

5<sup>th</sup> December 2025



# **Building a Reliable R Environment and CSV Readiness for Submissions**

EPS Corporation

Yutaka Morioka, Takashi Uota

# **Agenda**

- 1. Establishment of a Governance Framework**
- 2. Selection of Packages**
- 3. Establishment of CSV Policies for Each Package**
- 4. Validation for Each Package**
- 5. Managing the Status of Package Implementations**

# 1. Establishment of a Governance Framework

## Necessity of a Governance Framework for R Usage

- Statistical analyses submitted to regulatory authorities  
➔ CSV is mandatory
- R has high versatility and extensibility  
➔ Additional considerations are required beyond general CSV governance
- Simply installing “R” does not satisfy the requirements  
➔ Additional package implementation is necessary

# **1. Establishment of a Governance Framework**

## **Specific Considerations in R Governance**

- Requirements are directly linked to package selection**
- If requirements are not clearly identified, R cannot be used for business operations**
- Maintaining and managing the environment requires members with R-specific expertise**

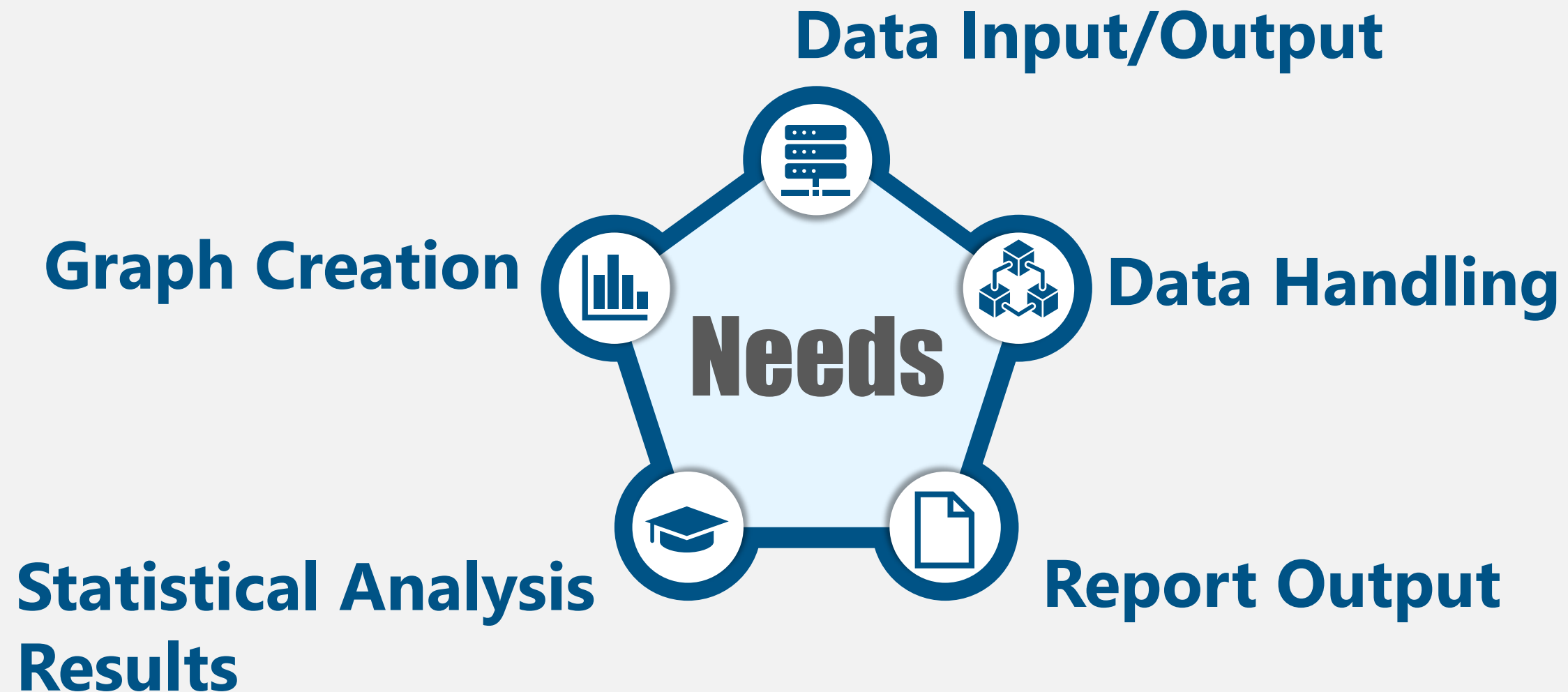
# **1. Establishment of a Governance Framework**

## **The Framework Itself Remains Unchanged**

- Responsibilities of user departments will increase**
- Scope of IT personnel will expand**
- Quality assurance review scope will broaden**
- Validation workload will increase**

## 2. Selection of Packages

**“What to Achieve in the R Environment”**



## 2. Selection of Packages

### Steps in Package Selection Process



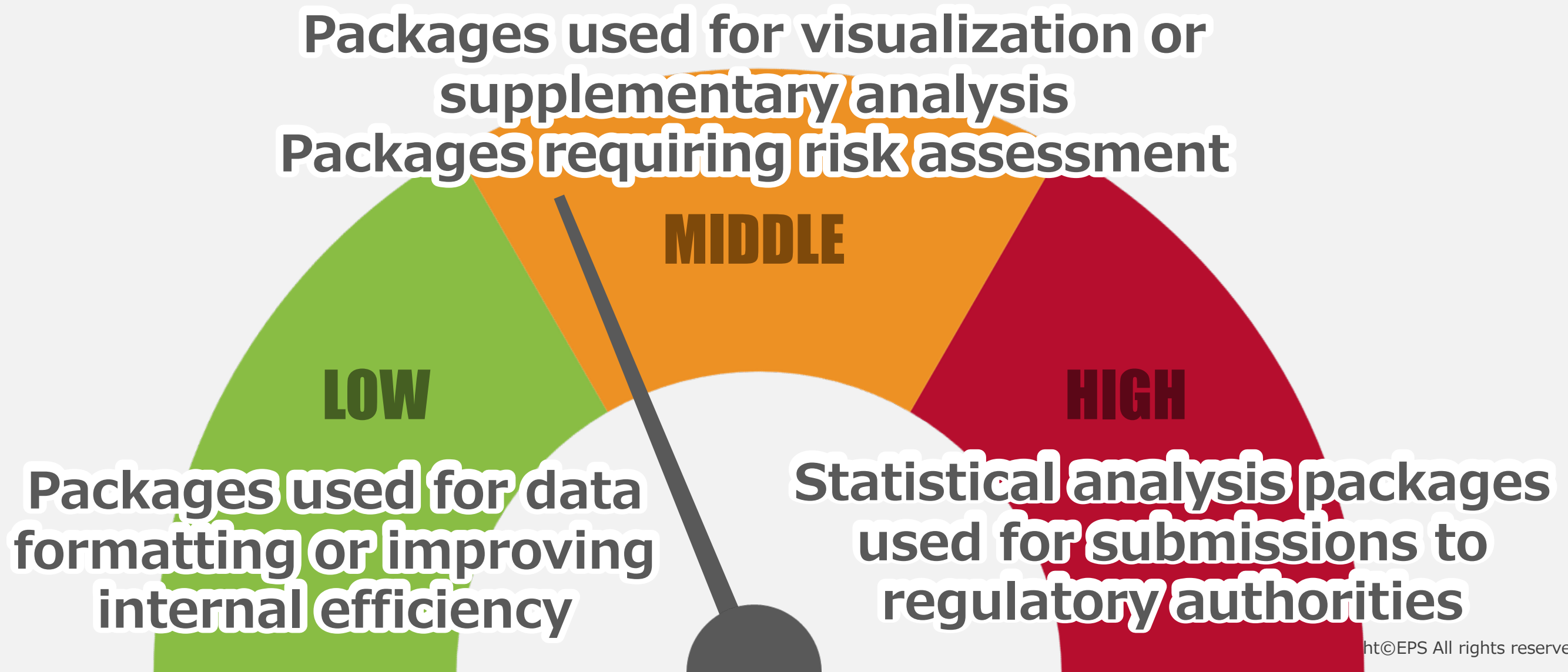
## 2. Selection of Packages

# Package Risk Management Framework

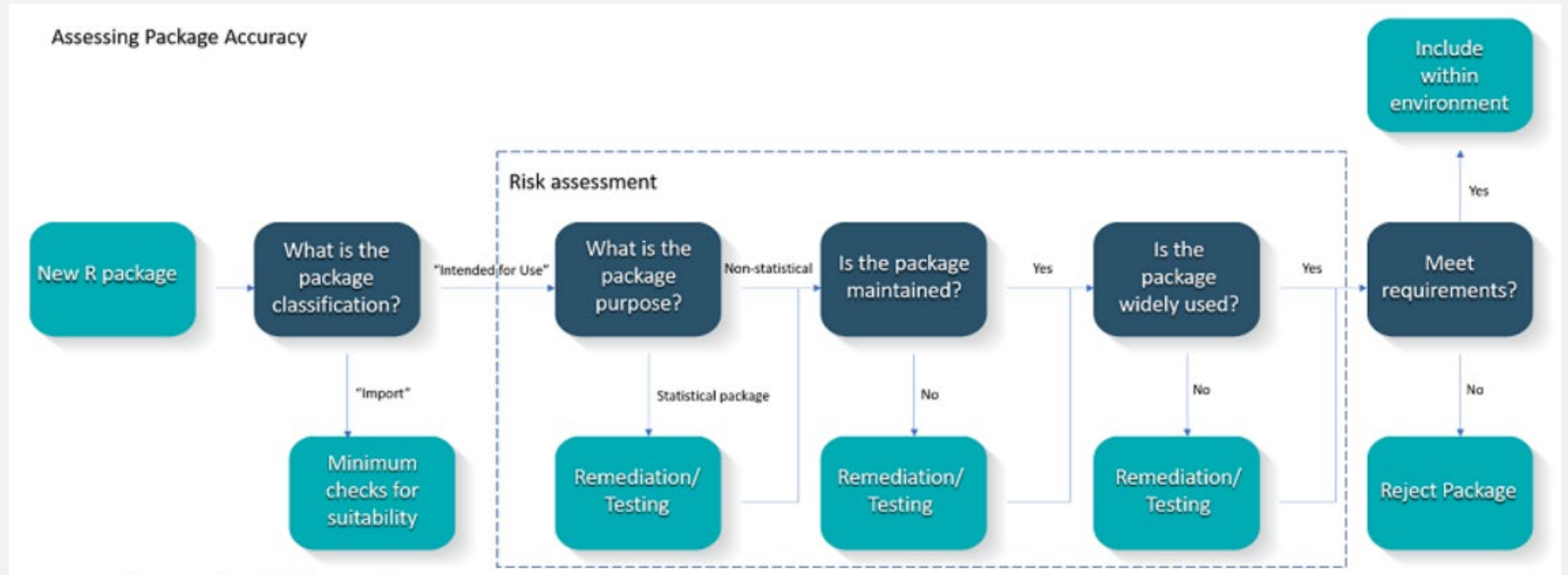


### 3. Establishment of CSV Policies for Each Package

## Example of Classification by Risk



### 3. Establishment of CSV Policies for Each Package



[A Risk-based Approach for Assessing R package Accuracy within a Validated Infrastructure](#)  
R Validation hub, "A RISK-BASED APPROACH FOR ASSESSING R PACKAGE ACCURACY WITHIN A VALIDATED INFRASTRUCTURE"からの引用

## 4. Validation for Each Package

|        | <u>IQ</u> | <u>OQ</u> | <u>PQ</u> |
|--------|-----------|-----------|-----------|
| LOW    | ✓         |           |           |
| MIDDLE | ✓         | ✓         |           |
| HIGH   | ✓         | ✓         | ✓         |

**IQ:** Check that the correct version is installed

**OQ:** Perform functional verification of representative functions with sample data.

**PQ:** Compare with other software, existing reports, or literature to confirm validity

## **4 . Validation for Each Package**

### **Automation and Traceability in Package Validation**

- Script-based validation ensures reproducibility**
- Automated testing with frameworks**
- Reusable test procedures across version updates**
- Validation logs and outputs maintained as CSV audit trails**

# 4. Validation for Each Package

## installed.packages()

### Retrieve Installed Package Information as a Data Frame

|    | A           | B           | C       | D                                                   |
|----|-------------|-------------|---------|-----------------------------------------------------|
| 1  |             | Package     | Version | Depends                                             |
| 2  | abind       | abind       | 1.4-8   | R (>= 1.5.0)                                        |
| 3  | admiral     | admiral     | 1.2.0   | R (>= 4.1)                                          |
| 4  | admiraldev  | admiraldev  | 1.2.0   | R (>= 4.1)                                          |
| 5  | arsenal     | arsenal     | 3.6.3   | R (>= 3.4.0), stats (>= 3.4.0)                      |
| 6  | askpass     | askpass     | 1.2.0   | NA                                                  |
| 7  | assertthat  | assertthat  | 0.2.1   | NA                                                  |
| 8  | backports   | backports   | 1.5.0   | R (>= 3.0.0)                                        |
| 9  | base64enc   | base64enc   | 0.1-3   | R (>= 2.9.0)                                        |
| 10 | BiocManager | BiocManager | 1.30.25 | NA                                                  |
| 11 | bit         | bit         | 4.0.5   | R (>= 2.9.2)                                        |
| 12 | bit64       | bit64       | 4.0.5   | R (>= 3.0.1), bit (>= 4.0.0), utils, methods, stats |
| 13 | blob        | blob        | 1.2.4   | NA                                                  |
| 14 | brew        | brew        | 1.0-10  | NA                                                  |
| 15 | brio        | brio        | 1.1.5   | R (>= 3.6)                                          |
| 16 | broom       | broom       | 1.0.6   | R (>= 3.5)                                          |
| 17 | bslib       | bslib       | 0.8.0   | R (>= 2.10)                                         |
| 18 | cachem      | cachem      | 1.1.0   | NA                                                  |
| 19 | callr       | callr       | 3.7.6   | R (>= 3.4)                                          |
| 20 | car         | car         | 3.1-3   | R (>= 3.5.0), carData (>= 3.0-0)                    |
| 21 | carData     | carData     | 3.0-5   | R (>= 3.5.0)                                        |
| 22 | cellranger  | cellranger  | 1.1.0   | R (>= 3.0.0)                                        |
| 23 | checkmate   | checkmate   | 2.3.2   | R (>= 3.0.0)                                        |

## 4. Validation for Each Package

### test\_dir("Directory for Script")

```
> test_dir("project_for_test/tests/testthat")
```

```
✓ | F W S OK | Context  
✓ |           10 | one-way ANOVA のテスト  
✓ |           6  | functions
```

== Results ==

```
[ FAIL 0 | WARN 0 | SKIP 0 | PASS 16 ]
```

💣 Your tests are da bomb 💣

```
> |
```

## 5. Managing the Status of Package Implementations

### Reasons Why the R Package Environment Is Prone to Instability

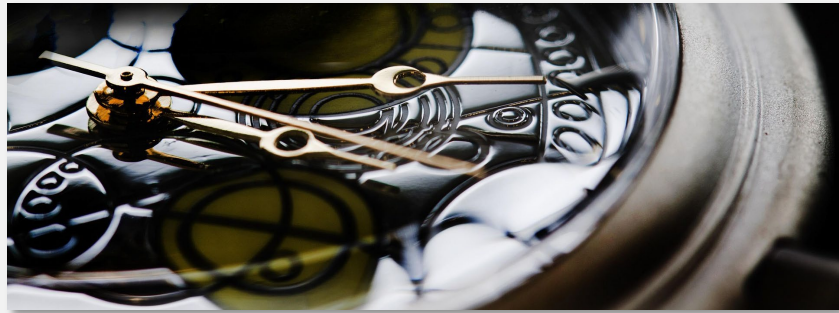
- Complex dependency structures
- Numerous packages in use
- Frequent updates



**Leads directly to quality risks in research and operational activities**

## 5. Managing the Status of Package Implementations

# Dependency Management and Repository Governance



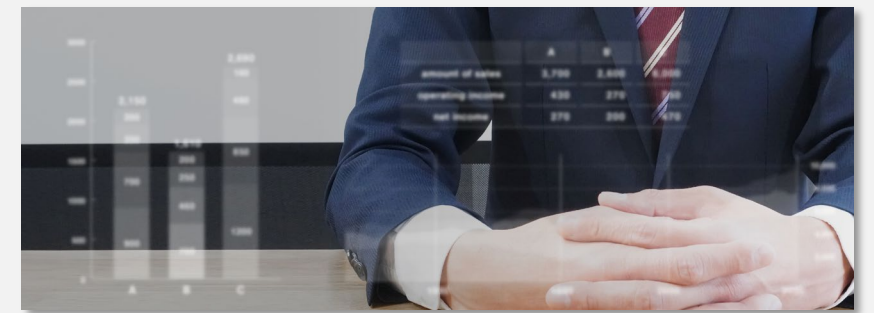
### Reproducibility

- Guarantee identical results for identical inputs
- Behavioral changes due to version differences  
→ Audit risk



### Stability & Security

- Risks from known bugs or vulnerabilities
- Version consistency across the organization is essential



### Auditability & Compliance

- Must record which version was used and when
- Directly linked to GCP compliance

## 5. Managing the Status of Package Implementations

### Dependency Management and Repository Governance

#### ■ Use of tools to lock dependencies at the project level

- ✓ Record the installation status of packages
- ✓ Reproduce the environment

#### ■ Centralized control of package sources

- ✓ Distribute only approved packages
- ✓ Fix by snapshot for date-based versioning → Ensure reproducibility

## 5. Managing the Status of Package Implementations

# Infrastructure Integration and CI/CD Management

- Manage OS-dependent libraries
- Integrate with CI/CD pipelines
- Maintain audit logs and change records

# Summary

## Essential Steps for Implementation

- Establishment of a Governance Framework
- Selection of Packages
- Establishment of CSV Policies for Each Package
- Validation for Each Package
- Managing the Status of Package Implementations

