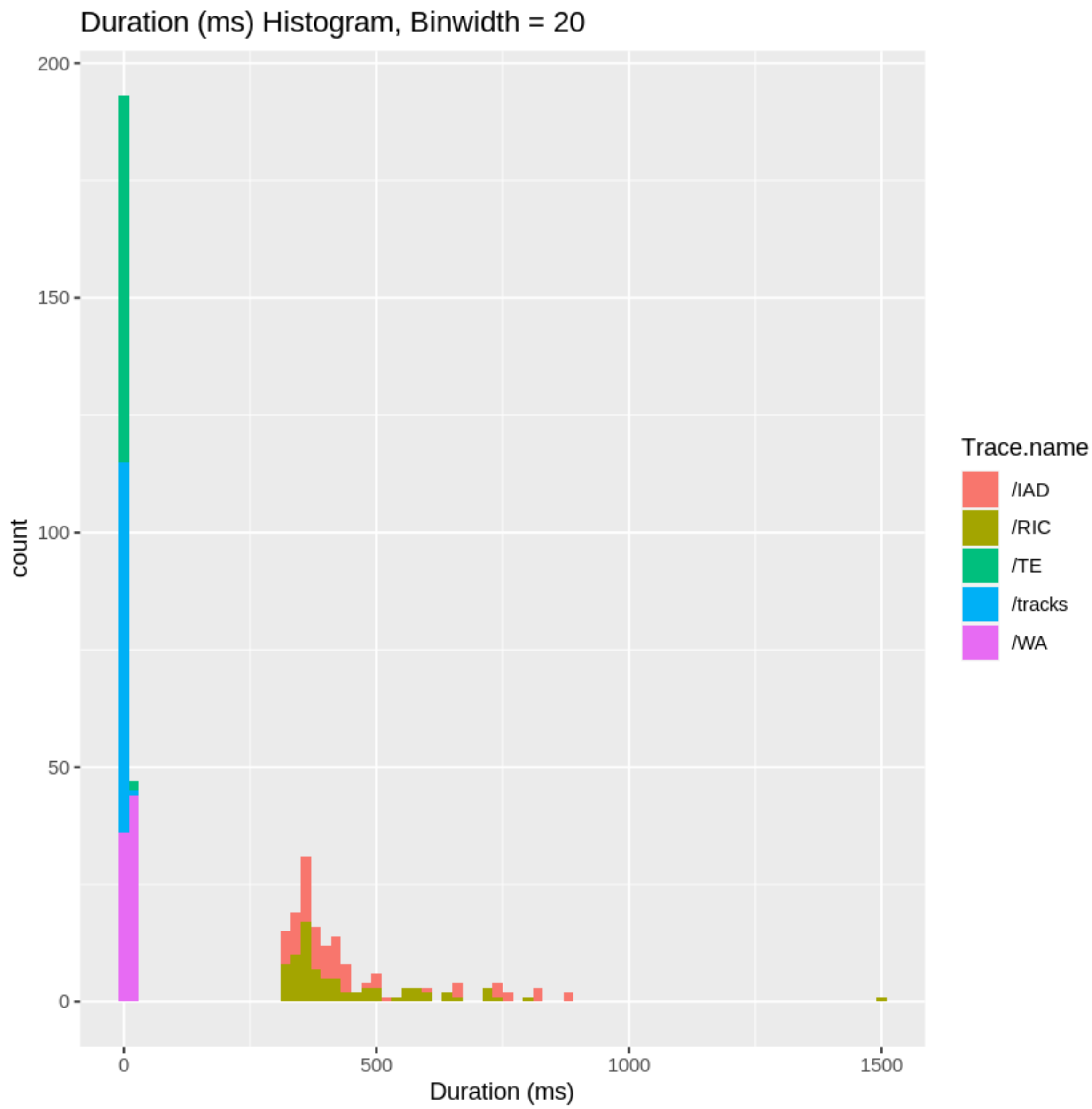
https://dev.to/ericnograles/why-is-docker-on-macos-so-much-worse-than-linux-flh
https://collabnix.com/how-docker-for-mac-works-under-the-hood/

```
noMacSpan <- spanMetrics[!spanMetrics$env == 0,]
```

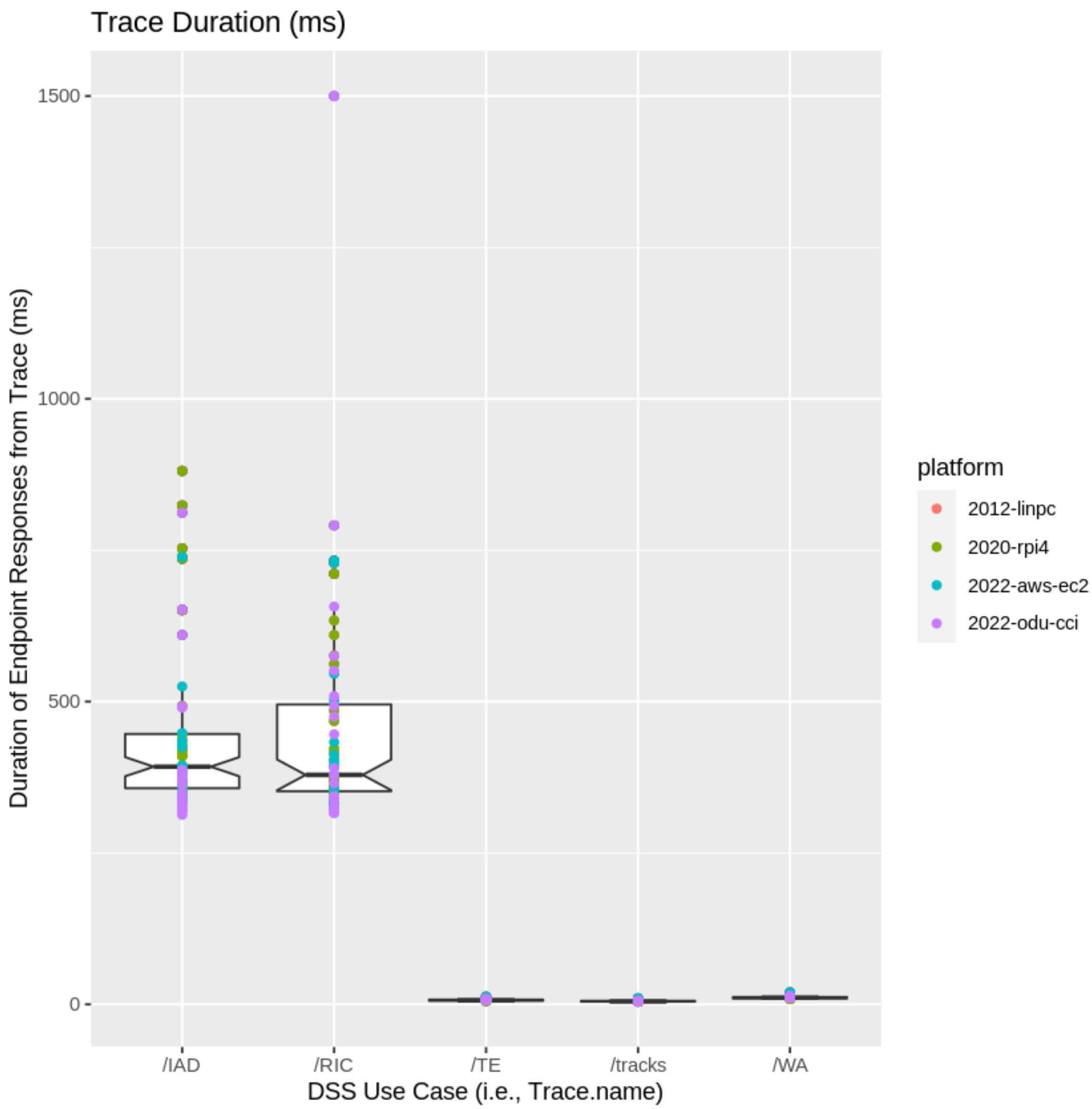
```
qplot(useCaseNum, Duration, data = noMacSpan, colour = platform)
```



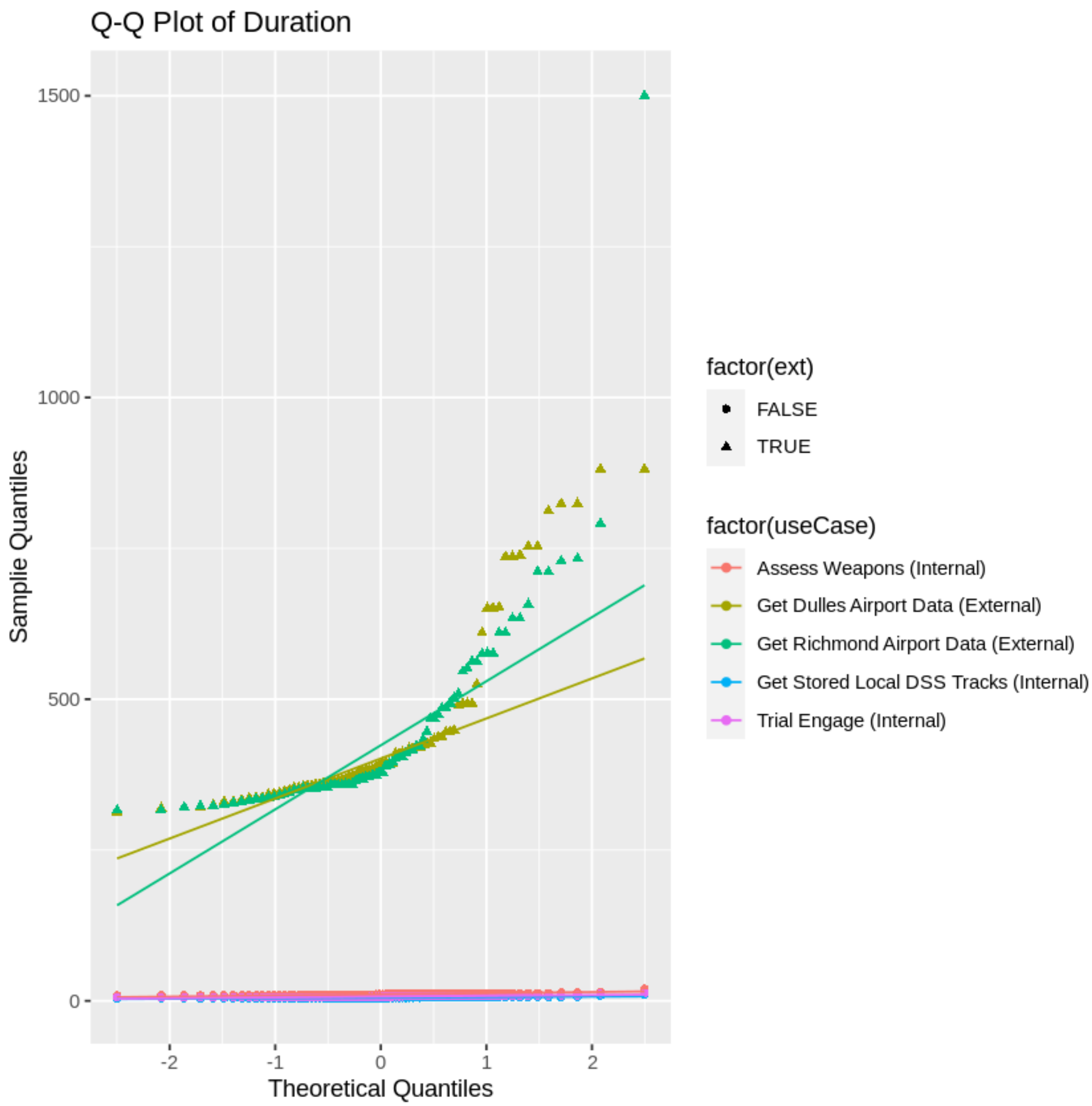
```
noMacSpan %>%
    ggplot(aes(Duration, fill = Trace.name)) + geom_histogram(binwidth = 20) +
    ggtitle("Duration (ms) Histogram, Binwidth = 20") +
    xlab("Duration (ms)")

noMacSpan %>%
    ggplot(aes(Trace.name, Duration)) +
    stat_boxplot(notch="TRUE") +
    geom_point(aes(colour = platform)) +
    ggtitle("Trace Duration (ms)") +
    ylab("Duration of Endpoint Responses from Trace (ms)") +
    xlab("DSS Use Case (i.e., Trace.name)")
# If notch went outside hinges. Try setting notch=FALSE.
```



Duration (ms) Histogram, Binwidth = 20
count
200
150
100
50
0
0
500
1000
1500
Duration (ms)
Trace.name
/IAD
/RIC
/TE
/tracks
/WA




```r
noMacSpan %>%
    ggplot(aes(sample = Duration, colour = factor(useCase), shape = factor(ext))) +
    stat_qq() +
    stat_qq_line() +
    labs(title="Q-Q Plot of Duration",
       x = "Theoretical Quantiles", y = "Samplie Quantiles")
```




The plots above seems to indicate the presence of 2 clusters. Each plot shows that internal and external duration data is heavily separated. We shall use mclust to investigate.

## 4.2 mclust

Used mclust to verify the separation of internal and external models as indicated from the useCaseNum vs. Duration plot; i.e. use cases 4 and 5 use an external API.

The library mclust is a contributed R package for model-based clustering, classification, and density estimation based on finite normal mixture modelling. It provides functions for parameter estimation via the EM algorithm for normal mixture models with a variety of covariance structures, and functions for simulation from these models.

mclustBIC returns an object of class 'mclustBIC' containing the Bayesian Information Criterion for the specified mixture models numbers of clusters. Auxiliary information is returned as attributes.

*Scrucca L., Fop M., Murphy T. B. and Raftery A. E. (2016) mclust 5: clustering, classification and density estimation using Gaussian finite mixture models The R Journal 8/1, pp. 289-317*

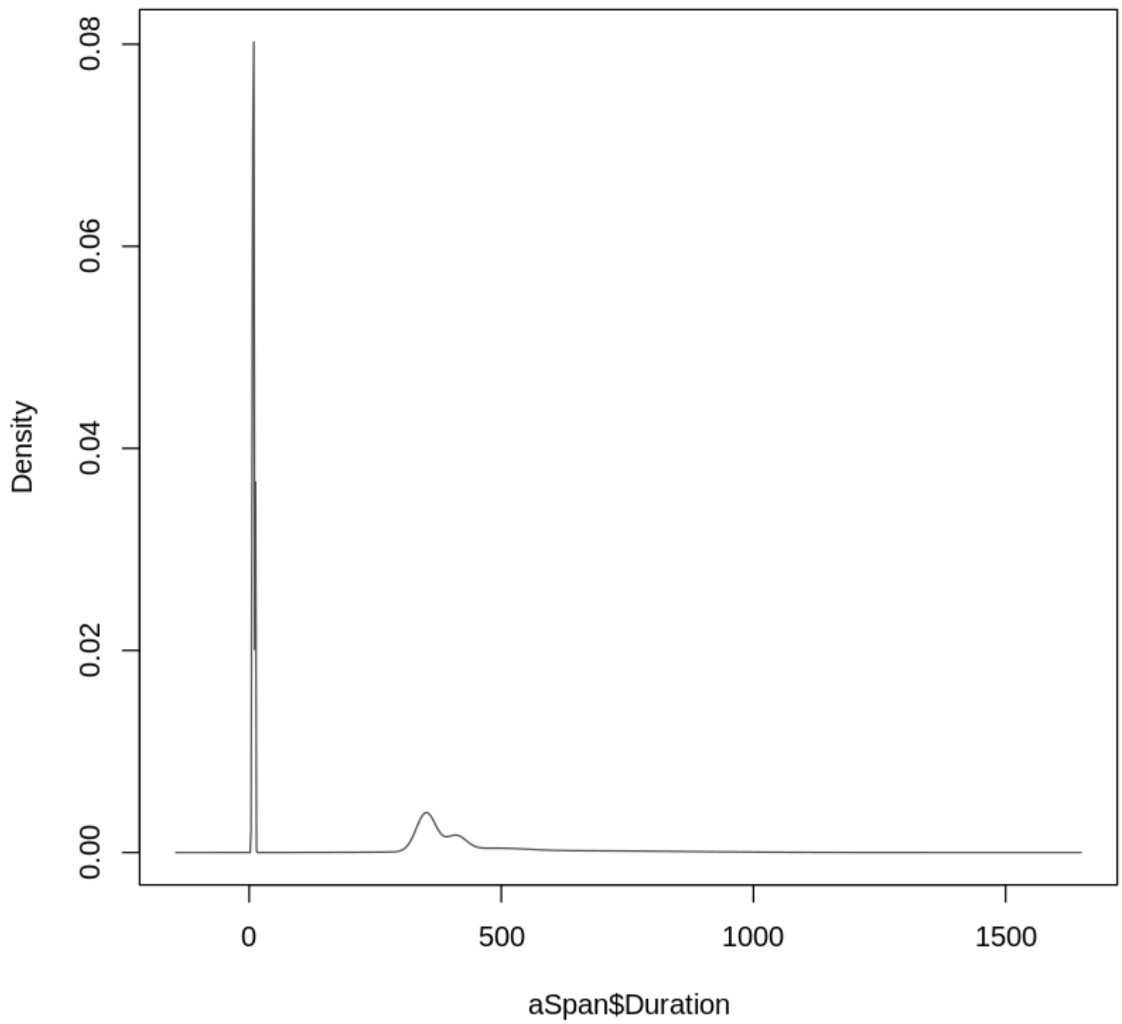
```
install.packages("mclust")
library(mclust, quietly = TRUE)
```

```
Error in download.file(url, destfile, method, mode = "wb", ...) :
  download from 'https://cran.r-project.org/src/contrib/mclust_5.4.10.tar.gz' failed
```

```
aSpan <- noMacSpan
```

### 4.2.1 Mclust Univariate Analysis of Duration

```
mod4 <- densityMclust(aSpan$Duration)
```



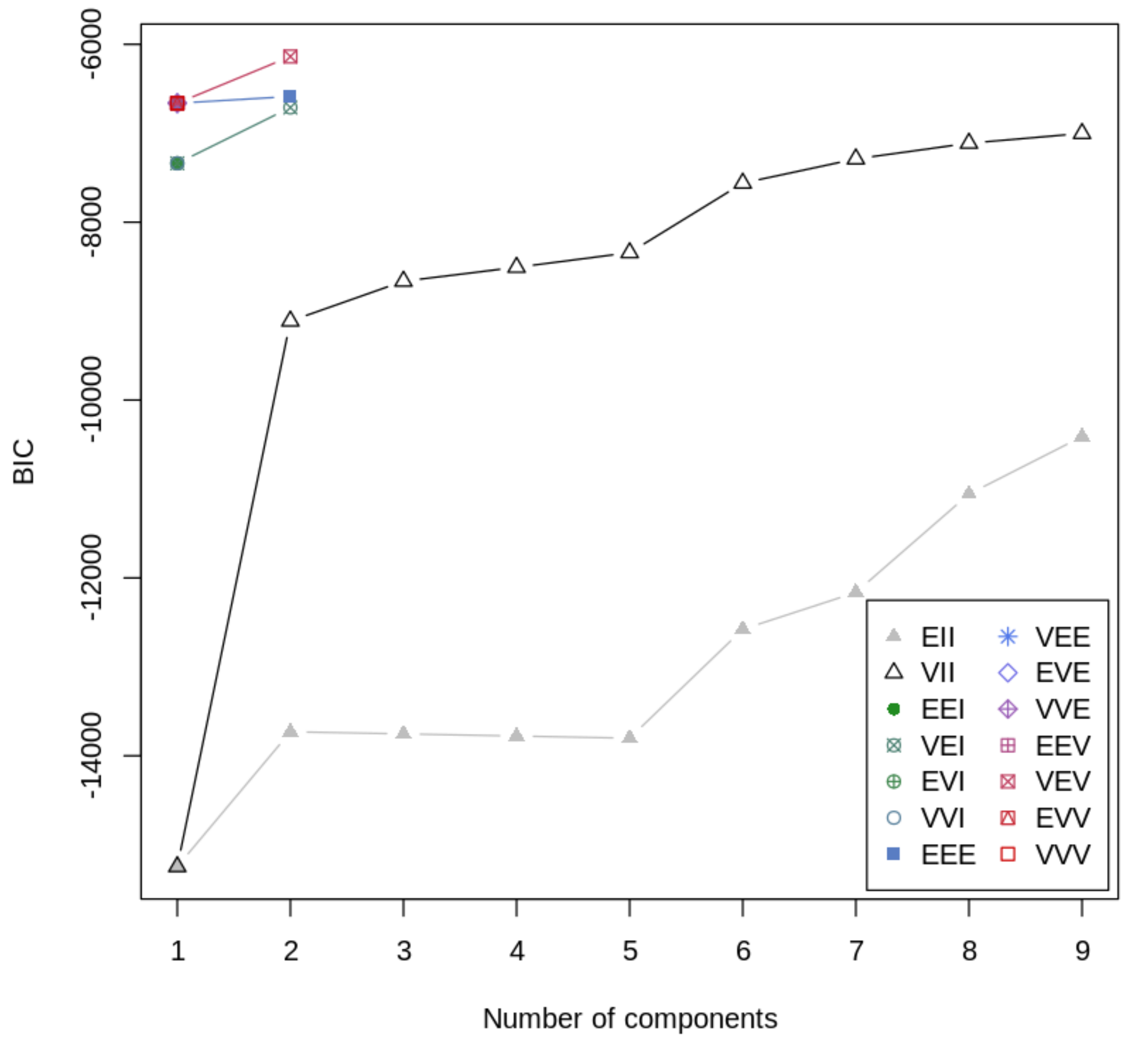

Based upon the plots above, it’s can seen that 2 clusters are indicated here separating internal and external data.

### 4.2.2 Mclust Multivariate Analysis

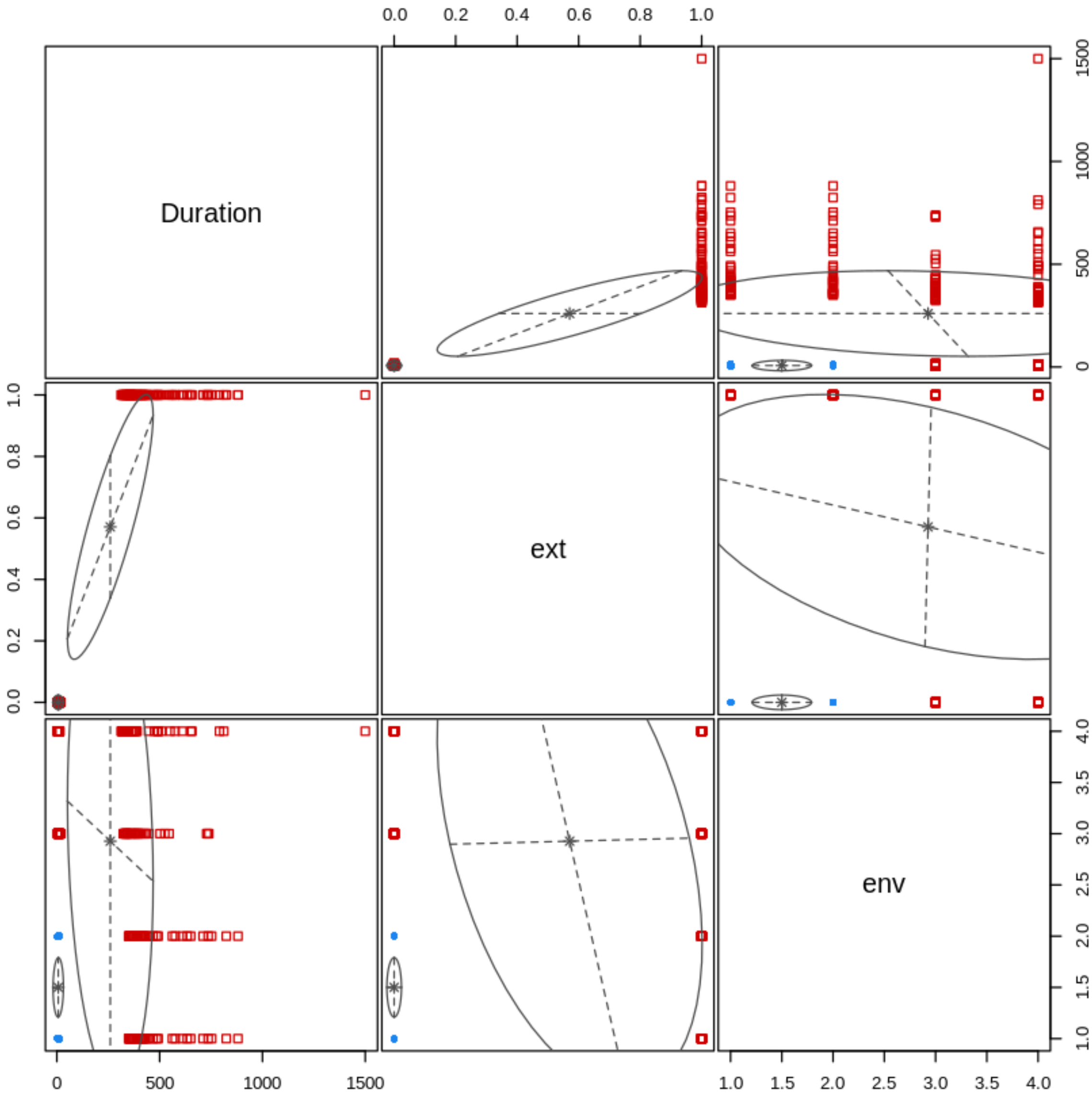
```r
uc <- aSpan$useCaseNum # Trace.name is char, used uc num conversion

# extract numerical data
X <- aSpan %>%
    # dplyr::select(useCaseNum, env, ext, Duration)
    dplyr::select(Duration, ext, env)
    # dplyr::select(Duration)

summary(X)
```

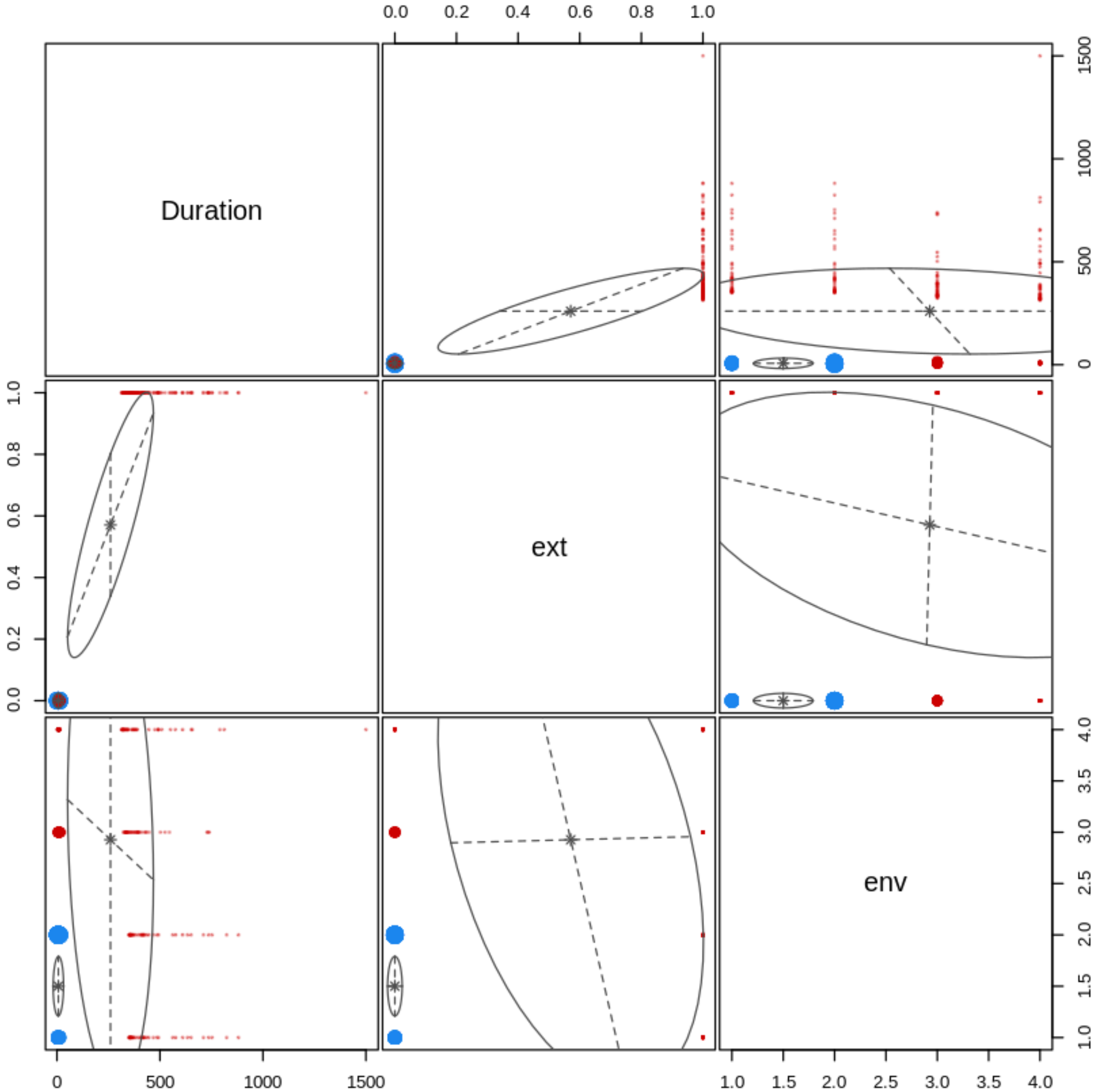
```
   Duration           ext                 env      
 Min.   :   4.290   Mode :logical   Min.   :1.00  
 1st Qu.:   6.048   FALSE:240       1st Qu.:1.75  
 Median :  11.150   TRUE :160       Median :2.50  
 Mean   : 184.172                   Mean   :2.50  
 3rd Qu.: 365.250                   3rd Qu.:3.25  
 Max.   :1500.000                   Max.   :4.00  
```

```r
BIC <- mclustBIC(X)
plot(BIC)

BIC <- mclustBIC(X)
mod1 <- Mclust(X, x = BIC)

plot(BIC, what = c("classification"))
plot(mod1, what = c("classification"))
plot(mod1, what = c("uncertainty"))
```



BIC
-6000
-8000
-10000
-12000
-14000
1
2
3
4
5
6
7
8
9
Number of components
EII
VII
EEI
VEI
EVI
VVI
EEE
VEE
EVE
VVE
EEV
VEV
EVV
VVV


Duration
ext
env
0.0
0.2
0.4
0.6
0.8
1.0
0
500
1000
1500
1.0
1.5
2.0
2.5
3.0
3.5
4.0




```r
summary(BIC)
```

```
Best BIC values:
             VEV,2      EEE,2      EEE,1
BIC      -6136.963 -6586.3351 -6662.2804
BIC diff     0.000  -449.3724  -525.3177
```

Note that 2 is included within the list of best Bayesian Information Criterion (BIC) values indicating two clusters.

VEV:varying volume,equal shape,varying orientation (ellipsoidal covariance)
EEE:equal volume,equal shape,equal orientation (ellipsoidal covariance)

We shall separate internal from external data.

# 5 Separating Internal from External Data

## 5.1 Internal Data

```
# Separate Internal Data
# Could use ext == FALSE

tracksSpanData = subset(aSpan, useCaseNum == 1)
TE_SpanData = subset(aSpan, useCaseNum == 2)
WA_SpanData = subset(aSpan, useCaseNum == 3)

internalSpanData <- rbind(tracksSpanData, TE_SpanData, WA_SpanData)
dssSpanData <- rbind(TE_SpanData, WA_SpanData)
```

```
iSpan <- internalSpanData
```

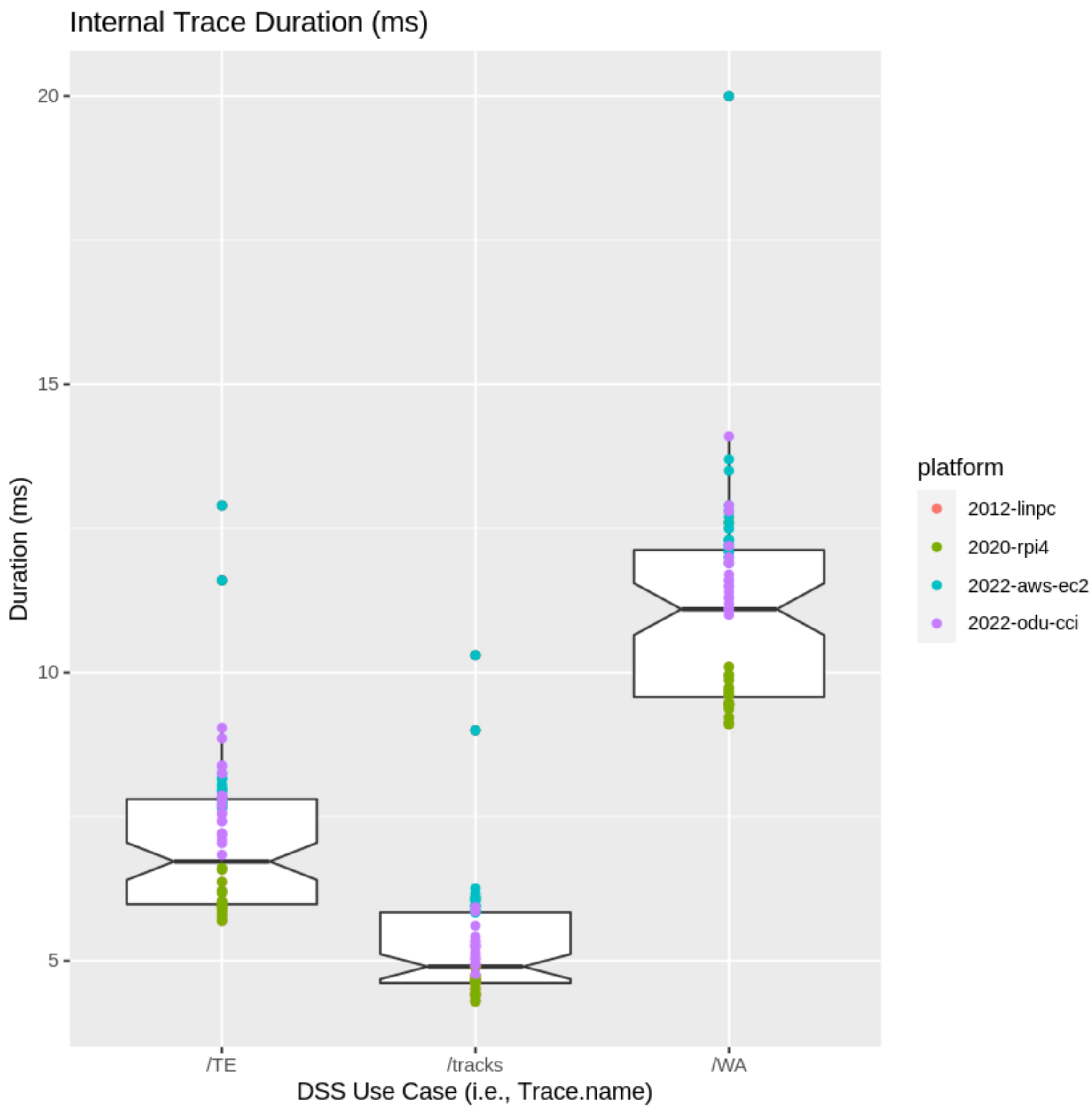
```
summary(iSpan)
sd(iSpan$Duration)
```

```
   Trace.ID           Trace.name          Start.time           Duration
 Length:240         Length:240         Length:240         Min.   : 4.290
 Class :character   Class :character   Class :character   1st Qu.: 5.713
 Mode  :character   Mode  :character   Mode  :character   Median : 7.070
                                                          Mean   : 7.745
                                                          3rd Qu.: 9.610
                                                          Max.   :20.000
   platform              env           useCase            useCaseNum
 Length:240         Min.   :1.00   Length:240         Min.   :1
 Class :character   1st Qu.:1.75   Class :character   1st Qu.:1
 Mode  :character   Median :2.50   Mode  :character   Median :2
                    Mean   :2.50                      Mean   :2
                    3rd Qu.:3.25                      3rd Qu.:3
```

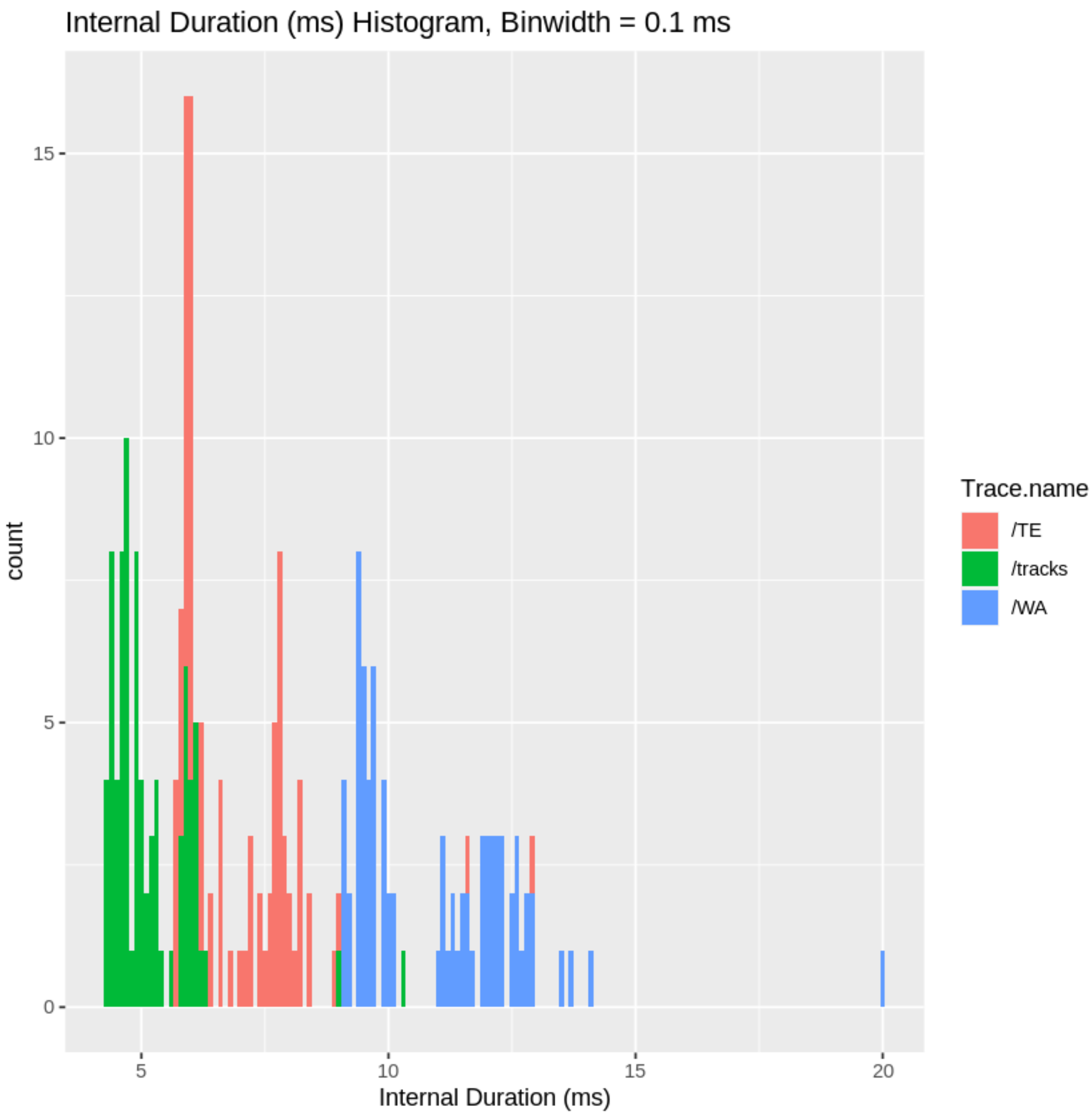
```
                    Max.   :4.00                         Max.   :3
   ext
 Mode :logical
 FALSE:240
```

2.77664210997812

```
iSpan %>%
    ggplot(aes(Trace.name, Duration)) +
    stat_boxplot(notch="TRUE") + geom_point(aes(colour = platform)) +
    ggtitle("Internal Trace Duration (ms)") +
    ylab("Duration (ms)") +
    xlab("DSS Use Case (i.e., Trace.name)")
```




```
iSpan %>%
    ggplot(aes(Duration, fill = Trace.name)) + geom_histogram(binwidth = 0.1) +
    ggtitle("Internal Duration (ms) Histogram, Binwidth = 0.1 ms") +
    xlab("Internal Duration (ms)")
```




Note that the histogram plot indicates that the data is not normally distrbuted and suggests an adjustment will be needed enable application of statistics.

## 5.2 External Data

```
RIC_SpanData = subset(aSpan, useCaseNum == 5)
IAD_SpanData = subset(aSpan, useCaseNum == 4)

externalSpanData <- rbind(RIC_SpanData, IAD_SpanData)
```

```
eSpan <- externalSpanData
```

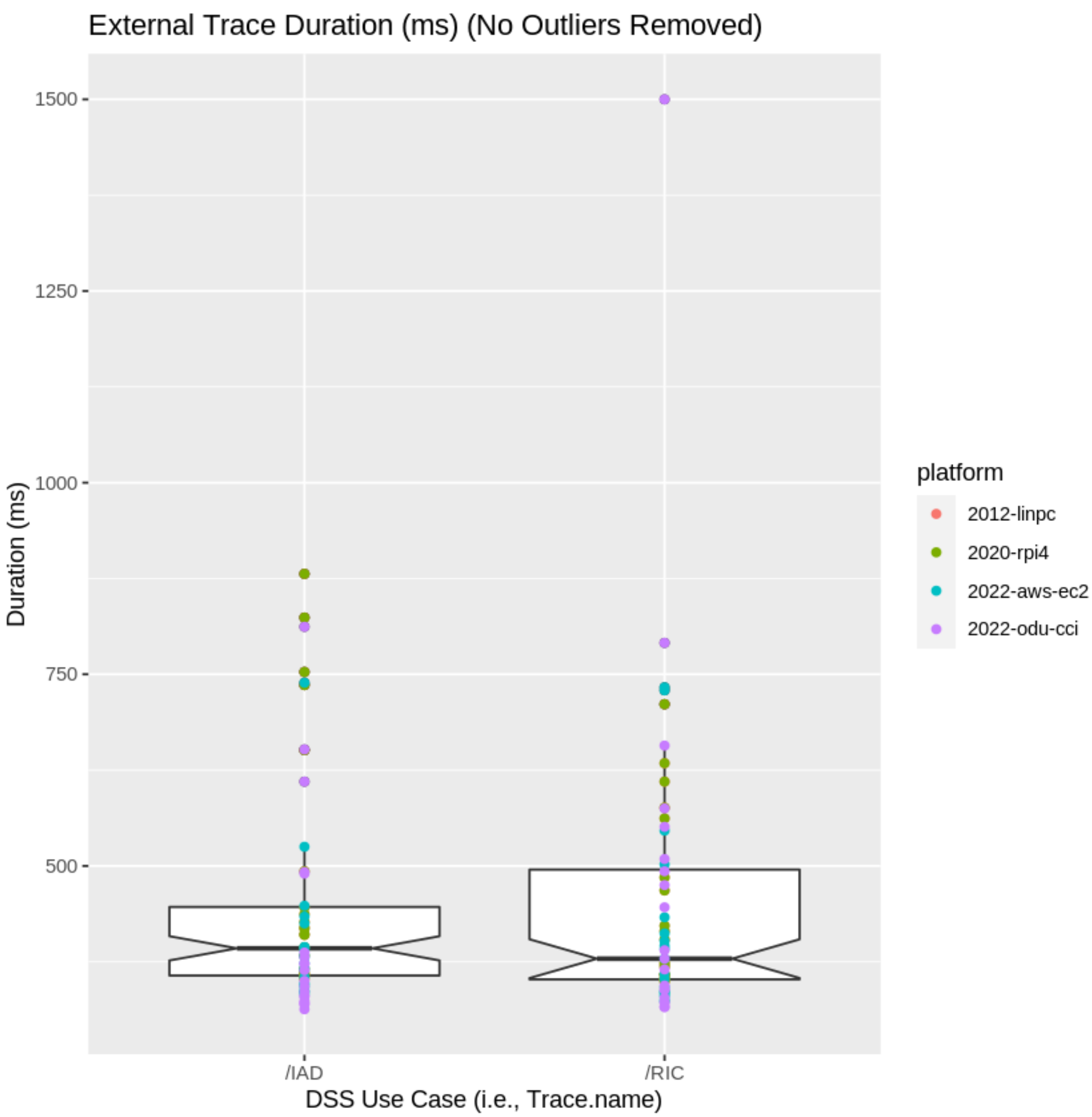
```
summary(eSpan)
sd(eSpan$Duration)
```

```
   Trace.ID           Trace.name          Start.time           Duration
 Length:160         Length:160         Length:160         Min.   : 313.0
 Class :character   Class :character   Class :character   1st Qu.: 353.0
 Mode  :character   Mode  :character   Mode  :character   Median : 388.5
                                                          Mean   : 448.8
                                                          3rd Qu.: 486.2
                                                          Max.   :1500.0
   platform              env          useCase            useCaseNum
 Length:160         Min.   :1.00   Length:160         Min.   :4.0
 Class :character   1st Qu.:1.75   Class :character   1st Qu.:4.0
 Mode  :character   Median :2.50   Mode  :character   Median :4.5
                    Mean   :2.50                      Mean   :4.5
                    3rd Qu.:3.25                      3rd Qu.:5.0
                    Max.   :4.00                      Max.   :5.0
   ext
 Mode:logical
 TRUE:160
```

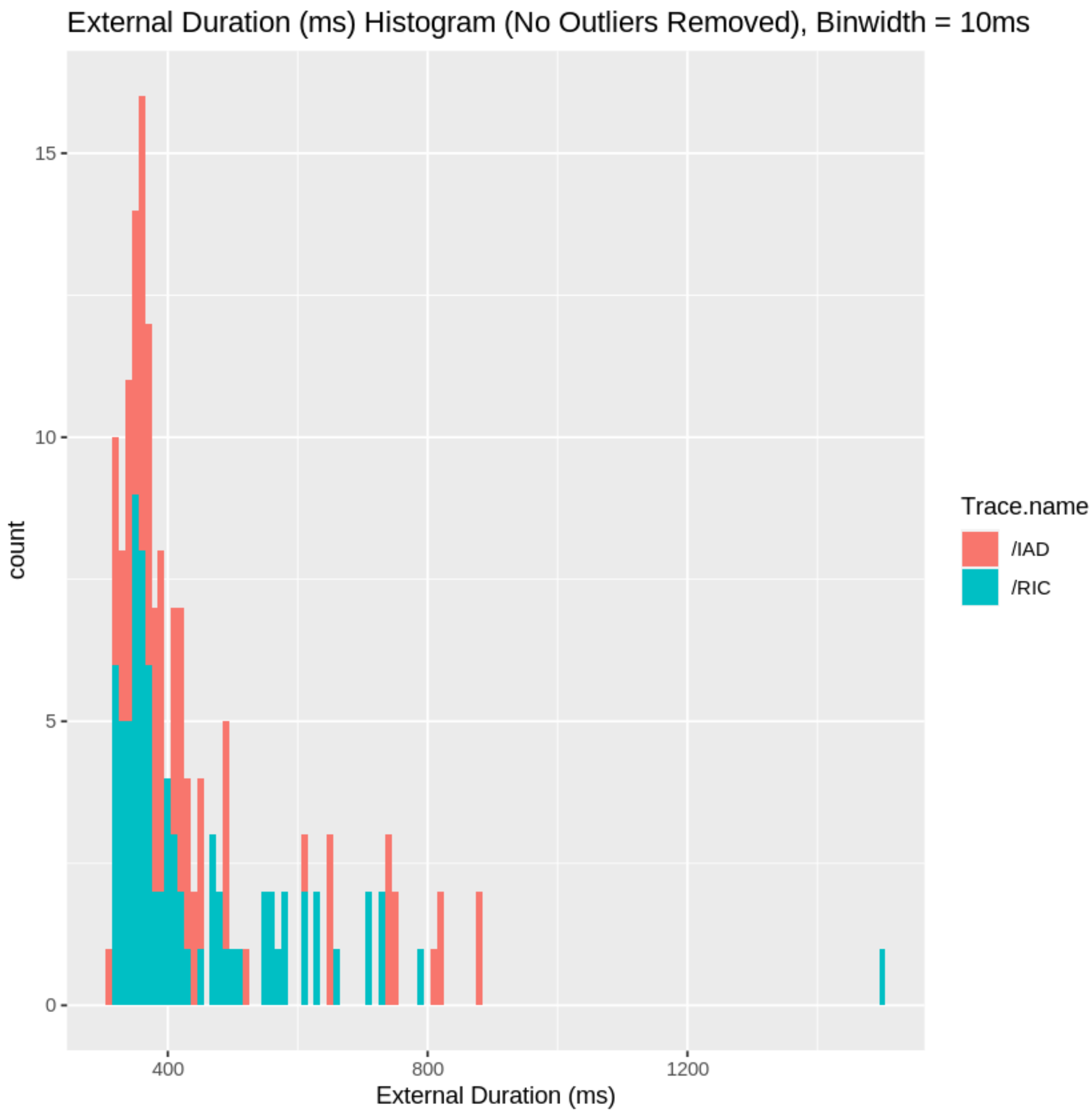
158.836658613485

```
eSpan %>%
    ggplot(aes(Trace.name, Duration)) +
    stat_boxplot(notch="TRUE") + geom_point(aes(colour = platform)) +
    ggtitle("External Trace Duration (ms) (No Outliers Removed)") +
```

```
    ylab("Duration (ms)") +
    xlab("DSS Use Case (i.e., Trace.name)")
```




```
eSpan %>%
    ggplot(aes(Duration, fill = Trace.name)) + geom_histogram(binwidth = 10) +
    ggtitle("External Duration (ms) Histogram (No Outliers Removed), Binwidth = 10ms") +
    xlab("External Duration (ms)")
```




## 6 Shapiro-Wilk Test for Normal Distribution

The null-hypothesis of this test is that the population is normally distributed. Thus, if the p value is less than the chosen alpha level, then the null hypothesis is rejected and there is evidence that the data tested are not normally distributed. On the other hand, if the p value is greater than the chosen alpha level, then the null hypothesis (that the data came from a normally distributed population) can not be rejected (e.g., for an alpha level of .05, a data set with a p value of less than .05 rejects the null hypothesis that the data are from a normally distributed population).

https://en.wikipedia.org/wiki/Shapiro–Wilk_test

```
shapiro.test(iSpan$Duration)
shapiro.test(eSpan$Duration)
```

```

    Shapiro-Wilk normality test

data:  iSpan$Duration
W = 0.9075, p-value = 5.081e-11


    Shapiro-Wilk normality test

data:  eSpan$Duration
W = 0.71543, p-value = 3.053e-16
```

The result indicates that the data is not normally distributed and needs to be adjusted to apply hypothesis testing. The EDA plots indicate significant gaps in the internal data with short durations. This seems to indicate measuring an API response with very little variation. The R rnorm function is used to add a normally distributed processing delay with a mean of 50 ms and a standard deviation of 10 ms.

## 6.1 Addition of Processing Delay

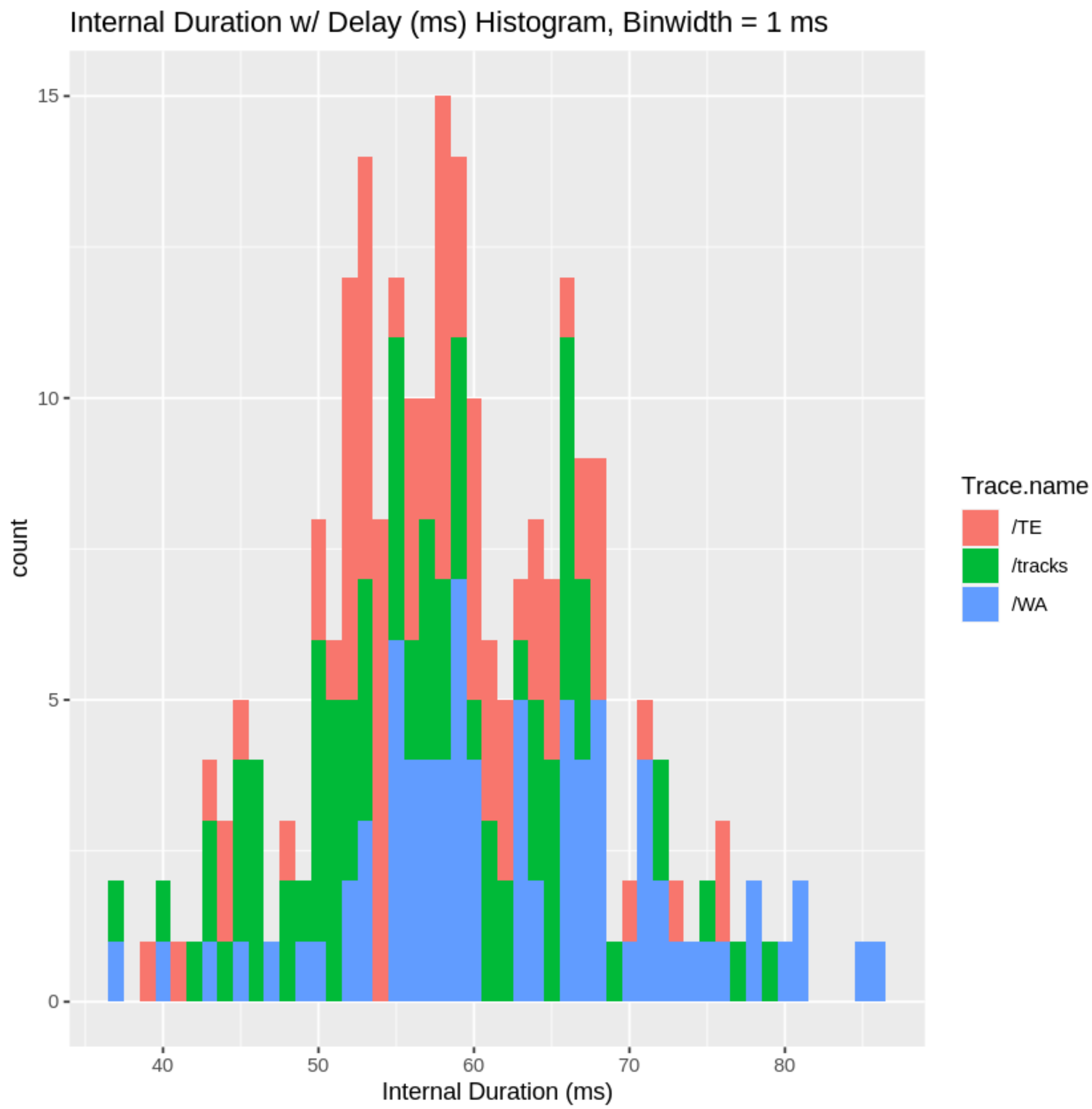
```
# Add processing delay from a normal distribution
pd_iSpan <- iSpan

for(index in 1:nrow(pd_iSpan)) {       # for-loop over rows

    # Add processing delay from a normal distribution
    pd <- rnorm(1, mean = 50, sd = 10)
    pd_iSpan[index,4] = pd_iSpan[index,4] + pd
}

pd_eSpan <- eSpan

for(index in 1:nrow(pd_eSpan)) {       # for-loop over rows
```

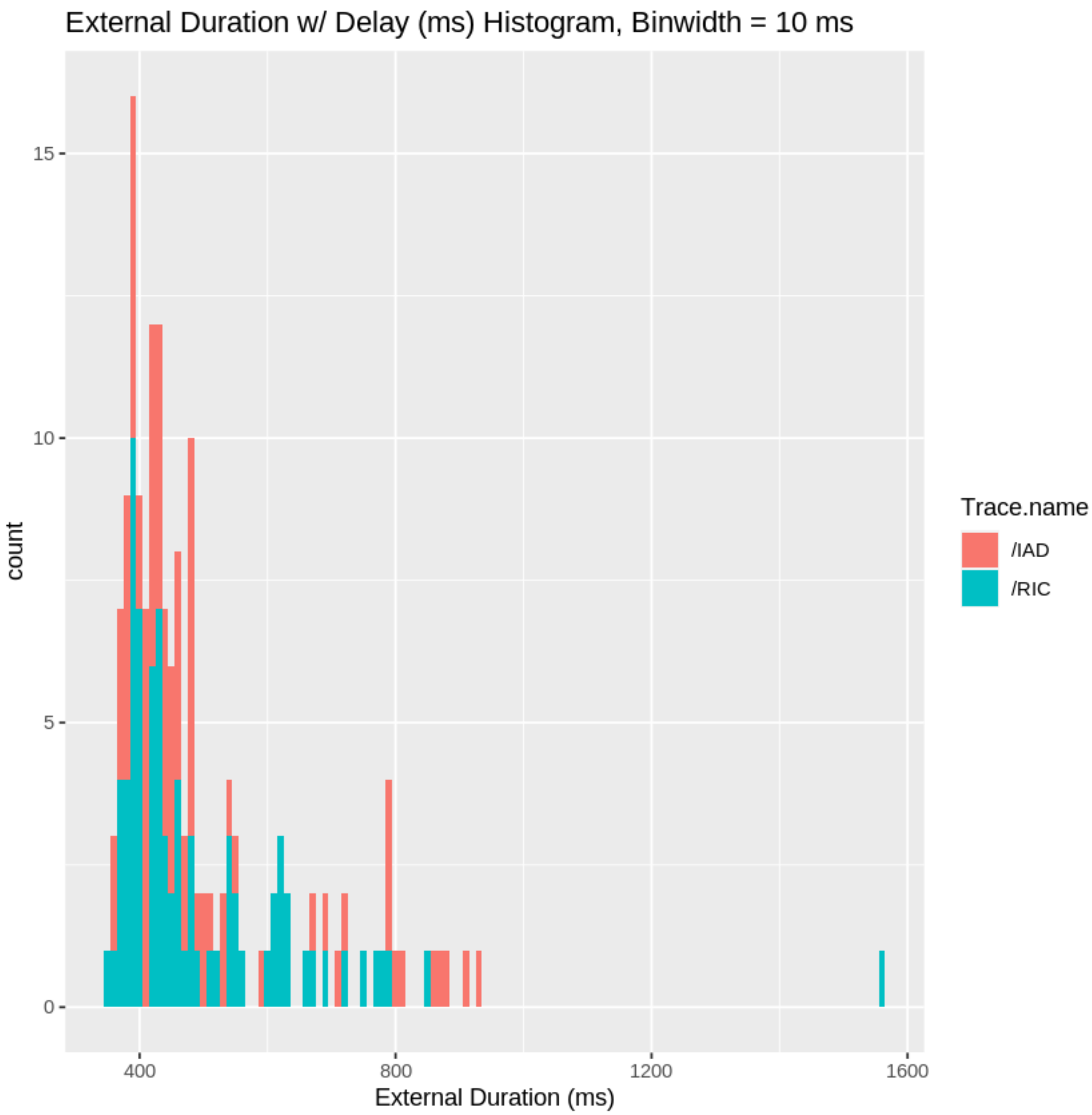
```r
    # Add processing delay from a normal distribution
    pd <- rnorm(1, mean = 50, sd = 10)
    pd_eSpan[index,4] = pd_eSpan[index,4] + pd
}

# glimpse(pd_iSpan)
```

```r
shapiro.test(pd_iSpan$Duration)
shapiro.test(pd_eSpan$Duration)
```

```

	Shapiro-Wilk normality test

data:  pd_iSpan$Duration
W = 0.9921, p-value = 0.2265

	Shapiro-Wilk normality test

data:  pd_eSpan$Duration
W = 0.72568, p-value = 6.034e-16
```

```r
pd_iSpan %>%
    ggplot(aes(Duration, fill = Trace.name)) + geom_histogram(binwidth = 1) +
    ggtitle("Internal Duration w/ Delay (ms) Histogram, Binwidth = 1 ms") +
    xlab("Internal Duration (ms)")

pd_eSpan %>%
    ggplot(aes(Duration, fill = Trace.name)) + geom_histogram(binwidth = 10) +
    ggtitle("External Duration w/ Delay (ms) Histogram, Binwidth = 10 ms") +
    xlab("External Duration (ms)")
```


Internal Duration w/ Delay (ms) Histogram, Binwidth = 1 ms
15
10
5
0
count
40
50
60
70
80
Internal Duration (ms)
Trace.name
/TE
/tracks
/WA




The data indicates that a normal distribution is achieved with the internal data but not with the external data do to the extreme variation in response times through external servers and routers. We shall apply a t-test for the internal data, but for the external data we shall apply a binomial test to see if the threshold of 500 ms can be maintained. This will require adding a threshold indication to the data sets.

## 6.2 Addition of Threshold Indicator

```r
pd_iSpan$hthreshold = TRUE
pd_eSpan$hthreshold = TRUE
```

```r
# Assess hypothesis threshold of 500 ms

for(index in 1:nrow(pd_eSpan)) {        # for-loop over rows
    if(pd_eSpan[index,4] > 500) {
        pd_eSpan[index,10] = FALSE
        # spanMetrics$hthreshold = FALSE
    } else {
    pd_eSpan[index,10] = TRUE
    # spanMetrics$hthreshold = TRUE
    }
}

for(index in 1:nrow(pd_iSpan)) {        # for-loop over rows
    if(pd_iSpan[index,4] > 500) {
        pd_iSpan[index,10] = FALSE
        # spanMetrics$hthreshold = FALSE
    } else {
    pd_iSpan[index,10] = TRUE
    # spanMetrics$hthreshold = TRUE
    }
}
```

```r
glimpse(pd_eSpan)
```

```
Rows: 160
Columns: 10
$ Trace.ID   <chr> "0d8efde6f35af9599ae0ffc9cd68b6fb", "d6c36d3d53a329daf1f72e…
$ Trace.name <chr> "/RIC", "/RIC", "/RIC", "/RIC", "/RIC", "/RIC", "/RIC", "/R…
$ Start.time <chr> "2022-06-06 21:36:51.531", "2022-06-06 21:36:45.723", "2022…
$ Duration   <dbl> 476.2477, 416.5003, 671.6718, 404.2449, 427.0154, 391.0872,…
$ platform   <chr> "2012-linpc", "2012-linpc", "2012-linpc", "2012-linpc", "20…
$ env        <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,…
$ useCase    <chr> "Get Richmond Airport Data (External)", "Get Richmond Airpo…
$ useCaseNum <dbl> 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5,…
$ ext        <lgl> TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, TRUE,…
$ hthreshold <lgl> TRUE, TRUE, FALSE, TRUE, TRUE, TRUE, TRUE, FALSE, TRUE, TRU…
```

# 7 Hypothesis Testing

## 7.1 t-Test (Internal Data)

Given that we were able verify a **normal distribution** with the process delay applied to the internal span data, we are able to use a Student's t-Test to test the hypothesis on the internal span data. Our mean is 500 ms (e.g. $\mu = 500$ ms) and our null hypthesis is less than 500 ms. This is an example of what is called a one-tailed hypothesis; e.g. evidence against the null hypothesis comes from only one tail of the distribution (namely, duration above 500).

```
mu = 500
x = pd_iSpan$Duration
t.test(x=x, mu=mu, alternative = 'greater')
```

```
    One Sample t-test

data:  x
t = -745.31, df = 239, p-value = 1
alternative hypothesis: true mean is greater than 500
95 percent confidence interval:
 57.9299     Inf
sample estimates:
mean of x
 58.90716
```

A t-test of the internal span data indicates a p-value of 1 so we fail to reject the null hypothesis that that the duration mean is less than 500 ms. The p-value converges to 1 because all of the internal duration results are far less than 500 ms.

## 7.2 Binomial Tests

We'll use Binomial Tests to test the probability of success for meeting the 500 ms duration. For Binomial Test we need to review the number of trials and number of successes.

### 7.2.1 All Data

Let's look at the combined data first.

```
pd_aSpan <- rbind(pd_iSpan, pd_eSpan)
summary(pd_aSpan)
```

```
   Trace.ID           Trace.name         Start.time           Duration      
 Length:400         Length:400         Length:400         Min.   :  36.61  
 Class :character   Class :character   Class :character   1st Qu.:  56.59  
 Mode  :character   Mode  :character   Mode  :character   Median :  67.28  
                                                          Mean   : 235.09  
                                                          3rd Qu.: 421.57  
                                                          Max.   :1558.64  
   platform              env          useCase            useCaseNum
 Length:400         Min.   :1.00   Length:400         Min.   :1  
 Class :character   1st Qu.:1.75   Class :character   1st Qu.:2  
 Mode  :character   Median :2.50   Mode  :character   Median :3  
                    Mean   :2.50                      Mean   :3  
                    3rd Qu.:3.25                      3rd Qu.:4  
                    Max.   :4.00                      Max.   :5  
    ext          hthreshold    
 Mode :logical   Mode :logical  
 FALSE:240       FALSE:46       
 TRUE :160       TRUE :354      
```

```r
# 354 True, 400 Trials

# binom.test(354, 400, p = 0.95, alternative = "less")
binom.test(354, 400, alternative = "less")
```

```
	Exact binomial test

data:  354 and 400
number of successes = 354, number of trials = 400, p-value = 1
alternative hypothesis: true probability of success is less than 0.5
95 percent confidence interval:
 0.0000000 0.9102965
sample estimates:
probability of success 
                 0.885 
```

The Binomial Test indicates that the probability of success for all data is 89%. We next look at internal and external separately.

```r
pd_aSpan <- rbind(pd_iSpan, pd_eSpan)
summary(pd_aSpan$hthreshold == TRUE)
```

```
   Mode   FALSE    TRUE 
logical      46     354 
```

```r
# binom.test(354, 400, p = 0.95, alternative = "less")
# pass = subset(pd_aSpan, summary(pd_aSpan$hthreshold) == TRUE)
binom.test(354, 400, alternative = "less")
```

```

	Exact binomial test

data:  354 and 400
number of successes = 354, number of trials = 400, p-value = 1
alternative hypothesis: true probability of success is less than 0.5
95 percent confidence interval:
 0.0000000 0.9102965
sample estimates:
probability of success 
                 0.885 
```

Results indicate that we can achieve the threshold with a probability of success of 89%.

```r
summary(pd_iSpan)
names(pd_iSpan)
```

```
   Trace.ID          Trace.name         Start.time           Duration    
 Length:240         Length:240         Length:240         Min.   :36.61  
 Class :character   Class :character   Class :character   1st Qu.:52.64  
 Mode  :character   Mode  :character   Mode  :character   Median :58.39  
                                                          Mean   :58.91  
                                                          3rd Qu.:65.46  
                                                          Max.   :85.86  
   platform              env          useCase            useCaseNum
 Length:240         Min.   :1.00   Length:240         Min.   :1   
 Class :character   1st Qu.:1.75   Class :character   1st Qu.:1   
 Mode  :character   Median :2.50   Mode  :character   Median :2   
                    Mean   :2.50                      Mean   :2   
                    3rd Qu.:3.25                      3rd Qu.:3   
```

```
                    Max.   :4.00                          Max.   :3
    ext            hthreshold
 Mode :logical   Mode:logical
 FALSE:240       TRUE:240
```

1. 'Trace.ID'
2. 'Trace.name'
3. 'Start.time'
4. 'Duration'
5. 'platform'
6. 'env'
7. 'useCase'
8. 'useCaseNum'
9. 'ext'
10. 'hthreshold'

```
min(pd_iSpan$Duration)
```

36.6068315642576

```
mu = 500
x = pd_iSpan$Duration
t.test(x=x, mu=mu, alternative = 'greater')
```

```
    One Sample t-test

data:  x
t = -745.31, df = 239, p-value = 1
alternative hypothesis: true mean is greater than 500
95 percent confidence interval:
 57.9299     Inf
sample estimates:
mean of x
 58.90716
```

The results of the t-Test on the normalized internal data indicates a p-value = 1. This because the results are well below $\mu = 500$ ms.

### 7.2.2 Internal Data

```
summary(pd_iSpan)
```

```
   Trace.ID          Trace.name         Start.time           Duration    
 Length:240         Length:240         Length:240         Min.   :36.61  
 Class :character   Class :character   Class :character   1st Qu.:52.64  
 Mode  :character   Mode  :character   Mode  :character   Median :58.39  
                                                          Mean   :58.91  
                                                          3rd Qu.:65.46  
                                                          Max.   :85.86  
   platform              env         useCase            useCaseNum
 Length:240         Min.   :1.00   Length:240         Min.   :1   
 Class :character   1st Qu.:1.75   Class :character   1st Qu.:1   
 Mode  :character   Median :2.50   Mode  :character   Median :2   
                    Mean   :2.50                      Mean   :2   
                    3rd Qu.:3.25                      3rd Qu.:3   
                    Max.   :4.00                      Max.   :3   
    ext          hthreshold    
 Mode :logical   Mode:logical  
 FALSE:240       TRUE:240      
```

```
# 240 True, 240 Trials

# binom.test(240, 240, p = 0.95, alternative = "less")
binom.test(240, 240, alternative = "less")
```

```
	Exact binomial test

data:  240 and 240
number of successes = 240, number of trials = 240, p-value = 1
alternative hypothesis: true probability of success is less than 0.5
95 percent confidence interval:
 0 1
sample estimates:
probability of success 
                     1 
```

The Binomial Test indicates that the probability of success is 100% for internal data. This is consistent with the t-Test results.

### 7.2.3 External Data

```r
summary(pd_eSpan)
```

```
   Trace.ID          Trace.name         Start.time           Duration     
 Length:160         Length:160         Length:160         Min.   : 348.7  
 Class :character   Class :character   Class :character   1st Qu.: 401.2  
 Mode  :character   Mode  :character   Mode  :character   Median : 440.1  
                                                          Mean   : 499.4  
                                                          3rd Qu.: 535.7  
                                                          Max.   :1558.6  
   platform              env          useCase            useCaseNum 
 Length:160         Min.   :1.00   Length:160         Min.   :4.0  
 Class :character   1st Qu.:1.75   Class :character   1st Qu.:4.0  
 Mode  :character   Median :2.50   Mode  :character   Median :4.5  
                    Mean   :2.50                      Mean   :4.5  
                    3rd Qu.:3.25                      3rd Qu.:5.0  
                    Max.   :4.00                      Max.   :5.0  
   ext          hthreshold     
 Mode:logical   Mode :logical  
 TRUE:160       FALSE:46       
                TRUE :114      
```

```r
# 114 True, 160 Trials

# binom.test(114, 160, p = 0.95, alternative = "less")
binom.test(114, 160, alternative = "less")
```

```
	Exact binomial test

data:  114 and 160
number of successes = 114, number of trials = 160, p-value = 1
alternative hypothesis: true probability of success is less than 0.5
95 percent confidence interval:
```

```
 0.0000000 0.7711356
sample estimates:
probability of success
                0.7125
```

The Binomial Test indicates that the probability of success is 71% for external data.

# 8 Observations

## 8.1 General Discussion of Normality

It was required to separate external data from internal to establish normality of the data samples. A processing delta with a gaussian distribution was applied to the internal data set to replicate the variation in processing time for each call to the services. The external data could not be transformed into a normal distribution; however, a binomial test was used to asses the probability of maintaining within the 500 ms threshold with external data routing uncertainties.

## 8.2 Hypothesis Results

Hypothesis testing using the Student's t-Test and Binomial Test indicates that latency constraints of 500 ms can be maintained internally and external. However, serveral external samples were greater than 500 ms. This is most likely due to the non-deterministic nature of internet (e.g. http) requests. Within the internal environment, data is directly routed between microservices within the Docker environment within a private network. The data shows that a container based microservice architecture can meet the requirement; however, care must be taken to manage processing per container that may increase container response times.

## 8.3 DSS Prototype Environment

The non deterministic nature of the Docker environment on the MacBook laptop significantly affected the ability to assess deterministic behavior. Boxplots of data inclusive of what was sampled from the MacBook clearly depicted this issue. Linux platforms truly run a container as intended; however, non-linux platform require the use of a Linux based Virtual Machine on top of the host OS to implement containers. While the MacBook met the needs for rapid software development, the use of a separate integration and test environment was clearly validated through the collected data.