

★★★★ Day 114 ★★★★★  
of Our Daily UiPath Q&A  
Challenge!

Welcome to Day 114 of our  
exciting UiPath Q&A Challenge!  
Each day, we'll be answering one  
key question to help you master  
UiPath and revolutionize your  
automation journey. 🚀

🚫 Question for Today

How can you modify  
RE-Framework to work with  
non-queue data sources?

Modifying the REFramework to work with non-queue data sources (like Excel, databases, CSV files, APIs, etc.) is a common customization. Here's a step-by-step guide to help you adapt it:

## 1. Disable Orchestrator Queue Dependency

In the default REFramework, transactions are pulled from an Orchestrator queue. To use a different data source:

- Open GetTransactionData.xaml.
- Replace the logic that gets a queue item with logic to fetch a data row, list item, or record from your source.
- `out_TransactionItem=dt_InputData.Rows(in_TransactionNumber - 1)`

## 2. Load Data in Init State

In Init.xaml or InitAllApplications.xaml:

- Load your data source (e.g., read an Excel file or query a database).
- Store the data in a global variable (e.g., a DataTable or List) in the Main.xaml file using a DataTable or List<T> argument.

## 3. Track Transaction Index

Since you're not using a queue, you need to manually track which transaction you're processing:

- In Main.xaml, create a variable like `transactionIndex` (type `Int32`, default 0).
- In `GetTransactionData.xaml`, use this index to fetch the current item from your data source.
- Increment the index after each transaction.

## 4. Modify TransactionItem Type

By default, TransactionItem is of type QueueItem.

Change it to match your data source:

Change the type of TransactionItem in Main.xaml to DataRow, String, or a custom class depending on your data.

Update all references to TransactionItem in Process.xaml, SetTransactionStatus.xaml, etc., accordingly.

## 5. Handle Business and System Exceptions

The exception handling logic remains mostly the same. However:

- BusinessRuleException: Skip to the next item.
- System Exception: You can retry the same item or log and continue, depending on your needs
- Retry can be defined in Config.xlsx file. Update MaxRetryNumber in file