Paper SM13

AI supported validation of open source software

Sven Prasse, Entimo AG, Berlin, Germany

ABSTRACT

Programming in R for submissions is advancing rapidly. It is easy to find many high-quality specialized R packages contributed to and maintained on open-source platforms (e.g. Pharmaverse, Tidyverse). These packages are usable according to the statistical requirements but not ready for the regulatory requirements because of the lack of computer system validation. The R Validation Hub provides guidelines and tools, but R package validation is a repetitive, resource- and cost-intense company-specific task.

AI is designed to solve repeating tasks based on textual descriptions. Consequently, entimo has embarked on a project using AI to support R package validation to thus save time and resources for clinical trial analysis. The project aims to automate the process for package validation, and it generates a holistic set of guidelines, documents and tools, enabling biometrics groups to perform low-effort, yet risk-based and fully regulatory-compliant R package validations from selection to approval.

INTRODUCTION

The use of R in submissions is becoming more and more common. The R Consortium, a non-profit organization, has exercised a series of five pilot projects together with the FDA to establish a working model for eCTD submissions with R in recent years and shared the progress and learnings with the industry (1). Several pharma companies have successfully submitted clinical trial data and analysis results produced in R to the FDA (2).

The prerequisite to using R in submissions is validation of the R version and all packages used for creation of any deliverables in the submission, according to the regulations of the FDA. Yet the procedure of R package validation is not standardized.

GAMP 5 (Good Automated Manufacturing Practice) defines guidelines for how software must be developed, validated and maintained when used in regulatory compliant clinical research and development. These guidelines are known and followed by many commercial providers for software used in clinical trials. R and R packages are classified as software according to GAMP category 3 and must fulfill the GAMP requirements for computer system validation of this category. Existing communities are maintaining tools to assist in very specific aspects of validation such as risk assessments (3), traceability matrices (4) and testing or providing white papers describing the validation approach for R packages (5). Yet a fully functional toolkit and framework to validate R packages is not available.

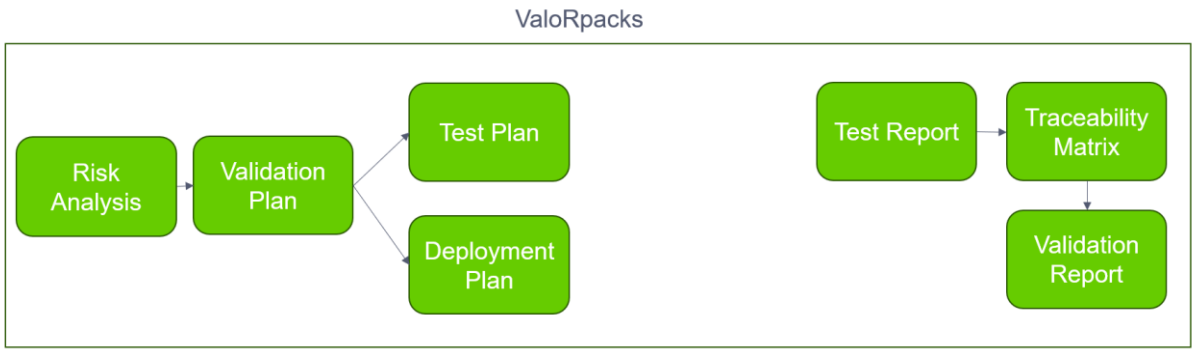
The fast innovation and frequent updates of R are a burden for a manual validation. Entimo is running the ValoRpacks project with the goal of providing the pharmaceutical industry with a self-service framework and toolkit to standardize and automate GxP compliant validation of R packages for biometrics teams of any size, no matter how far the R adoption has progressed. Entimo is aware of this need from many conversations throughout the industry and conducted a thorough market analysis.

This paper reflects the current state of the ValoRpacks research project at entimo and the usage of AI to automate the validation of the R Packages. Design decisions for our product solution may differ from the concepts described in this paper.

AUTOMATED VALIDATION PROCESS

DOCUMENT CREATION CHAIN

The basic concept behind a validation of software according to the GAMP rules is to plan and process all validation steps and the respective documents in a traceable way. The first process step is to create a risk assessment. The output of the risk assessment determines the scope of the downstream process steps, which is documented in the validation plan document. The package test plan is derived from the validation plan and describes in detail all testcases that have to be executed. For the test execution the deployment of the R package, including all dependencies, has to be planed. The execution is documented in the package test report. Over all steps a traceability matrix has to be created that shows the relation between the text cases and their results, with the functional specifications and the implemented user requirements.

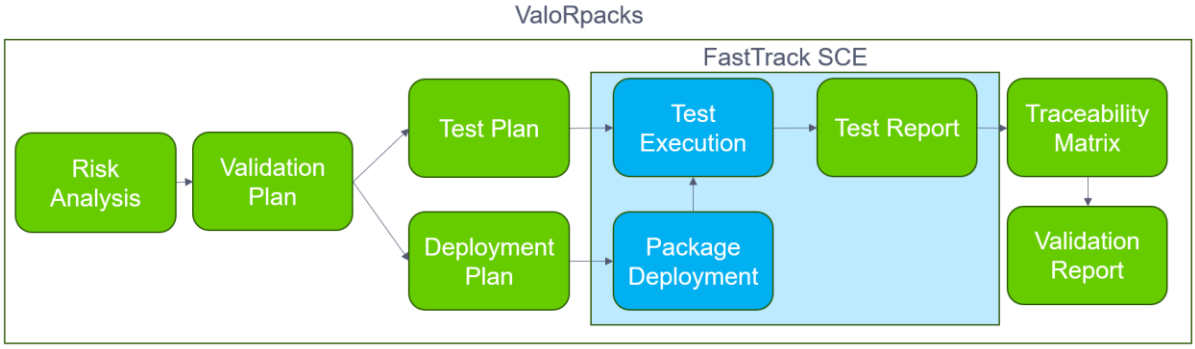


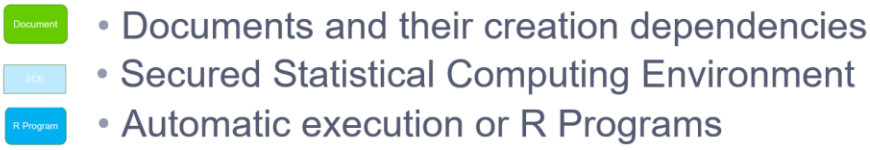
ARCHIVING OF VALIDATION INPUT ARTEFACTS

The user initiates the validation of one or more R packages and has to specify the scope and use for the selected packages, and define restrictions on the page use if necessary. ValoRpacks utilizes one or multiple separate R package repositories to maintain independent copies of the R packages which are subject to validation and retrieved from publicly available repositories, such as CRAN. In this repository it is granted that the source and package contents from validation of all versions of the package are stored and archived for regulatory reasons.

AUTOMATED TEST EXECUTION

For the execution of the test cases standard R testing packages testthat is used. And lintr (7) can be utilized for whitebox testing by analyzing the source code and identifying security issues, potential errors and derivations from a coding standard. The key to achieve regulatory compliance is the use of an execution environment that satisfies the requirements of the GAMP rules and ensures that correct versions are installed and used.





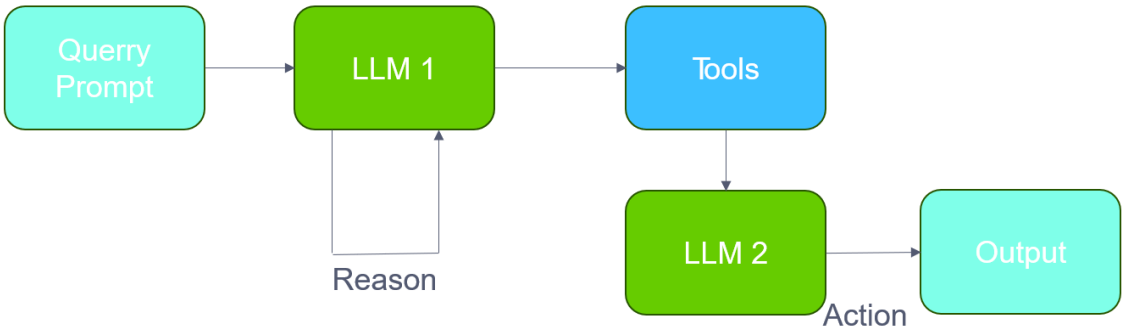
To plan, document and automatically execute this ValoRpacks creates a Deployment plan and a matching package deployment as an executable script.

AI SUPPORTED DOCUMENT GENERATION

REACT AGENTS

ReAct agents are AI systems that combine reasoning and action in a continuous, iterative loop to solve complex tasks. Inspired by human problem-solving, they use a large language model (LLM) as a "brain" to think step-by-step, decide on actions, execute them via tools (like search engines or APIs), and then observe results to refine their next

steps.

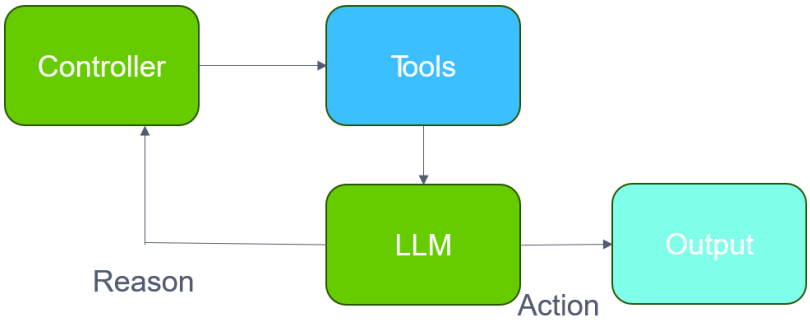


The ReAct agent's core component is an LLM as the central reasoning engine, generating thoughts and decisions. It receives a prompt input and breaks it down, and plans different execution steps, chooses tools to retrieve with the final action the requested output.

The key advantage of ReAct agents is the ability of dynamic problem solving and is designed to resolve multi-step tasks like research, planning or troubleshooting. The solution requires powerful LLMs for optimal performance.

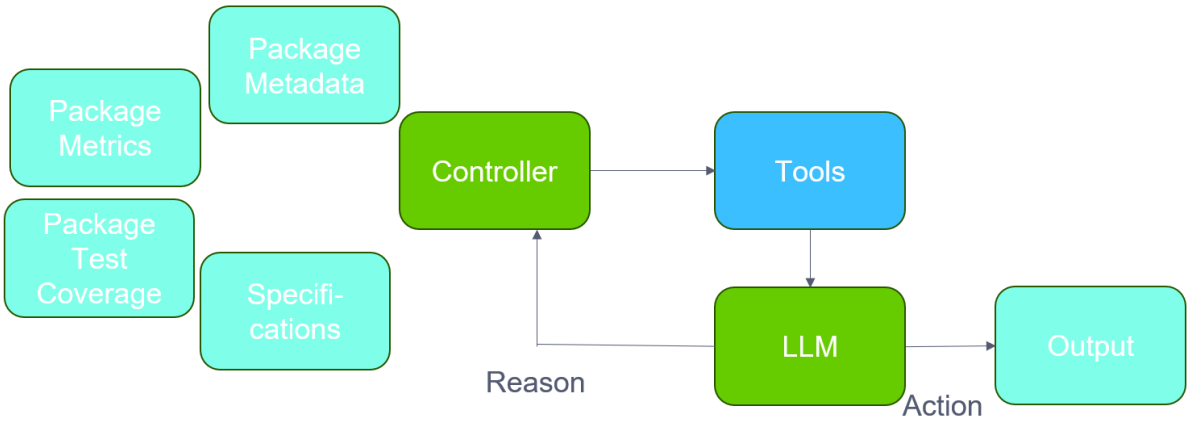
REACT AI MODIFICATION

To support the automatic process, the prompt and LLM combination will be replaced by an algorithm to control the LLM. The controller analyzes the output of the LLM to keep the flexibility and dynamic of the ReAct agent. The controller algorithm determines the functions of a package or the parameter list of a function by using different publicly available R packages, official resources and public tools. The controller replaces the prompting for an LLM by automated retrieved information and presenting action and observation steps to the LLM, retrieving the outputs and combining them to an output document in natural language or using it for the next iteration as an input.



The input for the starting iteration has to be defined without the help from AI. As the starting point, the R package documentation and tools analyzing the source code of the package are available and can be used to generate the input for the first iteration. The package metadata can be analyzed with the R packages pkgnet, pkdown, or roxygen and the test coverage be calculated with covtracer. In addition to the code base and package documentation the package metrics from CRAN, github and other sources are used for the risk assessment. Examples include the commits, release frequence releases, download counts and bug statistics which can be used to evaluate risks for a specific package.
Most of the information can be generated by open source and de facto standard programs, and the rest is gathered by ValoRpacks' integrated tools. All the information is also used in the next steps of the validation process together

with output of the LLM to produce also all other artifacts.



LIMITATIONS

The validation of R packages with ValoRpacks is following the GAMP 5 guidelines for non-configurable software of GAMP category 3 by the book. The validation is driven by a risk assessment and produces comprehensive and traceable validation artifacts.

Even then, the regulation needs a human review cycle and approval at least on the specification and risk assessment level. Based on these documents ValoRpacks generates a Validation Plan, Package Test Plan and Package Deployment Plan based on standard templates. Testthat (6), Lintr (7) and available tests will be utilized for the package test coverage. In case the risk assessment demands further tests for specific package functions, these will be generated by ValoRpacks and reviewed by humans. Again, LLMs assist in natural language text generation where needed.

Since the selected R packages have to be validated in conjunction with a validated target Statistical Computing Environment (SCE), the deployment and test execution have to be performed on this target system. The level of automation of deployment and test execution depends on the type of integration with the SCE. Deployment and test execution generate reports which will be returned to the ValoRpacks framework. Based on these reports, the specification and risk assessment ValoRpacks generates a holistic Traceability Matrix, and the Validation Report based on further standard templates. ValoRpacks also generates a standardized usage instruction document based on available package documentation in conjunction with the generated specification, according to the intended scope and use for the R package.

CONCLUSION

Current AI application models are highly focused and geared toward human communication. This opens up new possibilities for automation that previously failed due to the inability to interpret natural language and produce appropriate results. In combination with conventional algorithms, there is great untapped potential for cost savings.
The focus of major AI providers on comprehensive AI models capable of everyday communication with humans makes it difficult to develop appropriate automated solutions without using prompting. The use of ReAct Agents fills the gap between the process controller and the heavy usage of various R tools to fully automate the system.
This concept enables ValoRpacks to offer a unique, automated approach to master the validation of R packages.
The framework empowers companies to take full control of the automated validation process and the management of validated R packages and repositories inside their own individual landscape. The high level of automation paired with a sensible level of human oversight enables companies to validate, revalidate and utilize R packages with low cost and low effort.
entimo is planning to release ValoRpacks to the industry in 2026 and looking for partnerships with interested parties and early adopters. For us this is a proof that AI can automate complex problems in an engineering context.

REFERENCES

(1) https://rconsortium.github.io/submissions-wg/pilot_background.html (last accessed 08.01.2026)

(2) https://phuse.s3.eu-central-1.amazonaws.com/Archive/2024/Connect/EU/Strasbourg/PAP_SA02.pdf | https://posit.co/webinar/novo-nordisk-journey-to-an-r-based-fda-submission/ | https://r-consortium.org/posts/johnson-and-johnsons-success-story-a-hybrid-r-sas-strategy-for-fda-submission/ (last accessed 08.01.2026)

(3) https://pharmar.org/riskassessment/ (last accessed 08.01.2026)

(4) https://github.com/genentech/covtracer (last accessed 08.01.2026)

(5) https://pharmar.org/white-paper/ (last accessed 08.01.2026)

(6) https://testthat.r-lib.org/ (last accessed 08.01.2026)

(7) https://lintr.r-lib.org/ (last accessed 08.01.2026)

(8) https://www.plainconcepts.com/react-agent-ai/ (last accessed 30.01.2026)