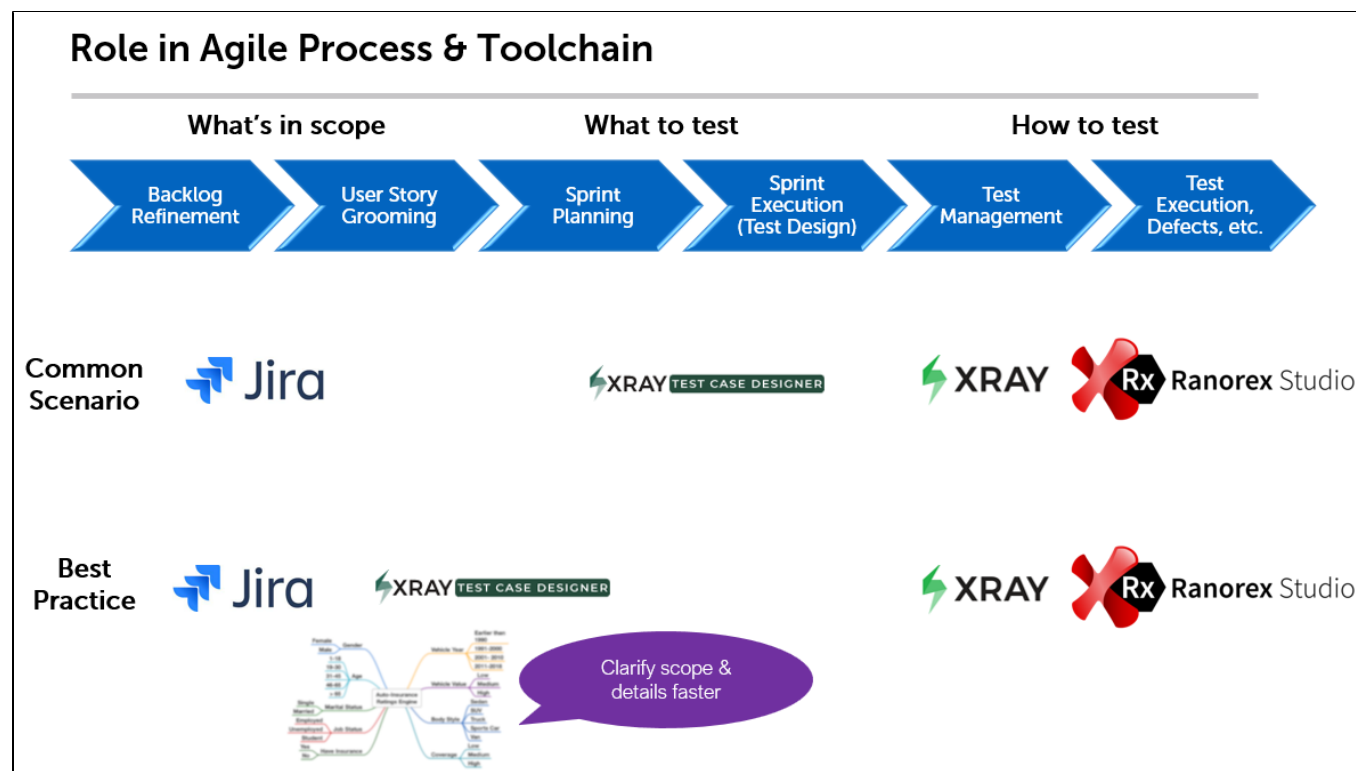


# Test Case Designer Role in the Toolchain

Keeping in mind that our integration capabilities are constantly evolving, we can consider the following simplified toolchain as the status quo for this discussion:



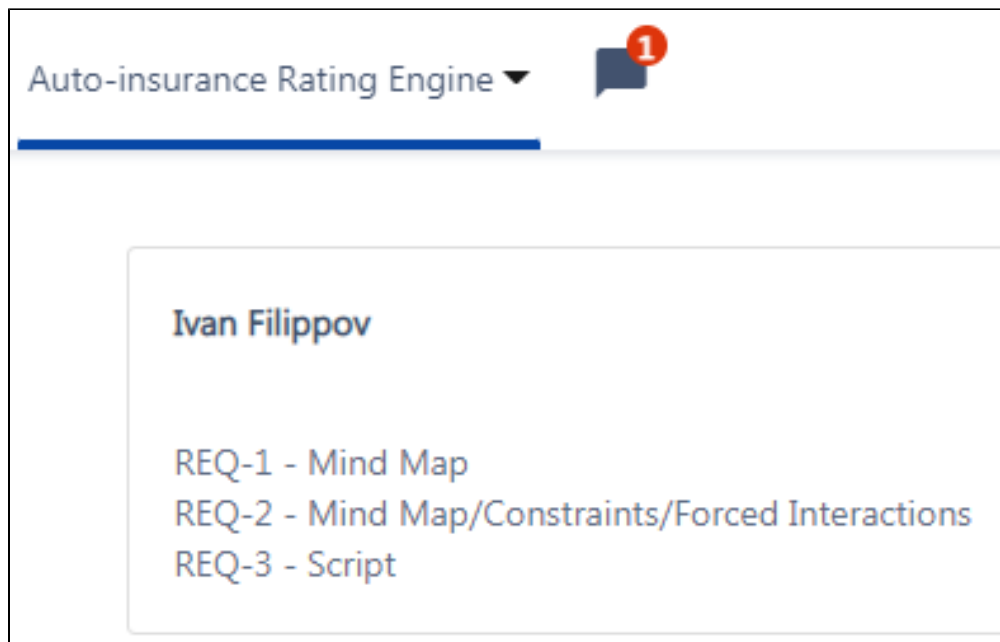
Specific tools around Test Case Designer in the image above are mentioned for example purposes and do not represent the exhaustive list.

Let's look at both integration sides from Test Case Designer's perspective in more detail.

- [Left Side](#)
  - How is "Best Practice" different?
- [Right Side](#)
- [How to deal with changes](#)

## Left Side

The first part is to include traceability notes inside the Test Case Designer model identifying which part of the model is affected by which story (there are largely 3 categories you can see on the image below).



Additional traceability elements include the Mind Maps (you can put the REQ ID in the value name), Forced Interactions, and in-line/per-tab comments in Scripts.

```
Scenario: Sample Script
  Given sample step #REQ-3
  When some action is taken
  Then some verifiable outcome
```

These steps reduce redundancy risk across the test suite, given the common one-to-many relationships between tests and requirements.

## How is “Best Practice” different?

**Potential improvement #1:** Write story descriptions in Gherkin. That would improve the clarity and communication and provide the kickstart to scripting in Test Case Designer.

**Potential improvement #2:** Group parameters & values together in their own part of the story (similar to “Description” or “Acceptance Criteria”), in the mind map format (for visual analysis) and/or in the parameter table one (for direct import into Test Case Designer). That speeds up the process of reading through paragraphs of text to identify the key model elements.

**Potential improvement #3:** Include the “mini-scenarios table” that covers the most important interactions from the story (which can also be imported directly into Test Case Designer). It can help clarify ambiguities in complex rules much sooner.

For example,

**Before:** "If the named insured and the driver don't match, ask the extra question"

**After:**

PNI	SNI	Driver 1	Driver 2	Question is displayed
Ivan	no SNI	Ivan	no driver 2	No
Ivan	no SNI	Jackson	Ivan	No
Ivan	no SNI	Ivan	Jackson	No
Ivan	no SNI	Jackson	no driver 2	Yes
Ivan	no SNI	Jackson	Justin	Yes
Ivan	no SNI	Jackson	Conor	Yes
Ivan	Conor	Ivan	Jackson	No
Ivan	Conor	Conor	no driver 2	No
Ivan	Conor	Jackson	Ivan	No
Ivan	Conor	Jackson	Justin	Yes


**Potential improvement #4:** Add "Test Case Designer" label and relationship link type "shares assets with". Displaying both labels and links in the sprint board view will provide the quick view into possible connections and increase collaboration efficiency.

## Right Side


The primary method is syncing or exporting from Test Case Designer into Xray. Test Case Designer also supports export into the formats directly compatible with the execution tools at the last stage of the toolchain (like Ranorex Studio).

Export


To get started, click the target for your exported test model.




Automated Test Framework




Automate Gherkin Scripts




Automate Robot Framework Scripts




Azure DevOps




Comma-separated values



Excel



Mind Map



Xray

## How to deal with changes

Once the Test Case Designer model is built, the story link + traceability notes will map the requirement updates and let the user know which parts to change. The interaction coverage can be focused on the new areas via Mixed-strength and Parameter structure (e.g., more Value Expansions for less important parameters).

TCs in Jira/Xray would be updated via re-sync/re-import, either for the full scope or only for the new elements using Freeze/Script features in Test Case Designer.

If the execution history must be preserved, the sync/import is typically either performed into a new repository folder/Xray Test Plan or by overwriting the existing Issue IDs. The archiving function could also be considered.

If it doesn't have to be preserved, the existing tests can just be deleted and the new ones synced/imported.