

# THE BEGINNER'S GUIDE TO AUTOMATION TRIGGERS

The First Step in Business Automation

# Table of Contents



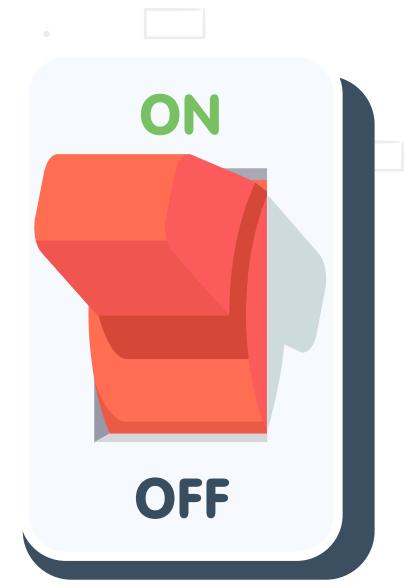
# What's an automation trigger?

An **automation trigger** is the initial input that kicks off the activation of an automated workflow. In a sense, an automation trigger is, quite literally, the first step in automation. Automation triggers can kick off an entire automated process in response to a single stimulus, including:

- Web-based input - In response to a user interacting with your company's website
- Email-based input - In response to emailing a specific address
- Manual input - In response to manually triggering an automation
- Application-based input - In response to a change in your business software

To help better understand what automation triggers (and automated workflows) are, why they're important, and how you can master them, it might be helpful to step back and define a few high-level concepts:

- Automation (for business processes)
- Integrations / API integrations
- Examples, and benefits, of automation
- Automated workflows



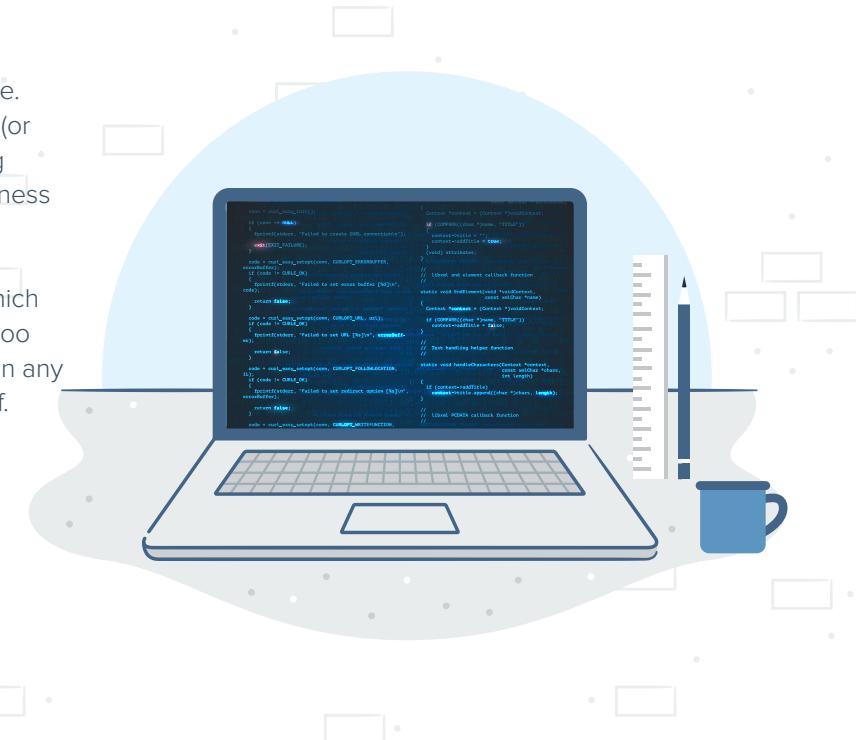
# What is automation?

## (for business processes)

**Automation** is, at its core, the use of equipment, processes, or systems to perform tasks that might otherwise have been attempted manually. But with modern-day software automation, entire organizations are taking advantage of application programming interfaces (APIs) and serverless computing to innovate their processes and rapidly grow their businesses.

Confusingly, you've probably seen numerous articles, some controversial, about **robotic process automation**, which uses software robots or artificial intelligence (AI) to do the work of humans, and how it might affect the workforce of the future.

To clarify, we're not talking about that here. We're talking about **process automation** (or simply "automation" as we'll call it moving forward), the optimizing of important business processes through software. In practice, automation of this sort uses software to perform time-sensitive business tasks, which would either take too long or are simply too big or data-intensive to handle manually in any reasonable period of time, on your behalf.

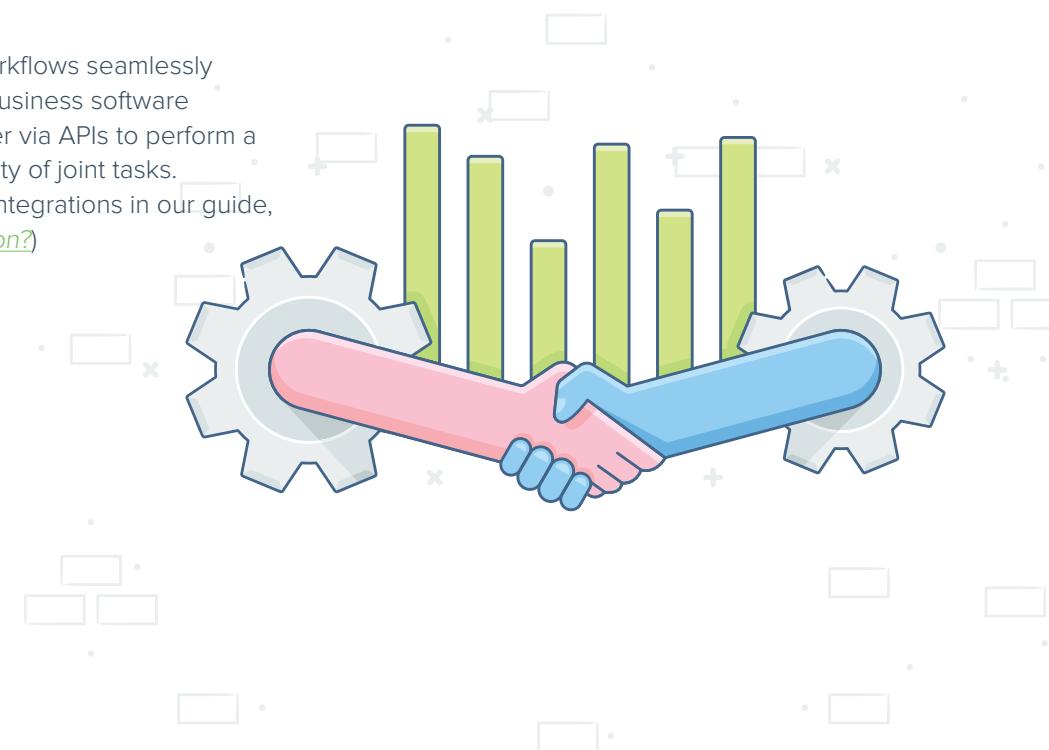


# Automation: Hand-in-hand with API integrations

Automation of this sort typically requires **integrations** - connections between the major software components of your technology stack (your marketing platform, CRM, task management, and others) that enable them to talk to each other and share important data, such as lead lists and customer invoices.

Modern automation is usually a function of **API integrations** - integrations that take place at the software API layer - a layer that acts as a shipping container, standardized specifically to deliver software. API integrations are the most modern and painless way to get different software solutions to talk to each other, and nearly all modern software solutions use APIs for this exact purpose.

The best automated workflows seamlessly integrate your various business software solutions with each other via APIs to perform a practically limitless variety of joint tasks. (Learn more about API integrations in our guide, [What Is an API Integration?](#))



# Automated processes and their benefits

Here are just a few examples of processes that make sense for automation:

- **Marketing: Sync lead records from MAP integrated with CRM**

Matching the thousands of lead records your company gathered in its marketing automation platform (MAP) to your customer relationship management (CRM) solution is an important task that marketers must complete quickly after capturing new or warmed-up existing leads from a recent marketing program.

- **Sales: Manage and tracking outbound email campaigns**

Many sales organizations use an outbound email automation platform to prospect for new leads. Companies sending thousands of outbound touches integrate their email automation with their CRM to log which prospects opened the emails, which didn't, which replied positively, and which requested an unsubscribe.

- **Customer success: Rapidly onboard new customers**

The onboarding process for new customers is most successful when it has all the info and context from the sales cycle, including use case details and any specific customer needs. By syncing sales intel recorded in CRM to helpdesk software and alerting apps, support teams can rapidly get new customers up to speed.

- **Finance: Route invoices from payment processing to accounting software.**

Companies process thousands of customer invoices every month, and must then transfer the payment information to their accounting system to record which customers have paid, which are overdue, which are churning, and which are upgrading to a higher-priced package.

These are big tasks that require flowing significant amounts of data from one place to another, and at larger-scale enterprise companies, these big jobs only get bigger. Processes like these would be completely unreasonable to attempt manually, and would definitely introduce tons of human error in the process.

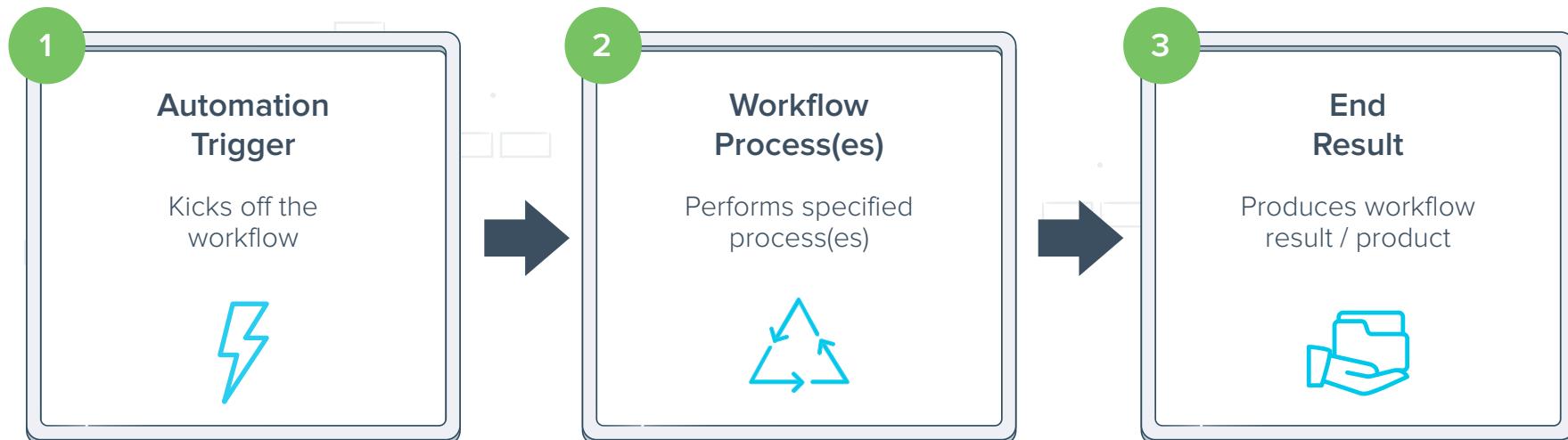
Time-sensitive, data-heavy processes that require flawless accuracy are exactly the kind of business processes that should be automated by software, so busy professionals can focus on being strategic about their business in the meantime. (Learn more about business use cases for automation in [The Beginner's Guide to General Automation Platforms.](#))

What's the benefit of streamlining business processes with automation? [Research firm Aberdeen Group](#) found that organizations that take advantage of smart process automation capture 2.2x more revenue, 2.9x higher profits, have deals that average 9x higher, and see exponentially higher customer retention than those that don't.



# OK. So what's an "automated workflow"?

An **automated workflow** is effectively a series of automated steps that carry out a business process, such as the ones described above. In the absolute simplest terms, most automated workflows contain these steps:



The initial stimulus that initiates the subsequent steps of the process.

The actual “work” of the workflow - whether that be lead tracking from your MAP, updating sales prospects in CRM, or any other process or combination of processes that involve validating, transforming, and/or flowing data from point A to point B.

The final product of the workflow, such as an updated database, or a properly-formatted CSV of important business data.

# So why are automation triggers important? What should you look for?

Automation triggers are important because they act as the stimulus-response step to begin your automated workflows.

You also need automation triggers that can handle your specific needs. You won't get far with automation triggers that frequently fail, or can only initiate workflows from a very narrow set of inputs.

To make sure you get the most from your automation, you need automation triggers that are:

- **Reliable**

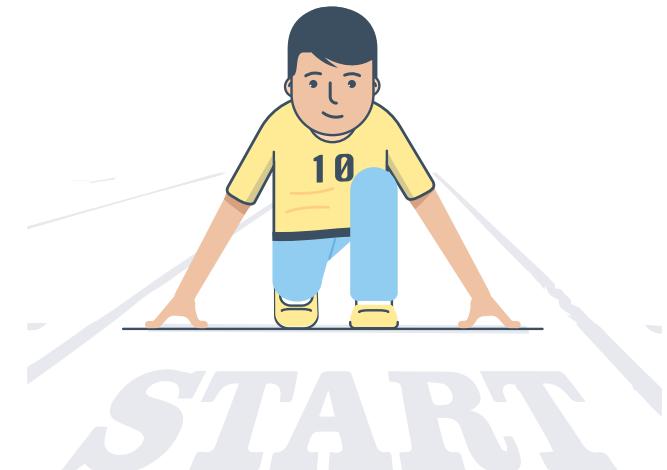
One of the most important benefits of automated workflows is their ability to proceed without having to babysit them manually. To get the full value of automation, you need workflows that dependably kick off their processes every time they encounter a valid trigger. (This also raises the issue of fail-safe measures for workflows in case of errors, as well as alerting - we'll discuss these later.)

- **Varied**

The more types of automation triggers you have at your disposal, the better. Good automation platforms provide multiple avenues to kick off workflows, not simply including manual triggers, but also including the ability to initiate processes according to a pre-set schedule; or the ability to start a workflow on receiving an email input; or in response to a marketing form fill...among others. Best-in-class automation platforms even include a variety of app-specific triggers that you can use to trigger automated workflows in response to changes in your CRM, marketing platform, outbound email automation, and more.

- **Easy to use / modify**

One of the most important aspects of automated workflows is working in an environment that is flexible enough to allow for tinkering and iterative upgrades as your tech stack changes. In the same way, it's important to have the ability to quickly swap different automation triggers into your workflows to better customize them over time to your company's changing needs.



# What kind of automation triggers are there? And how can you master them?

Below is a list of automation triggers for you to familiarize yourself with, as well as sample use cases with suggested automated workflows to accompany them.

**PLEASE NOTE:** For the sake of example, many of these workflows have been highly simplified and streamlined into their most basic components. A full-featured General Automation Platform is flexible enough to offer a great many more enhancements and potential workflow steps to help you fully customize your workflows to your specific needs, including features such as:

- Boolean logic - If/then statements that can fork workflows depending on whether data meets or doesn't meet certain criteria
- Object helpers - Open-ended function to retrieve, compare, merge, or otherwise interact with specific data-value objects
- Data storage helpers - Function to temporarily store relevant data for later steps
- Loop operators - Function to iteratively loop certain workflow steps until all data in question is complete - such as enriching each lead in every row of a CSV file
- CSV processor - Function to ingest, format, or export data contained in a CSV file
- List helpers - Function to list relevant data values to add to your workflow
- Script helpers - Functions to let you directly add scripted commands as needed

## Manual triggers

Manual triggers use a manual input to begin the associated workflow. They give you full control of exactly when a specific automated workflow begins - in this case, not until you click the "Start" button. While this may seem counterintuitive for functionality that's supposed to handle processes for you, manual triggers have a few important uses:

### Some potential use cases:

- **Testing purposes:** To put a new workflow through its paces, it's common to use manual triggers to repeatedly re-run a workflow until its results match up to expectations.
- **One-off processes:** Manual triggers also make sense for handling one-off processes that don't necessarily take place on a regular schedule, such as emailing leads an invitation to any trade shows your company attends.



Example: Send automated trade show invite email, update CRM

**Let's break down the steps in this automated workflow:**

**1. Manual trigger:** Use a manual trigger to begin this automated workflow. Since this pertains to specific trade shows that may occur at irregular dates, a manual workflow makes sense, as you can set it to run on-demand, whenever you need.

**2. Get leads from marketing automation platform:** After the trigger fires, the workflow integrates the marketing automation platform (in this case, Eloqua) to pull up a list of all relevant leads for this trade show.

**3. Outbound email:** The workflow then routes those leads to the outbound email automation platform (in this case, Outreach) to send an automated invite to your company's booth by email.

**4. Update CRM:** This automated workflow concludes by updating the records of every lead in your CRM (in this case, Salesforce) to reflect that each has received an email invite to your booth.

**NOTE:** With a General Automation Platform, you can extend the functionality of this workflow greatly, such as by:

- Adding corroborating loops to ensure lead info is fully accurate by also integrating your lead enrichment solution to flesh out each lead's entry
- Add follow-up functions for your sales team, such as automated alerts via internal chat to personally follow-up with high-priority contacts from strategic accounts
- Add follow-up functions for your field marketing team, such as adding confirmed email replies to a scheduled appointment list or tallying orders for "swag" gift items

## Scheduled triggers

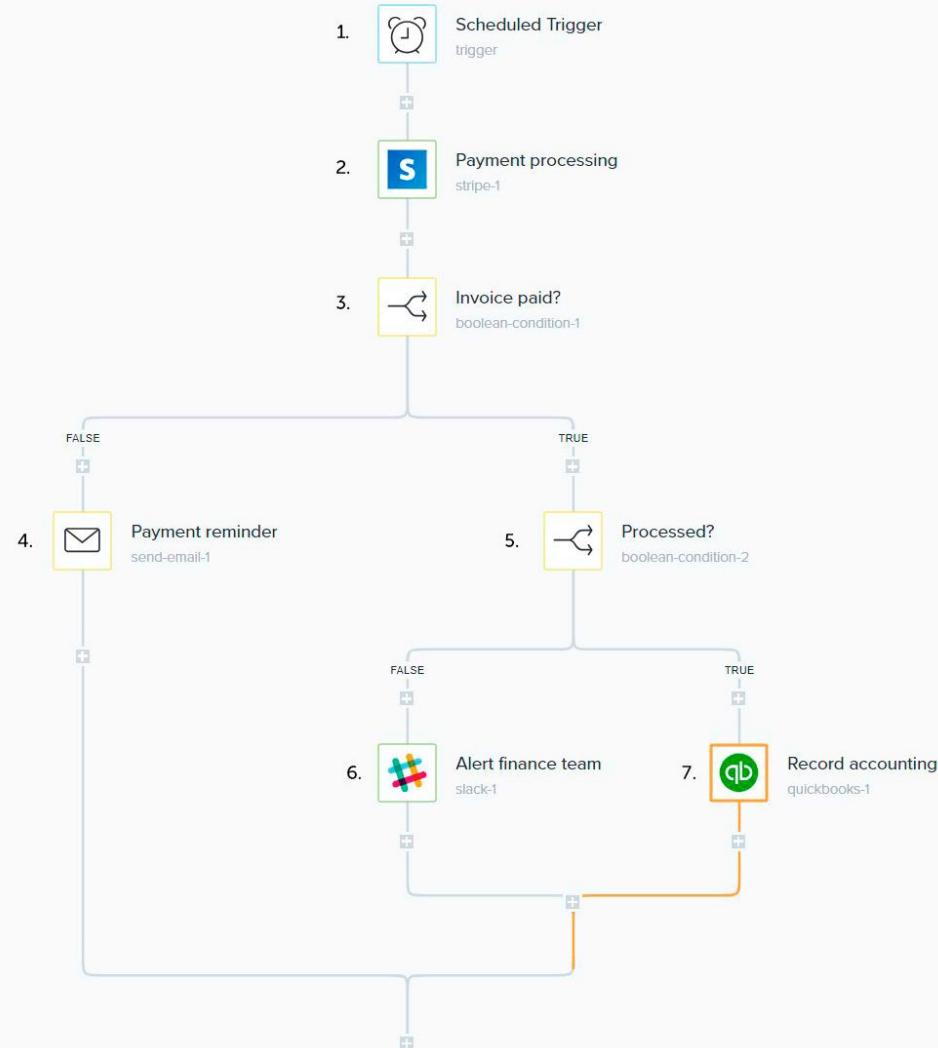
Scheduled triggers, as the name suggests, initiate specific workflows according to a pre-set schedule. At minimum, your scheduled triggers should let you initiate workflows at the following intervals:

Monthly   Weekly   Daily

(The best scheduled triggers are granular enough to run hourly or even minute-to-minute, and also allow for [cron scheduling](#) - a type of scripted job scheduler to set up custom schedules for any time-sensitive tasks you may have in the future).

### Some potential use cases:

- **Daily software diagnostics for SaaS:** Use a scheduled trigger to run diagnostics on your software product through a data warehouse (like Amazon Redshift) fed into an analytics platform (like Segment). This type of workflow can deliver daily reports to find potential technical issues and find upsell opportunities for customers with increasing usage.
- **Weekly/bi-weekly sales follow-up reminders:** Use a scheduled trigger kick off a weekly workflow that scans CRM records to find active opportunities with no sales activity for a week or more. This type of workflow could then send automated reminders to your account executive (AE) team to follow



Example: Schedule invoice tracking through payment processing & finance

up with worthwhile opportunities, and to kick less-worthy opportunities back to nurture status.

- **Monthly invoice tracking and processing**

Use a scheduled trigger to run an automated workflow each calendar month to track invoices and fire off appropriate follow-ups depending on invoice status.

#### **Breaking down the steps in this automated workflow, we have:**

**1. Scheduled trigger:** Every calendar month, this workflow kicks off automatically to begin the invoice tracking process.

**2. Payment processing:** Begin by calling up invoices from your payment processing solution (in this case Stripe).

**3. Boolean if/then condition - Has this invoice been paid?** Use a Boolean logic loop to determine whether an invoice has been paid. If the invoice has not been paid, the loop outcome is considered “false;” if it has been paid, the loop outcome is considered “true.”

**4. Invoice not paid - send customer courtesy email:** If the Boolean logic loop regarding invoice status was “false,” meaning the invoice was not paid, this workflow automatically deploys a courtesy email to that customer to gently remind them of their invoice status.

#### **5. Invoice paid - Boolean if/then condition**

**- Has this been processed?** Use a nested Boolean logic loop to then determine whether this “paid” invoice has been processed (“true”), or if it has not been processed (“false”).

#### **6. Paid invoice not yet processed - Alert finance team:**

If the processing status of the invoice was found to be “false” - that is, this paid invoice hasn’t been processed yet, this workflow sends an automated alert to the finance team through an internal communications tool (in this case, Slack).

#### **7. Paid invoice processed - Record in accounting software:**

If the processing status of the invoice was found to be “true” - that is, the paid invoice has been processed, this workflow triggers the invoice’s recording in the company’s accounting software (in this case, Quickbook).

**NOTE:** With a General Automation Platform, you can extend the functionality of this workflow greatly, such as by:

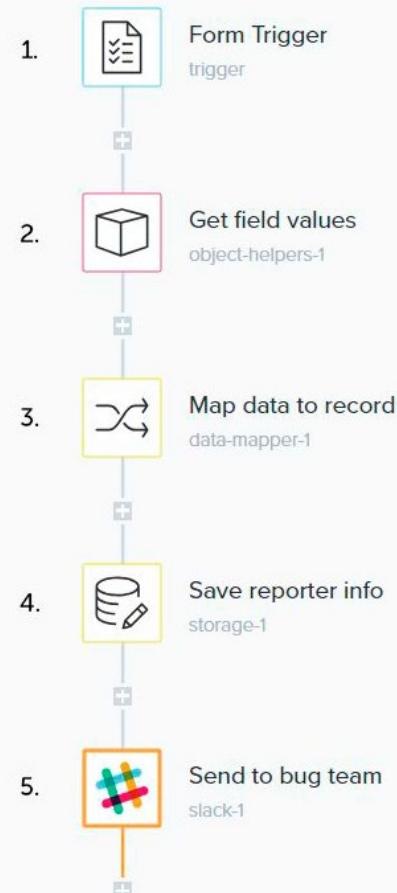
- Integrating your helpdesk solution to segment accounts for customer support, ensuring that the highest-value, fully-paid accounts receive support first
- Integrating your analytics program to track consistently delinquent customers and proactively address potential churn risk

## Form triggers

Form triggers kick off a workflow based on input from a custom web-based form. Since the best automated workflow solutions can seamlessly route all captured data to any other part of your stack, the form trigger unlocks a variety of potential automation use cases.

### Some potential use cases:

- **Marketing landing page:** Landing pages for a gated asset, like an eBook, typically have a form-fill that asks for your lead's name, and other info - which you can then route to the rest of your stack.
- **Pre-demo Q&A:** To prep for sales demos, you can use a form trigger to capture last-minute qualitative questions, then customize the presentation to those questions.
- **Ad hoc bug reports:** For cases in which you need responsive bug reporting - such as when launching a beta, use a form trigger to solicit feedback, and have a workflow instantly route those reports to the appropriate team.



*Example: Instantly send ad hoc bug reports to relevant teams*

This automated workflow proceeds as follows:

**1. Form trigger:** Set up a form trigger to capture bug report information with as much specificity as needed (specific feature with technical issues, specific URL where issue was sighted, etc.).

**2. Pull data from form fields:** Use an object helper to pull data objects (in this case, the submitted text) from each of the submitted fields.

**3. Map field data to the bug report:** Use an additional automation helper to map the pulled data into an internal repository for the submitted data.

**4. Save reported bug data and info on bug reporter:** Use an automation helper to store the recorded data and optionally, the identity of the bug reporter - a potentially nice-to-have option to help sort observed bugs vs. specific customers' use cases.

**5. Send bug report to relevant team via chat:** Send the stored and formatted data from each bug report field directly to the appropriate bug report team via a team chat application (in this case, Slack).

**NOTE:** With a General Automation Platform, you can extend the functionality of this workflow greatly, such as by:

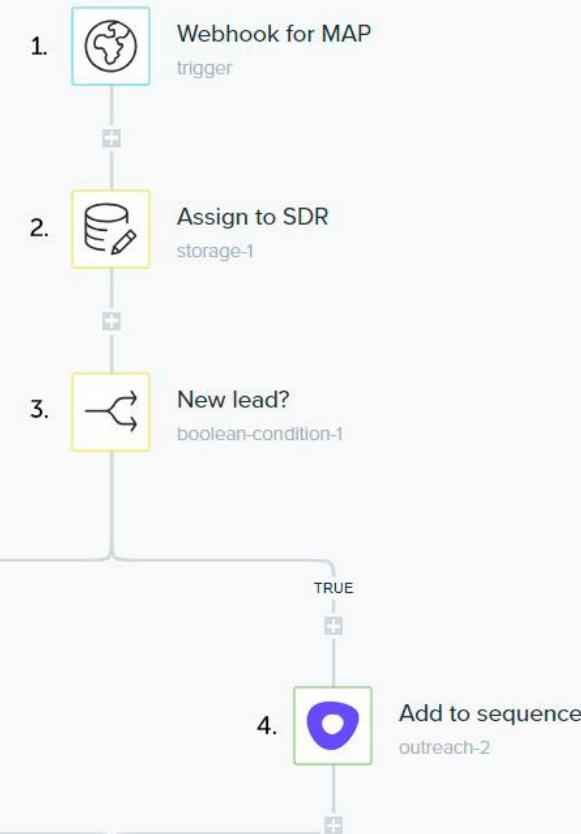
- Integrating your helpdesk solution to log ongoing issues and track progress over time
- Integrating your analytics platform to track and compare performance and outage frequency among different versions

## Webhook triggers

**Webhooks** are online notifications sent via web-based [HTTP](#) channels in response to an online data change - even something as simple as visiting a URL in your browser. In plain English, they are, most commonly, the online response to someone interacting with your website. Webhook triggers key off this type of notification to initiate an appropriate automated workflow in response to whatever action was taken at your website. (Learn more about webhooks in our guide, [What Are Webhooks?](#))

### Some potential use cases:

- **Enrich inbound marketing leads:** When leads visit your website, you'll want to identify them as new or existing ones, then update your records. You can use webhook triggers to kick off workflows that integrate your marketing platform and lead enrichment solutions to compare that lead's info to your existing database.
- **Rapid response to qualify 7x more new leads:** Research shows that following up with new leads within an hour of contact makes you [7x more likely to qualify them](#). You can use a webhook trigger with a workflow integrating your marketing platform and outbound email automation to rapidly send a follow-up email.



Example: Use a webhook trigger to add new visitors to outbound email sequence

Here's how this automated workflow works:

**1. Webhook for marketing automation platform:**

As soon as a new visitor arrives at your website, the webhook trigger keys off the website activity and confirms the lead status in your marketing platform.

**2. Assign to sales development representative (SDR):**

The next step in the workflow is to assign the lead to your SDR - in this case, by assigning and storing the data value of that particular SDR ("Lead owner value = 'Larry'") to this lead.

**3. Boolean if/then condition - Is this a brand-new lead?**

The next step in the sequence checks to see whether the lead that just visited your website is new - that is, whether this lead is already part of an existing email sequence in your outbound email automation platform.

**4. Add new lead to email sequence:**

If the Boolean condition is "true" - that is, if the lead was not already present in your outbound email automation list, the workflow then assigns this new lead to an email sequence, which then rapidly deploys a follow-up email to this new lead as part of a new automated email sequence.

**NOTE:** With a General Automation Platform, you can extend the functionality of this workflow greatly, such as by:

- Using Boolean if/then loops, plus random number generator helpers, to randomly assign inbound leads among multiple SDRs
- Integrating the UTM codes from the search terms these leads used to find your website, and make your outbound emails more powerful by surfacing those search terms directly in your outbound email subject lines

## Email triggers

As the name suggests, [email triggers](#) key off email inputs to kick off an automated workflow. Effectively, you can create a custom email address to accept an email input, then direct your prospects (or your customers or colleagues, as the case may be) to email that address to kick off the workflow. The best email triggers not only let you parse incoming email subjects and bodies for specific text strings, but they also let you email file attachments that you can then process using the components in your stack.

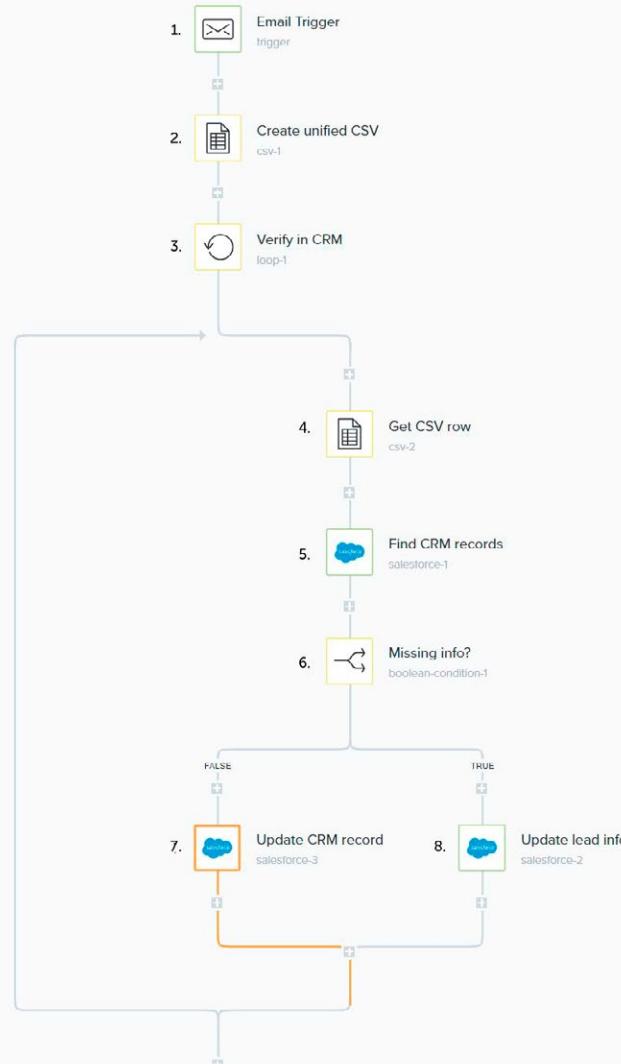
### Some potential use cases:

- **Migrating/managing customer order data:**

Direct-sales businesses may have extensive order histories for their customers on file in CSV format. They can use an automated workflow with an email trigger to email those CSV files as attachments to have their workflows concatenate and re-format as needed.

- **Compiling lead lists from multiple sources:**

You can use email triggers to start a workflow to unify multiple lead lists, such as from marketing programs or trade shows, by having teams send CSV lead lists to the email trigger. Each time it receives an email with a CSV attachment, that workflow combines those lead lists into a single, properly-formatted file.



Example: Collate lead list CSVs sent to an email trigger address

Here's how this automated workflow functions:

**1. Email trigger:** The workflow begins whenever someone emails the address created for the email trigger, then parses any CSV attachments.

**2. Create unified CSV:** The workflow then creates a new internal CSV to be populated with the info from the file received.

**3. Loop: Verify each lead vs CRM records:**

The workflow then begins a series of iterative, row-by-row loops to verify the info for each lead against the information contained in your CRM (in this case, Salesforce).

**4. Get CSV row:** The next step in the workflow is to begin checking the emailed CSV, row-by-row - each row of which contains a single lead.

**5. Find CRM records:** The workflow then pulls up the corresponding info for that lead in existing CRM records.

**6. Boolean if/then condition:** The workflow then starts a Boolean if/then loop to determine whether each lead is missing any important details (such as full name, email address, company name).

**7. Update CRM record:** If the Boolean condition is "false" - that is, if the lead info is complete and not missing anything, the workflow updates CRM data to record that the lead is already in the CSV.

**8. Update lead:** If the Boolean condition is "true" - that is, if the lead info is missing important details, the workflow calls those missing details from CRM data and updates them in the CSV file.

**NOTE:** With a General Automation Platform, you can extend the functionality of this workflow greatly, such as by:

- Exporting CSV lead lists to your marketing platform to prepare for upcoming marketing programs
- For certain leads designated for specific follow-up, such as sending out "swag" gifts, you can set these statuses to also automatically be updated in CRM

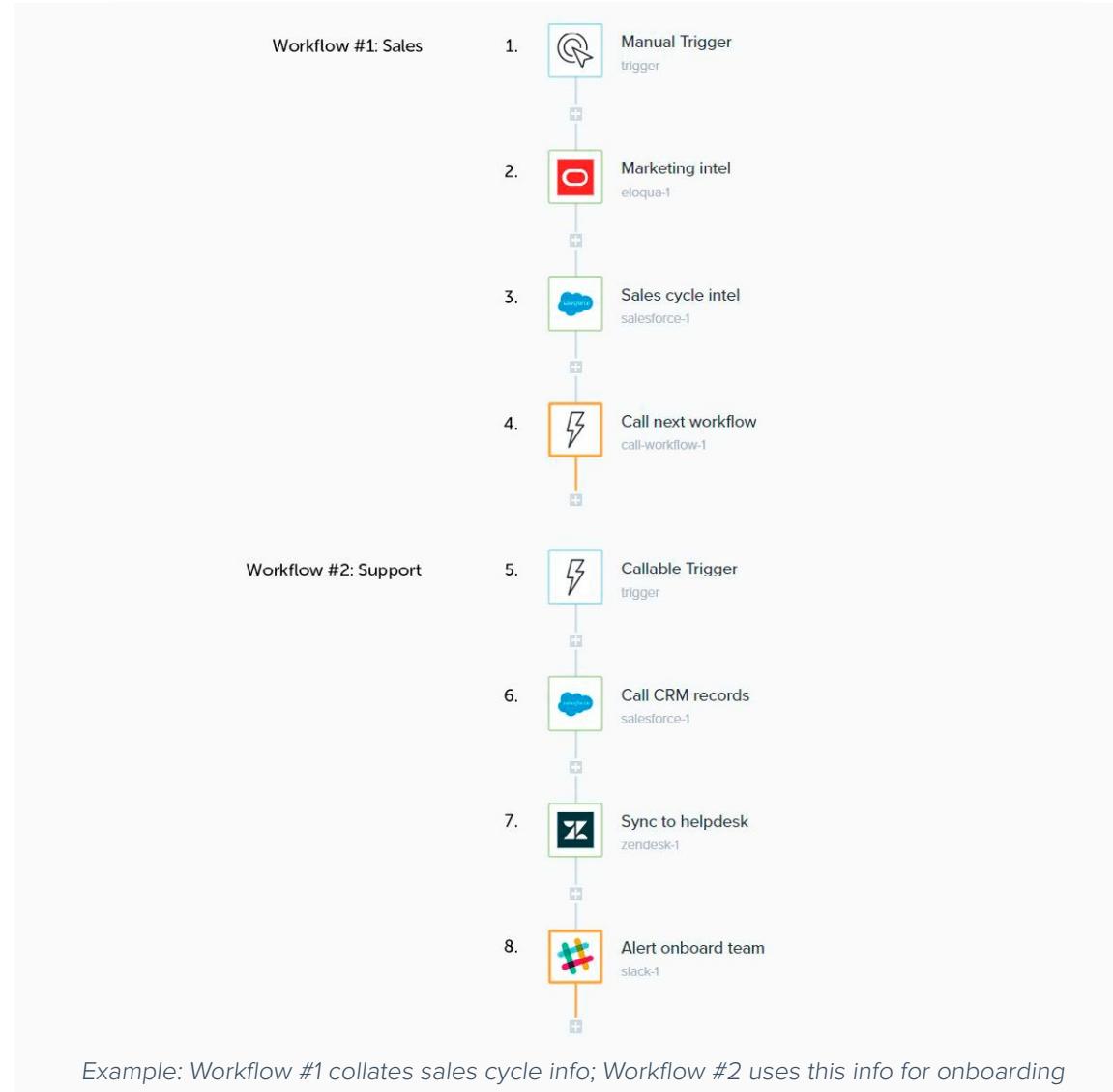
## Callable triggers

Callable triggers are powerful tools that trigger new automated workflows in response to an input from a previous workflow. It's quite possible - and with the right automation platform, much easier than you might think - to create complex arrays of workflows that cascade into each other to perform a nearly-unlimited number of data-related tasks for you.

**NOTE:** Though the best automation platforms can scale without limitation to accommodate all your needs, it may sense to break out task groups into different workflows. For example, flowing data across two teams - such as sharing a lead list from marketing to sales. Having smaller, modular workflows also makes them easier to maintain and re-use, making even very large and complex automations simpler to manage.

### Some potential use cases:

- **Handing off qualified leads to sales:** Once marketing has pushed leads through enough funnel stages to qualify them as sales-ready, it needs to flawlessly deliver every lead to sales as efficiently as possible. This may mean updating the sales team's CRM instances with account information, and/or sending alerts to account owners and specific team leads over internal chat.



However, these CRM instances or points of contact may change over time. Rather than get caught up in manually updating the marketing workflow's recipients, the marketing team may choose to end its workflow with a call to a separate alerting workflow owned by sales that starts with a callable trigger.

- **Handing off closed-won accounts to customer support:** By the same token, once deals have been closed-won, they need to be passed from sales to the customer team for onboarding. While it's ideal for organizations to integrate their CRM sales records with their customer support software to ensure the support team has all the intel from the sales cycle, again, the exact processes and owners may change over time.

Rather than manually update its workflows every time support updates its internal processes, the sales team may opt to end its workflow with a call to a different, support-owned workflow that initiates with a callable trigger.

#### Let's go through how this workflow functions:

**1. Workflow #1: Manual trigger for account handoff:** The trigger for workflow #1 is set to manual to kick off whenever a sales deal closes (sadly, this doesn't reliably occur on a regular calendar day).

**2. Marketing intel - transfer info from marketing touches to CRM:** The workflow then pulls any relevant intel from the newly-won account's previous marketing touches from its marketing platform (in this case, Eloqua) into CRM.

**3. Sales cycle intel - compile marketing & sales intel into CRM:** The workflow then compiles any notes from the sales cycle itself, recorded in CRM (in this case, Salesforce).

**4. Call next workflow - support team handoff:** With the revenue team's work at an end, this workflow draws to a close by calling up the next workflow, owned by the customer success team, to begin the onboarding process.

**5. Workflow #2: Callable trigger:** The second automated workflow, owned by the customer success team, triggers after receiving the call to do so from the previous workflow.

**6. Call CRM records - load sales & marketing intel:** The workflow then calls all the compiled intel on the account from the revenue teams to prepare it for transfer.

**7. Sync to helpdesk:** The workflow then transfers relevant intel into the support team's helpdesk solution (in this case, Zendesk).

**8. Alert onboard team:** Finally, the second workflow sends an automated alert to the support team to inform them that the new customer account has been migrated from CRM to helpdesk, and to begin the onboarding process.

**NOTE:** With a General Automation Platform, you can extend the functionality of this workflow greatly, such as by:

- Use calendar-based date & time helpers to set reminders for important onboarding milestones such as contract due dates and training session dates
- Using a CSV Processor helper to add any account data stored in CSVs (such as from previous lead lists) or to export account data to a CSV for later use

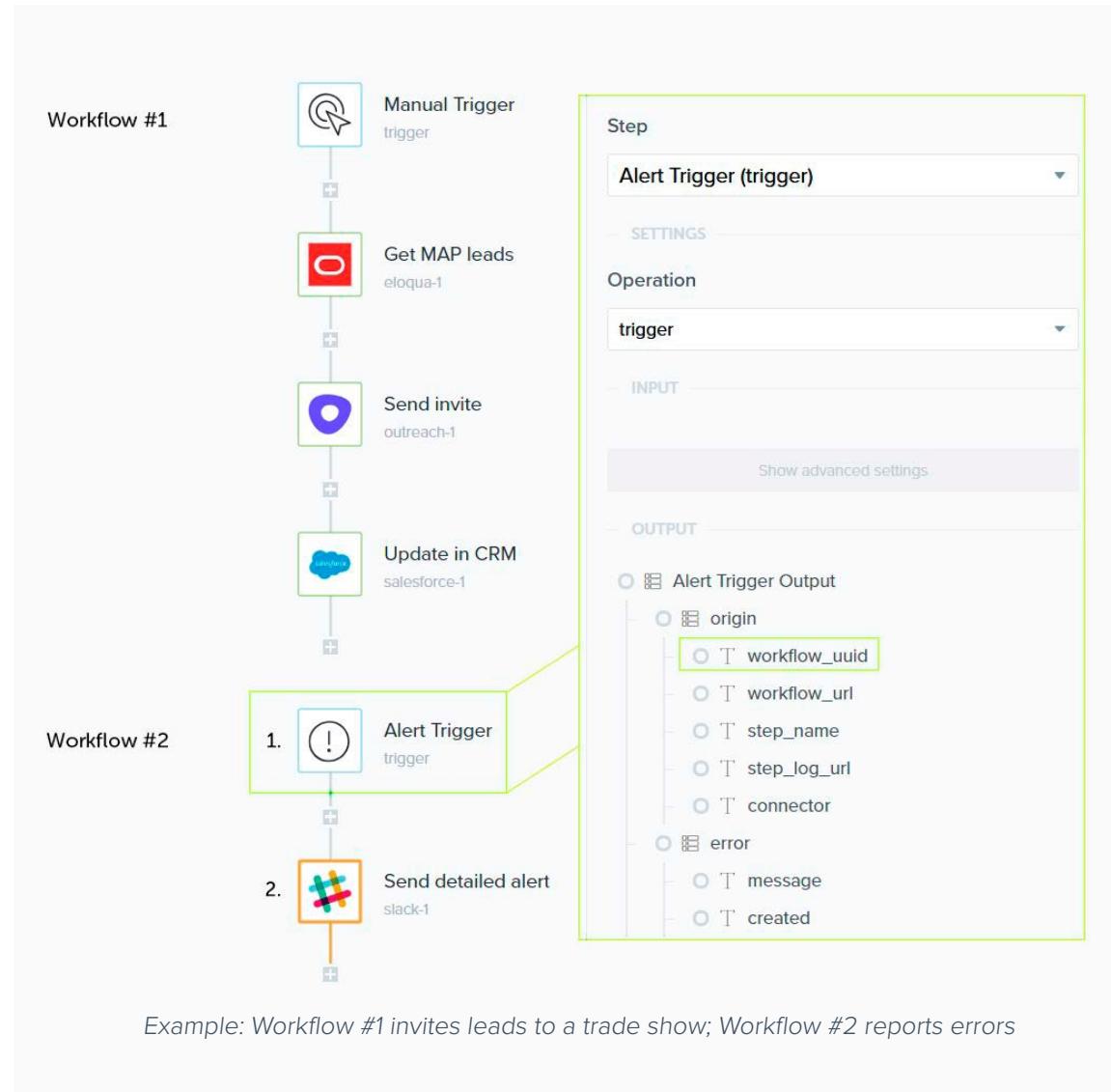
In this example, should either the sales team or customer success team make changes to their tech stack or processes, they need only update their own individual workflows so long as the CRM remains their shared source of truth.

## Alert triggers

*Alert triggers* are a subset of callable triggers that send detailed alerts whenever another automated workflow experiences any kind of error. The best automation platforms let you create alert trigger workflows separately from the workflow you'd like to monitor and simply use that workflow's ID to associate with the alerting workflow. They also provide full breakdowns of exactly which step in the workflow caused the error and link you directly to the relevant error logs.

### Some potential use cases:

- **In-depth error alerting from other automated workflows:** By setting inputs from previous workflows into an alert trigger, you can then create a separate alert workflow that, when initiated, messages you specific error information.



You'll note that we have our sample manual trigger workflow listed above, but now, it's associated with a second, alert trigger workflow:

**1. Alert trigger:** This alert trigger step in the second workflow is directly associated with the first workflow, in this case, via the first workflow's universally unique identifier (UUID) code. In this way, any errors that take place in Workflow #1 automatically get logged in Workflow #2.

**2. Send detailed alert:** In the second step, this workflow sends a detailed alert on the error, including:

- The workflow that experienced the error (linked by URL)
- The name of the workflow step that caused the error
- A link to the error logs for that step

In the case of this workflow, the alert is set to go out through internal chat (Slack).

**NOTE:** In addition to this type of automated functionality, an important companion to alerting is strong fail-safe functionality that will reliably run through your automated workflows even in the event of errors.

- In a best-case scenario, when any (hopefully) minor errors occur, your automated workflows will be supported with a fail-safe that still carries out their processes to conclusion, and deliver a full alert message with details on the exact issues.
- In a worst-case scenario, without alerting or fail-safes, in the event of any errors, your workflows will simply grind to a halt, leaving you empty-handed - without either the needed results of your workflow or any idea of exactly what went wrong!

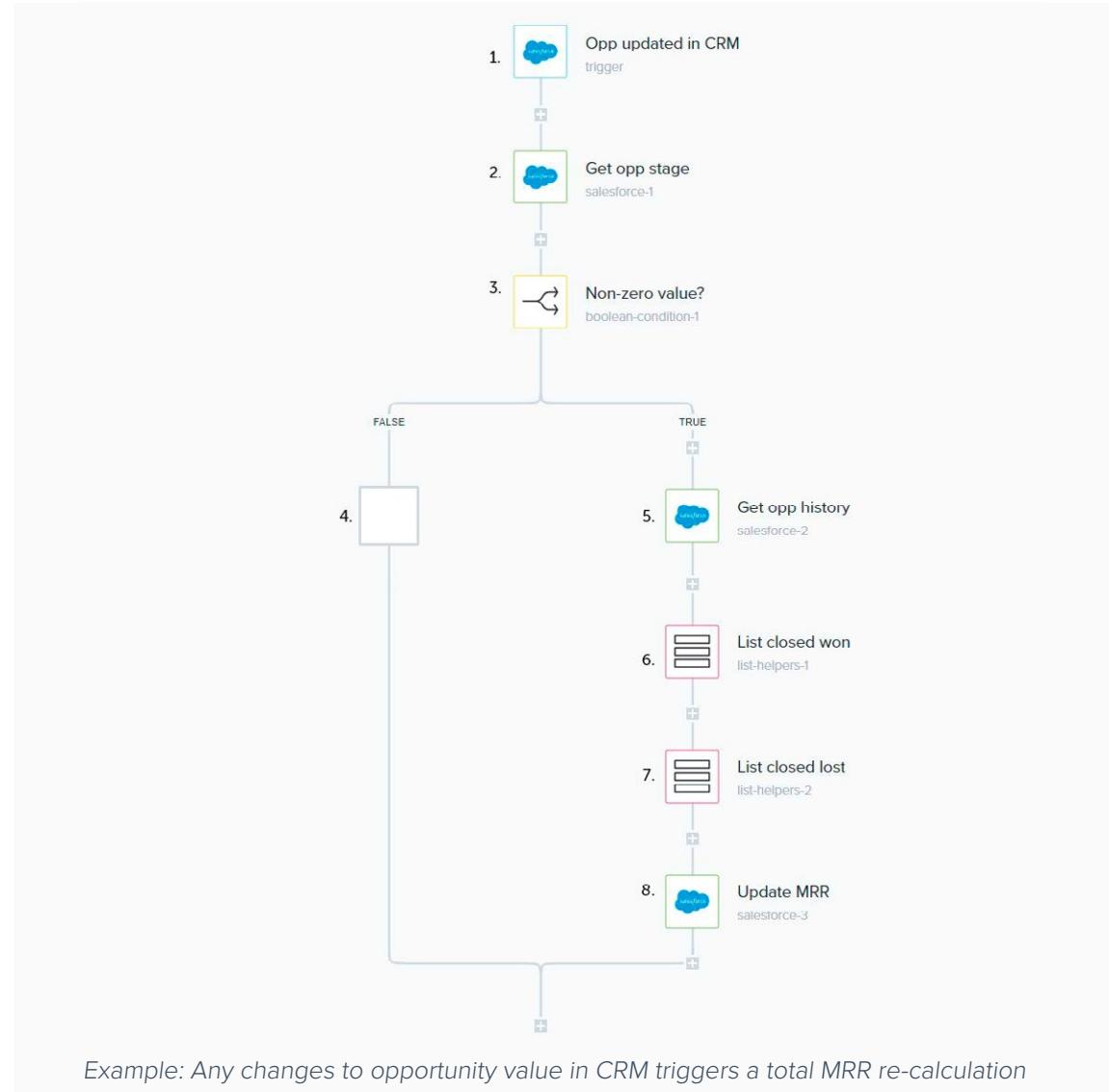
## Individual application triggers

Absolute best-in-class automation platforms also offer specialized automation triggers that are connected to important cloud stack applications you use every day. In the best cases, they offer triggers with varied use cases that key off different, important tasks performed by each of your software components, including:

- Marketing automation platform (MAP) - e.g. HubSpot
- CRM - e.g. Salesforce
- Cloud chat/collaboration - e.g. Slack, Trello
- Email automation/sales intelligence - e.g. Outreach
- Customer support/helpdesk - e.g. Zendesk, Intercom

### Advanced example:

Let's run through just one very simple example using a CRM trigger - in this case, using a CRM (Salesforce) trigger to automatically update the total value of a sales account when an opportunity associated with that account changes value.



Here's how this automated workflow functions:

**1. CRM trigger - Opportunity updated:** This automated workflow triggers from a CRM (Salesforce) event - in this case, an opportunity being updated. This opportunity could be the initial (and only) opportunity associated with this account, or one of many associated with a mature account to which your sales team has upsold other products or service packages.

**2. Get opp stage:** The workflow then confirms the stage of the opportunity, to ensure it's a valid opp that was entered correctly.

**3. Boolean if/then condition: Confirm opp value:** The workflow then confirms that the change in the opp value is not equal to zero (if an opportunity has been updated for reasons other than changing its value, then the total account value won't need to be adjusted.)

**4. If opp value change is zero, end the workflow:** If the Boolean condition is "false" - that is, the change made to the opportunity in CRM doesn't affect the value, the workflow ends (again, the point of this workflow is to automatically adjust the total value of the account based on any changes in value to individual opportunities).

**5. Get all previous opportunity history for the account:** If the Boolean condition is "true" - that is, the change made to the opportunity increased or decreased its value, this means the total monthly recurring revenue (MRR) for the account may be affected. This step retrieves previous opportunity history.

**6. List closed won opportunities:** To calculate the updated MRR for the account, the workflow uses a list helper to retrieve the values of all opportunities that were closed won.

**7. List closed lost opportunities:** To calculate the updated MRR for the account, the workflow uses another list helper to retrieve the values of all opportunities that were closed lost.

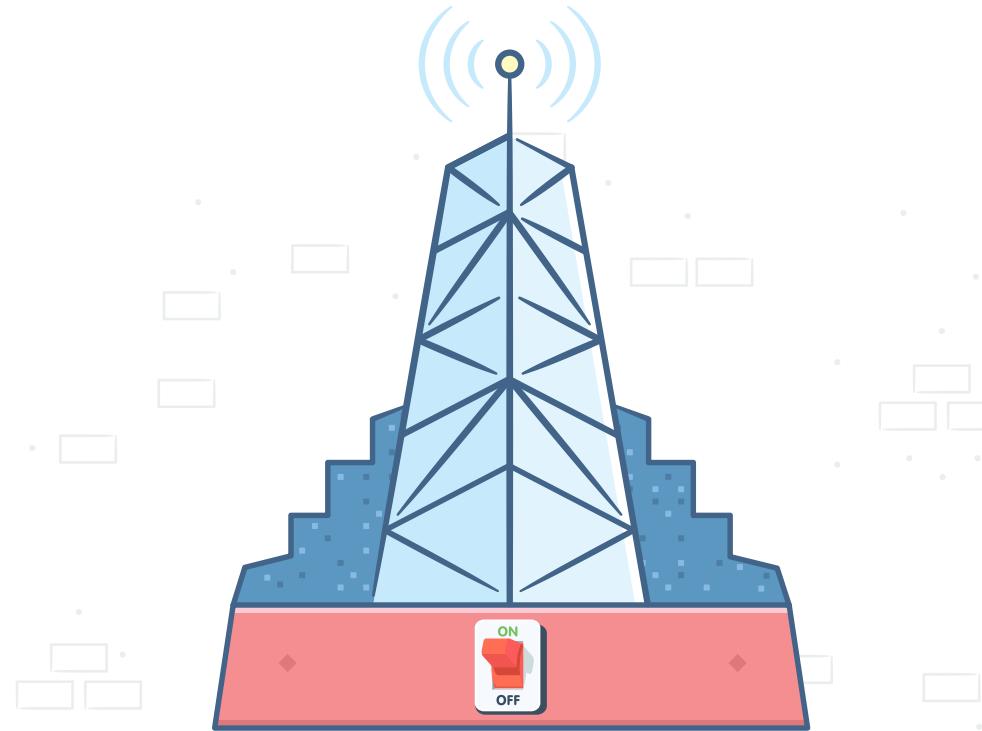
**8. Update total MRR:** The workflow then factors in the value change of the opportunity, plus the value of any previous closed won and closed opportunities, and calculates an updated MRR for the account.

**NOTE:** While this is somewhat of an internal bookkeeping workflow, having an array of triggers that can be keyed off the business software you use every day, your ability to set up powerful, time-saving automated workflows is practically limitless.

# Takeaways

You should now have a good idea of what automation triggers are, how they work, what varieties exist, and a few examples of the types of automated workflows you can build with them.

As you can see, automation triggers open up a whole new world of potential efficiency and time-saving gains, as well as new ways to intelligently take control of your most important data for marketing, sales, support, finance, operations, and many other functions.



# About this guide

Written By:



Andrew Park

Senior Content Marketing Manager  
Tray.io



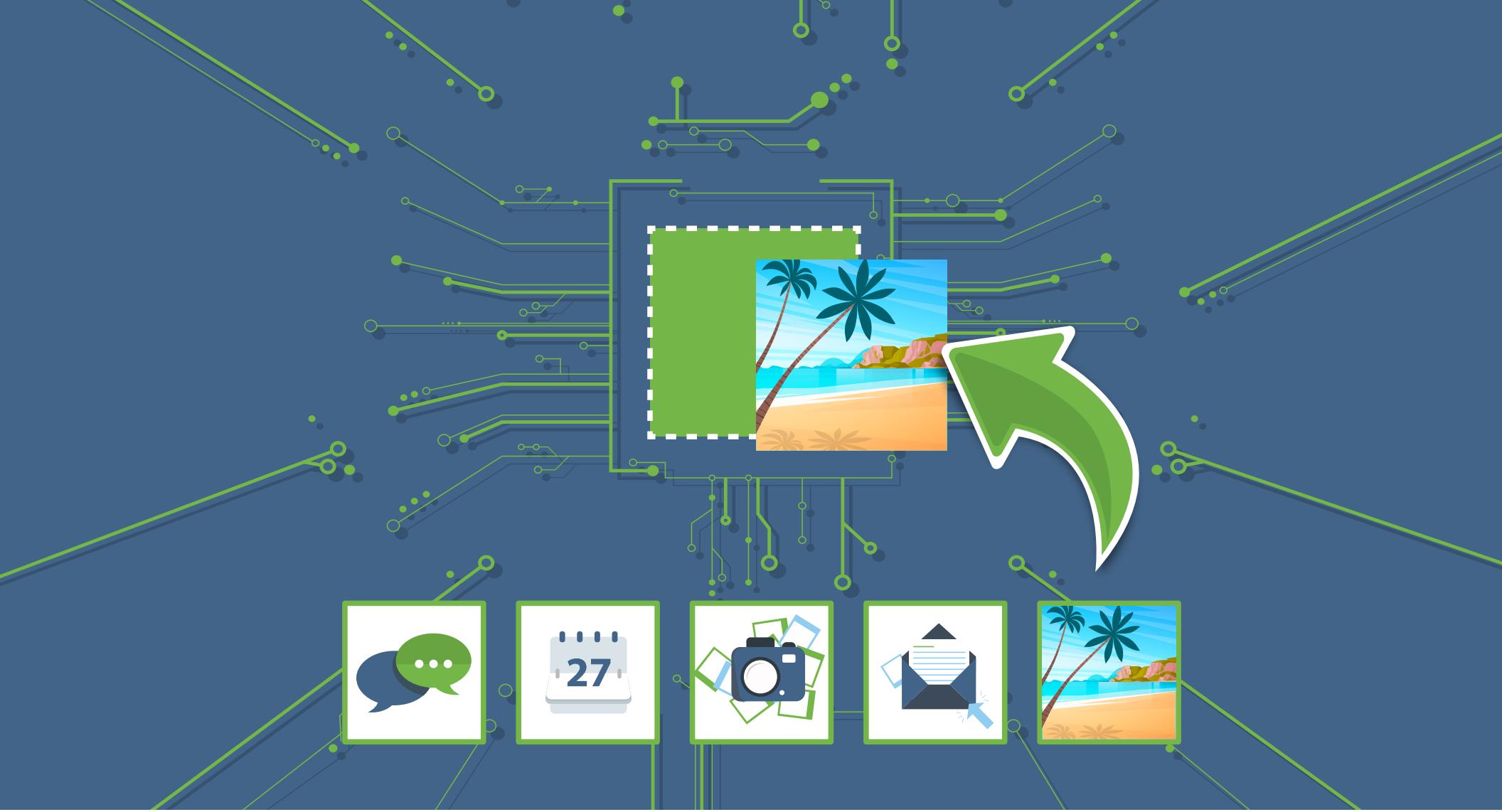
Additional  
Contributors:



Alex Ortiz  
VP Marketing  
Tray.io



Hank Taylor  
Director of Revenue Operations  
Tray.io



## Contact Tray.io

Telephone: +1-415-418-3570

Email: sales@tray.io

Website: tray.io

Blog: tray.io/blog