Automation Metrics

Background

Inaccurate or poorly performing automations can undermine trust within organizations. Bearing this in mind, an undeniable reality in the realm of building automations is that their performance will inevitably change over time. A multitude of variables, including graphical user interface (GUI) alterations, enterprise software upgrades, server traffic/capacity, and more, all exert influence on the performance of an automation. The question arises: What can be done?

As is the case with most operational challenges, one cannot address what is not measured.

“Measurement is the first step that leads to control and eventually to improvement. If you can’t measure something, you can’t understand it. If you can’t understand it, you can’t control it. If you can’t control it, you can’t improve it”.

- H. James Harrington

Overview

This guide can serve as a dictionary and checklist for building your organization’s automation dashboard. We’ve outlined the must-have and nice-to-have metrics to track for every automation. In addition, we’ve shared two statistical techniques that will identify and diagnose issues. With metrics and techniques established, it's easy to measure the health of each automation and prioritize maintenance activity.

The Metrics

The must-have automation metrics			
Metric	Definition	Formula	Comments
Skip Rate	This metric measures the percentage of skipped accounts compared to the total number of queued accounts. A skip is any instance the automation chooses to not process an account due to a lack of information. If the automation starts the process and subsequently fails, then it should be calculated in the Error Rate.	**Numerator:** Skipped accounts **Denominator:** Total number of accounts queued	A lower skip rate indicates better business rules, data quality, and report writing.
Error Rate	This metric measures the percentage of accounts that are unsuccessful compared to the total number of accounts processed. A processed account is one that the automation has started (i.e., not skipped due to a lack of information). This excludes skipped accounts. An unsuccessful account is any instance where the automation doesn't finish the final step. This excludes skipped accounts.	**Numerator:** Unsuccessful accounts **Denominator:** Total number of accounts processed	A lower Error Rate indicates better automation design and build. Tarpon recommends an error rate less than 5%.
Average Automation Execution Time	This metric measures the average time it takes to process an account.	**Numerator:** (Account 1 execution time) + (Account 2 execution time) + (*Account n execution time)* **Denominator:** Number of accounts processed	Calculate for each automation run. Averages are highly influenced by outliers. Outliers, especially those that are greater than the average, should be investigated for improvement opportunities.

The must-have automation metrics			
Metric	Definition	Formula	Comments
Median Automation Execution Time	This metric measures the median time it takes to process an account.	The median is the middle Execution Time when times are organized from small to large across a given period of time	Calculate for each automation run. The median is less susceptible to outliers. Variances in the Average Execution Time and the Median Execution Time will present data skews (high or low) that should be investigated.
Queue Consumption Rate	This metric measures the percentage of accounts processed out of the total number of accounts queued. This measures an automations ability to consume its anticipated queue within its designated run time. Run times are based on coded schedules and anticipated automation volume and execution time.	**Numerator:** Accounts processed by the automation **Denominator:** Accounts queued by the automation	Calculate for each automation run.
Error Free Days	This metric measures the percentage of days that the automation runs within an acceptable error rate. Tarpon recommends an acceptable error rate being equal or lower than 5%.	**Numerator:** Number of error free days **Denominator:** Number of days the automation processed at least 1 account	A higher number of error free days indicate a more stable and high-quality automation. Defining what counts as an error will be unique to your organization and automation.

The nice-to-have automation metrics			
Metric	Definition	Formula	Comments
External Error Rate	This metric measures the percentage of inaccurately processed transactions due to external reasons (e.g., application/portal not being available) compared to the total number of accounts processed. External is generally defined as an issue that you cannot resolve with your internal resources.	**Numerator:** Unsuccessful accounts flagged as an external failure log **Denominator:** Total number of accounts processed	A lower error rate indicates better accuracy. Knowing the cause of internal and external failures allows for a more precise maintenance approach. Different logs must be developed for internal and external failures which takes effort when building.
Branch Error Rate	This metric measures the percentage of inaccurately processed transactions for each unique branch of an automation. A branch is a series of steps that leads to a distinct outcome.	**Numerator:** Unsuccessful accounts using branch n **Denominator** Total accounts using branch n	Knowing the error rate for each branch allows for a more precise maintenance approach. However, it will take more developer/analytics which requires meaningful effort.
Average Automation Branch Execution Time	This metric measures the average time it takes a unique branch of an automation to process. The average must be completed for each branch of the automation. A branch is a series of steps that leads to a distinct outcome.	**Numerator:** (Account 1, branch n execution time) + (Account 2, branch n execution time) + (*Account n, branch n execution time*) **Denominator:** Number of branch n accounts	Knowing the average time for each branch allows for a more precise maintenance approach. However, it will take more developer/analytics which requires meaningful effort
Downtime Rate	This metric measures the percentage of downtime as a portion of scheduled time. Downtime can be for any reason, whether it's internal or external. This is only applicable when automations are not dynamically scheduled.	**Numerator:** Hours the bot has been scheduled to run, but did not **Denominator:** Hours the bot has been scheduled to run	A lower rate indicates a more stable automation This metric can be calculated any time measurement. An alternative method is needed if using dynamic scheduling.

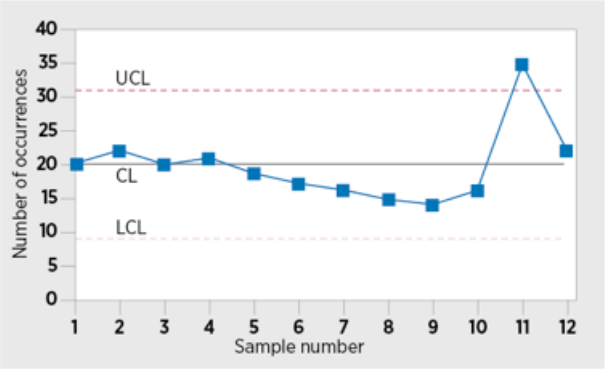
Other Techniques

Statistical process control (SPC) is defined as the use of statistical techniques to control a process or production method. SPC tools and procedures can help you monitor process behavior, discover issues in internal systems, and find solutions for production issues. Organization's should adjust expectations depending on the phase of the automation (i.e., is the automation new, steady-state, or in regression). Tarpon recommends using two of the common SPC tools: the Control Chart and the Histogram. For more information on SPC tools visit ASQ.

Control Chart

The control chart is a graph used to study how a process changes over time. Data is plotted in time order. A control chart always has a central line for the average, an upper line for the upper control limit, and a lower line for the lower control limit. These lines are determined from historical data. By comparing current data to these lines, you can draw conclusions about whether the process variation is consistent (in control) or is unpredictable (out of control, affected by special causes of variation).

Tarpon recommends using a Control Chart for Average Automation Execution Time and Average Automation Branch Execution Time. The baseline should be established using a minimum of 1 month's worth of data (assuming the automation is running most days). Instances where the chart exceeds the Upper or Lower Control Limits should be investigated for improvement opportunities.

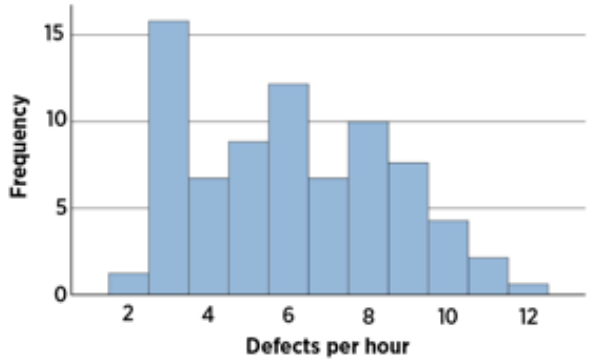


Control Chart Example

Histograms

A histogram is an approximate representation of the distribution of numerical data. It is a type of bar chart that shows the frequency or number of observations within different numerical ranges, called bins. The bins are usually specified as consecutive, non-overlapping intervals of a variable. The histogram provides a visual representation of the distribution of the data, showing the number of observations that fall within each bin. This can be useful for identifying patterns and trends in the data, and for making comparisons between different datasets.

Tarpon recommends plotting the Average Automation Execution Time and Error Rate for each automation run over a select period of time on a histogram. This will allow you to observe the distribution of data and also understand the frequency of automation runs falling outside acceptable levels. Variances are commonly associated with glitches, errors, or loss of functionality within the automation, and can be investigated for improvement opportunities.



Histogram Example