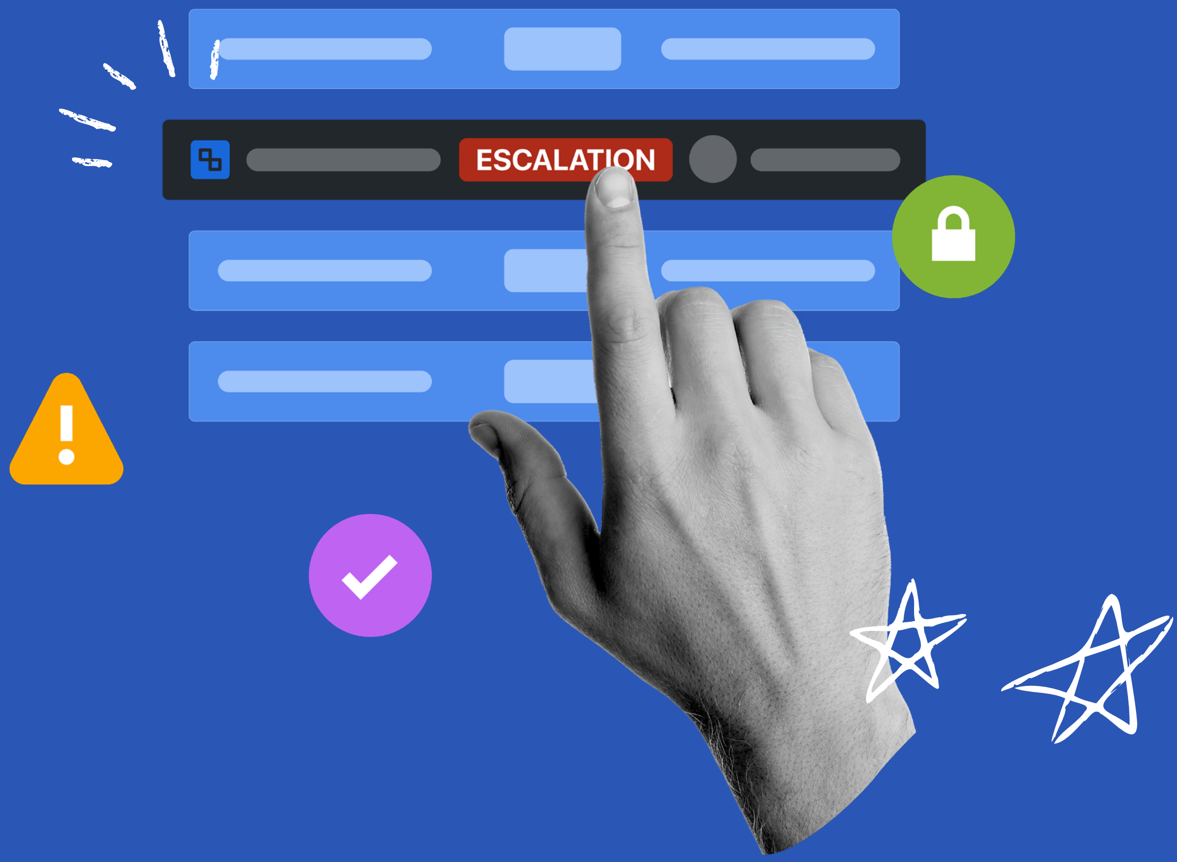
ATLASSIAN

The Official Atlassian Escalation Management Handbook

Resolve issues faster, satisfy more customers



When issues like bugs and urgent feature requests pop up, it's crucial for teams to align quickly on the next steps. This is the heart of escalation management, a complex challenge for businesses big and small. Effective escalation management uses clear steps for escalating issues to the right levels within your organization. This creates quick and efficient resolution, reducing the impact on customer service.

We're here to help teams around the globe improve their escalation management. Drawing on insights from industry leaders, we've captured Atlassian's approach to escalation management in one convenient guide. You'll learn all the lessons we've learned from handling escalations. While this guide is based on our unique experiences, we've designed it to be flexible enough to fit your team's needs. You'll also find sections suggesting popular alternatives for certain practices. Let's begin.

Table of contents

A quick note to readers

This guide is tailor-made for anyone involved in escalation management, or working closely with such teams. We'll zero in on the strategies these teams use to quickly solve major customer problems and amplify customer feedback throughout the company.

If you lead a company's Legal, Public Relations, or Human Resources team, you'll also find this guide useful, since escalation management relates to your company's big-picture strategy for handling critical issues. This requires a united effort from leadership, often bringing together folks from the executive board, IT, Legal, Communications, and HR.

Value of the Atlassian Cloud Platform

Escalations can be divided into two categories:

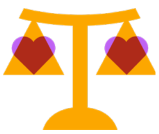
Hierarchical escalation
Hierarchical escalation is a structured process in which unresolved issues are passed up the chain of command within an organization. It starts when lower-level employees encounter issues they cannot resolve, and escalates through higher levels until resolution. This type of escalation focuses on internal hierarchy rather than involving a separate Escalation Management team.

Functional escalation
Functional escalation occurs when a support engineer can't resolve a customer's request using their expertise, which requires other individuals or teams with specific knowledge to address the issue. This type of escalation aims to apply the right resources and knowledge for efficient issue resolution, as opposed to hierarchical escalation that involves higher-ranking employees. This process is critical for effective escalation management, improved resolution times, and more customer satisfaction.

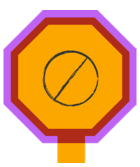
Atlassian's escalation values

Atlassian Cloud streamlines workflows and connects your teams on the common Atlassian platform The platform enables consistent experiences and information cross-flow, weaving smart collaboration experiences such as automation, analytics, and centralized administration into the fabric of all of our products.

Open company, no bullshit.
We believe in transparency and simplicity. Our escalation process is clearly defined, ensuring everyone knows their roles and responsibilities, and promoting trust and efficiency when we address challenges. We also believe that implementing an escalation tracking system develops transparency and accountability, allowing us to monitor progress and resolve all issues.

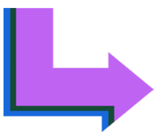


Build with heart and balance.
We provide easy internal access to contact information for escalation managers and teams. By responding with urgency and care, we demonstrate our committed support to both customers and colleagues.



Don't #@!% the customer.
By setting clear expectations for response times at different escalation levels, we respect our customers' time and trust in us, informing them when to expect solutions and updates.

Play, as a team.
We classify issues into different severity levels with clarity, helping our team prioritize effectively and mobilize quickly, as well as leveraging our collective strengths to address the most critical issues first.



Be the change you seek.
Escalation templates and guidelines empower everyone to communicate issues efficiently and effectively, streamlining the process and fostering a culture of proactive problem-solving.

Key strategies for effective, collaborative escalation management

- Foster autonomy and team empowerment in decision-making. During escalations and retrospectives, this promotes team independence, leverages expertise for solving issues, encourages proactive escalation management, and nurtures a culture of continuous improvement.
- Develop a consistent culture across teams for identifying, managing, and learning from escalations. A unified approach to escalations enhances efficiency, speeds up responses, and encourages ongoing improvement and teamwork.
- Ensure teams share a common approach to escalation handling, from identification to resolution and reflection. This fosters a collective attitude towards proactive problem-solving, collaboration, and learning, aiming to enhance future responses and cultivate a positive team environment.

Escalation stages and their related values

Stage	Escalation value	Related Atlassian value	Rationale
Discovery	Discover the source of the escalation.	Build with heart and balance.	Identify and understand the causes and impacts of an escalation.
Triage	Escalate, escalate, escalate.	Play, as a team.	Prioritize escalating issues to ensure timely resolution and appropriate resource allocation.
Action loop	Take action and fix the problem.	Don't !@#$ the customer.	Take necessary actions to resolve the escalation while maintaining a customer-centric approach.
De-escalation	Reduce the temperature.	Open company, no bullshit.	Communicate openly and transparently with customers and stakeholders to de-escalate the situation.
Retrospective	Find the root cause and always keep it blameless.	Be the change you seek.	Identify the root cause and the changes that will prevent similar escalations from occurring again. Commit to delivering specific changes by specific dates.

The importance of escalation management

At Atlassian, managing escalations is key to addressing customer issues and resolving them effectively. This involves clear communication, coordinated teamwork, structured approaches, and ongoing refinement that keeps our customers happy and maintains the quality of our service.

When facing an escalation, teams need a strategy that covers these:

- **Organization essentials.** Critical customer issues must be handled quickly and efficiently. Escalation management helps teams prioritize and tackle these issues promptly by using commonly shared thresholds, leading to quicker resolutions and improved service experience.
- **Streamlined escalation handling.** A well-defined escalation management strategy allows teams to respond swiftly, communicate transparently with stakeholders, and collaborate effectively to resolve issues faster. This approach is essential to learn from incidents and improve processes over time.
- **Defined roles and responsibilities.** The escalation manager (EM) plays a pivotal role at Atlassian, championing the customer's perspective. The EM leads the team in responding to feedback, managing expectations, and driving solutions.
- **Impact of a structured approach.** Atlassian has transformed the way we handle escalations. We've adopted this Escalation Management Playbook, which gives us a clear framework for managing these situations. This includes defining roles and responsibilities, building communication pathways, improving coordination, and strengthening the team's ability to manage escalations more efficiently. The Playbook aligns with our goal to alleviate customer concerns, rebuild trust, and minimize repeat issues.
- **Ongoing improvement and building trust.** Creating a specialized team for escalation management can address the significant impact of these events. This strategy focuses on refining our processes, ensuring consistent experiences, and leveraging insights to enhance our products, platforms, and services. Building and maintaining customer trust is our primary objective, which we achieve through strong leadership and dependable processes.

Recommended tools for escalation management

The escalation management process described in this Playbook uses several tools that are specific to Atlassian and can be substituted as needed.

- **Escalation tracking.** At Atlassian, every escalation is tracked as a Jira issue, with a follow-up issue created to track the completion of Retrospective Reviews. Atlassian uses a heavily customized version of Jira.
- **Chat room.** Maintain a real-time text communication channel to diagnose and resolve the escalation as a team. We use Slack.
- **Video chat.** For many escalations, team video chat like Zoom can help you discuss and agree upon approaches in real-time.
- **Documentation.** We use Confluence as our documentation tool for creating escalation-state documents and sharing Retrospective Reviews, and we also use Atlas to stay aligned with a wide range of stakeholders.

A note on escalation tracking

In Jira, we use a simple "New → Discovery → Triage → In Progress → Resolved → Closed" workflow to track escalations. Simplicity is good because we want to minimize complexity during escalations. Using a Jira ticket allows us to track escalations consistently, with a rich set of data that we can extend through automation, good reporting, and visualization features.

The escalation manager (EM)

The escalation manager (EM) leads each escalation, steering the wheel and wielding the authority to navigate towards resolution. The EM is identified as the escalation issue's assignee and has the power to pull in extra resources, delegate tasks, and maintain focus among all parties.

At Atlassian, the EM plays a pivotal role, acting as a champion for our customers. The EM collaborates closely with Support Leadership, Product Management, and Development teams to amplify the customer's voice and make sure it influences our actions.

Key functions of the escalation manager

- **Leading customer collaboration.** This involves spearheading customer communications, organizing and prioritizing Atlassian's response efforts, and keeping customers informed about progress and expected timelines.
- **Driving impact.** The EM plays a key role in educating the organization by sharing customer insights and trends. The EM also fosters teamwork across Support, Product, and Engineering to deliver tangible results. Additionally, the EM creates and oversees a list of priorities and an action plan to address and resolve concerns collaboratively.
- **Establishing effective communication.** The EM is in charge of ensuring clear communication, timely follow-ups, and meaningful outcomes with Atlassian's key stakeholders. The EM gauges the severity of situations, understands the business implications of critical issues, and keeps stakeholders updated on their status.

Beyond their core responsibilities, EMs collaborate with stakeholders and other departments to prioritize issues such as bugs. They also serve as a point of contact for team members facing obstacles, direct client interactions in partnership with support teams, and adhere to the established escalation process.

The EM is expected to understand complex situations quickly, identify a way forward, demonstrate empathy toward customers, possess strong communication skills, and have a keen interest and dedication to solving problems. EMs should also be adept at planning, coordinating, and managing various tasks and issues simultaneously, from start to finish.

Reporting an escalation

Escalations come from many sources, in many ways. They can be the result of an ongoing critical incident, created by support engineers responding to customer issues, or submitted by anyone who works with our customers. No matter how an escalation occurs, the first step is logging an ACE (Atlassian Customer Escalation) ticket – in our case, a Jira issue.

We use **go/raiseanace**, a short, easy-to-remember URL that redirects Atlassians to an internal Jira Service Desk portal. The Global Escalation Management (GEM) Team has set up dashboards to monitor escalations in progress.

The reporter may need the following information for an escalation:

- A brief summary of the issue
- Background of the issue and actions taken to lead to escalation
- The customer and business impact of the issue, including an assessment of customers' temperature and risk of churn
- A specific call to action for the escalation team
- A customer identifier (account ID)
- The affected product

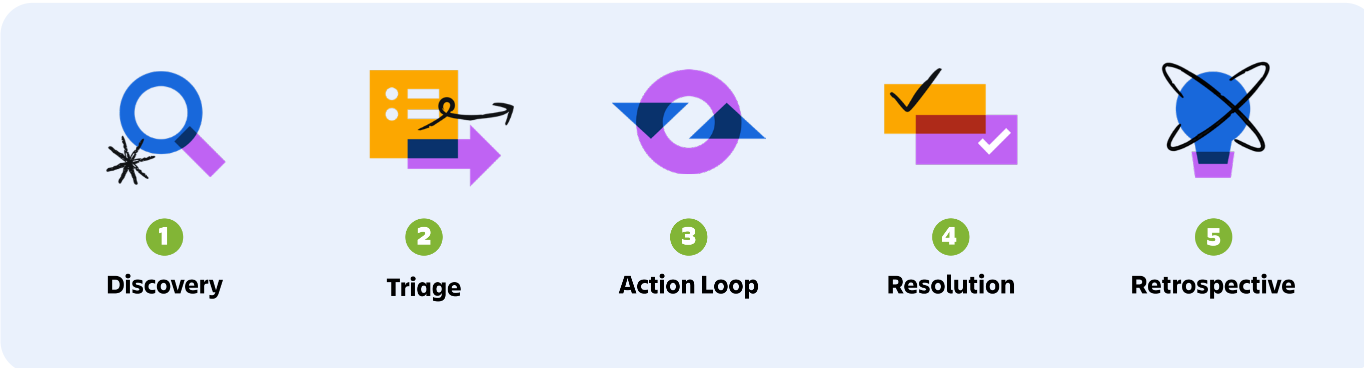
Jira fields to use when reporting escalations

Jira field	Type	Help text
Impacted customer	Text	Provide customer details and unique identifiers to help the Escalation Management team link the relevant customer to the ticket for reporting purposes.
What is the customer's primary working region?	Single-select	Identifies the customer's primary working region.
What is the impact to the customer and their business?	Text	How does this problem affect the customer? How is their way of working impacted?
Severity	Single-select	How detrimental is the impact to the customer's ability to operate?
What is the affected product?	Single-select	For which product is this escalation?
Are there related tickets associated with this escalation?		Choose whether there are any open internal or external customer tickets associated with this escalation.
Summary of events leading up to this escalation	Text	Describe the events leading up to this escalation. Please include the escalation trigger to help us to better understand the situation.
What are your expected outcomes from the escalation team?	Text	Please describe how an escalation manager can engage to help you.
Brief summary of this escalation	Text	Provide a summary of this escalation, keeping in mind your audience.

Once the escalation is created, its issue key (e.g. "ACE-1234") is used in all internal communications. We recommend having a reliable method to track your escalations, such as Jira or other project management tool.

Responding to an escalation

The following sections describe Atlassian's process for responding to escalations. The escalation manager (EM) takes the team through these steps to drive the escalation from detection to resolution.



Jira workflow statuses for managing escalations

The following sections describe Atlassian's process for responding to escalations. The escalation manager (EM) takes the team through these steps to drive the escalation from detection to resolution.

New
When the escalation issue has been created but hasn't yet been assigned to an escalation manager (EM), the escalation is unassigned and remains under the "New" status in the Jira workflow.

Jira offers several app integration options (e.g., Slack, Google Chat, Microsoft Teams) that trigger alerts when issues are created. This alerts the EM via an automated message to a designated chat channel.

The chat alert contains essential details such as the escalation's issue key, product, platform, region, situation summary, customer impact assessment, team's action request, and a descriptive pointer guiding the EM on how to initiate escalation management (with a link to the Jira issue).

Discovery
After receiving the alert, the EM immediately reviews and self-assigns the escalation issue to move it into the "Discovery" state, indicating that the escalation is being investigated. The Jira issue's assignee field always indicates the current EM. It's very important to clearly identify who's managing the escalation.

Next, the EM sets up the escalation team's communication channels. The goal at this point is to establish and focus all escalation team communications in well-known places. We normally use three team communication methods, each represented by a field on the Jira issue:

- **Chat rooms like Slack.** Messaging apps allow the escalation team to communicate, share and preserve timestamped observations, links, and screenshots. The naming convention used for the chat channel includes the escalation issue key and customer name (e.g. #ACE-1234-CustomerName), which makes it easier for stakeholders to find.
- **Video conferencing apps like Skype or Zoom.** Face-to-face synchronous communication helps teams build a shared understanding of the situation and make decisions faster. If you're all in the same physical location, gather the team in one room.
- **The escalation ticket itself.** All information concerning the escalation is captured here. We may also use a collaboration tool like Confluence to capture details of the "Discovery call."

The dedicated chat channel should be used to record important observations, changes, and decisions that occur in unrecorded conversations. The EM (or anyone on the escalation team) can simply note observations, changes, and decisions as they happen in real-time. Don't worry if it looks like people are talking to themselves! These notes are incredibly valuable during the Retrospective Review, when teams need to reconstruct the escalation timeline and figure out causes.

Most chat apps have a room topic feature. Jira automatically updates the room topic to include:

- The escalation summary and severity
- Who has which role, starting with the EM
- Links to the escalation issue, the video chat room, and the escalation-state document

Remember, we named the chat channel based on the escalation's issue key (e.g. ACE-1234-CustomerName). This consistency lets anyone easily locate the corresponding escalation chat channel and quickly get up to speed.

Triage

After the Escalation Management team has set up their communication channels, they can assess the escalation's severity and decide on an appropriate level of response. In order to do that, we consider these questions:

- What impact is the event having on the customer's business operations, financial health, reputation, etc.?
- When did the issue start?
- What events took place leading up to the escalation?
- What level of involvement or continued commitment do we need from the customer?
- Has a workaround or fix been identified?
- Are there other factors like public attention, security, or data loss?

Communications

After our team agrees on the escalation's severity, defining roles and responsibilities, and deciding on next steps with internal stakeholders, our next goal is to streamline communication, both inside and outside the company. This helps avoid duplicated efforts and reduce confusion. It's important to inform customers of when they can expect to receive the next communication, and where it will come from (e.g. via a ticket, email, or Statuspage). Prompt escalation updates build trust between both our teams and our customers.

How well are you communicating with customers during an escalation?
As a remote team, we shared the escalation timeline and collaborated directly on it using our video conference software. A designated team member acted as the scribe, capturing key points and adding them to the timeline.

We examined the escalation from beginning to end, starting with the initial signal of an issue to the final update provided to the customer.

After evaluating the escalation, we developed an action plan that focuses on people, processes, and technology. Each action was assigned an "owner," along with recommendations for next steps.

Our Global Escalation Management Standard Operating Procedure (SOP) provides a thorough approach for managing Atlassian Customer Escalations (ACEs) from their initiation to resolution. Here's an overview of the essential steps involved:

Acknowledgment
In this step, we recognize the escalation issue ticket in our Slack channel, then guide viewers to the primary escalation ticket in Jira for enhanced visibility. This serves as an initial measure to confirm the escalation is acknowledged and accurately recorded.

Investigation
The investigation phase requires a thorough examination of the escalation and related issues, to understand the problem and its impact on both the customer and Atlassian. This step also confirms whether the issue is within the escalation manager's scope. This includes checking related tickets, understanding the customer's perspective, and evaluating if the escalation summary and documentation are comprehensive.

Discovery
In the discovery phase, the escalation manager employs the Escalation Manager Scorecard Calculator, a tool designed to determine the severity level. The escalation manager then updates the escalation ticket with pertinent details like case type, severity, and trend. An informal collaborative space might be set up for casual communications, and a discovery call could be arranged to further explore the issue and define exit criteria.

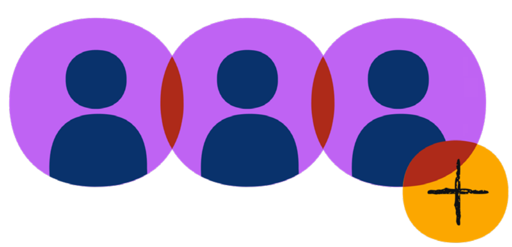
Triage

In this step, we move the customer ticket to the "Triage" status, identify stakeholders, and schedule a triage call within 48 hours of discovery. The purpose of the call is to complete the Triage Call Process, agree on a team matrix, and document a communication plan.

External team support

In order to route the escalation to the correct team, we must identify the type of support required. Here are some examples:

- **Non-engineering support.** If you require non-engineering assistance from Legal, Public Relations, Information Security, or other teams, the escalation manager promptly informs the relevant non-engineering team, creates action tickets, and oversees the resolution of the issue before closing the action ticket.
- **Developer on Support (DoS).** Here, we verify that the original support ticket is in a "DoS" status, which flags this ticket to the development team as an issue requiring their attention. Next, we create a sub-task action ticket so Engineering or the DoS can update the customer ticket status based on workflow details. This ensures timely notification and involvement of the engineering team in resolving the issue.
- **Bug support.** This step is designed to notify the engineering team about a customer issue that needs bug resolution. Here, we create a sub-task action ticket, link it to the Jira customer ticket, and confirm that the problem is resolved or earmarked for long-term action.
- **Suggestion and feature request support.** The escalation manager creates a sub-task action ticket that identifies the responsible product team member and confirms whether the feature request is promptly addressed or earmarked for long-term action.



Resolution and retrospective

In this final phase, we confirm that exit criteria have been met, carry out a retrospective review akin to a post-incident review (when needed), and change the customer ticket status to "Resolved." This stage allows us to learn from the escalation process and complete all necessary actions before closing the ticket.

Writing meaningful escalation reports

The complexities of escalation analysis present a significant challenge: the boundless volume of data and context that can be derived from a carefully crafted escalation report. Deciding what to capture, emphasize, or overlook depends on the information necessary to construct a narrative that catalyzes positive change. So, how do you determine what makes the cut?

1. Start by considering the audience driving the change. Are they developers who are deeply involved in the work? Product teams shaping the future? Department managers, customers, or perhaps executive leaders and investors? The audience significantly influences every aspect of an escalation report's design.
2. Recognize that there's no universal template for an effective escalation report. Each situation yields different insights. The authors of the escalation report must identify and focus on what this case reveals about teams, context, systems, etc. Organizations should also grasp proper categorization, trends, and other pertinent data points to refine their reports over time.
3. Cultivate time and dedication so you can master this skill set beyond drafting post-escalation documents. These reports can help you gather data, conduct group interviews, and more.

Often, these reports are mistakenly viewed as an excuse to list "remediation" or follow-up actions, which ignores a valuable opportunity for deeper learning. Remember, these documents should be crafted with the reader in mind.

Consider what the reader needs to understand the context of the escalation. For instance, the timing of an issue (coinciding with a new release, a surge of new users, or a high-profile event) may be crucial in some scenarios but irrelevant in others.

The approach to managing the escalation also warrants attention. Resolving certain issues might depend on the specialized knowledge of experienced engineers who are familiar with legacy systems. Highlighting these engineers and their insights could be vital. Or, if all the necessary engineers were attending a conference during the issue, this valuable context can also be added to your escalation report.

Effective escalation analysis demands the skills needed not only for writing these narratives, but also for knowing which data to collect, whom to interview, what diagrams to include, and more. Organizations that recognize and invest in this expertise will discover their competitive edge.

Escalation Retrospective Reviews

A blameless **Retrospective Review** transforms the challenges of escalations into opportunities for building resilience. It helps us capture insights and determine mitigation strategies for escalations. Here is a brief overview of how we conduct Retrospective Reviews at Atlassian.

What is a Retrospective Review?

A Retrospective Review is a comprehensive account of an escalation that includes:

- The impact of the escalation
- Measures taken to mitigate or resolve the issue
- Root causes of the escalation
- Steps to prevent future occurrences

At Atlassian, we organize all Retrospective Reviews using Jira issues to manage them effectively. We perform around 800 Retrospective Reviews annually. Depending on your specific requirements, you might opt for an alternative system, such as a Trello board or a Confluence page.

Why do we perform Retrospective Reviews?

A Retrospective Review extracts maximum benefit from an escalation by identifying all contributing factors, recording the details for future reference and pattern identification, and implementing robust preventative measures to minimize future occurrences.

Consider an escalation as an unplanned investment in your system's reliability. The Retrospective Review, therefore, is your strategy for optimizing the returns on this investment.

When is a Retrospective Review needed?

We always perform Retrospective Reviews for severity 1 and 2 ("major") escalations. For minor escalations, these reviews are optional. We encourage people to use the Retrospective Review process for any situation where it could prove useful.

Who completes the Retrospective Review?

The escalation manager is typically responsible for overseeing the entire Retrospective Review process linked to an escalation, from its initial drafting to its final approval and publication. The escalation manager appoints "retrospective participants" who make sure the review is conducted and documented thoroughly.

Because Retrospective Reviews are so complex and require significant effort – especially those reviews that concern infrastructure and platform-level escalations that impact multiple areas of the organization –the escalation manager might often involve a dedicated program manager or product lead. This strategy recruits teammates who have essential expertise for effective cross-functional collaboration, and who can commit to overseeing complex, cross-team initiatives identified during the review.

Why should retrospective reviews be blameless?

When things don't go as planned, we instinctually wonder who's at fault and try to shield ourselves from blame. Attributing blame can undermine the effectiveness of the Retrospective Review for several reasons:

- The fear of jeopardizing one's reputation or career prospects often takes precedence over the organization's best interests, leading individuals to conceal or distort the truth to safeguard their own interests.
- Assigning blame is unkind and can foster a culture of fear and mistrust if it becomes a pattern.
- Rather than focusing on why an individual made a certain decision that led to an escalation, it's more constructive to explore why the system allowed or encouraged that decision.

The facilitator of the Retrospective Review must counteract the instinct to assign blame. The review should aim to impartially and accurately assess the events leading to the issue, in order to identify root causes and implement effective mitigations. We start with the premise that our team members have good intentions, and we avoid attributing faults to individuals.

To ensure a safe environment for all participants in our Retrospective Reviews, we employ the following practices:

- Begin the meeting by emphasizing that the review is blameless and explaining why it's being performed.
- Refer to individuals by their roles (e.g. "the on-call Widgets engineer") rather than by name, while still clearly presenting the facts.
- Frame the review's timeline, causal analysis, and mitigations within the context of systems, processes, and roles, rather than focusing on individuals.

Our approach to blameless Retrospective Reviews, and the insightful concept of "second stories," is inspired by **John Allspaw's foundational article**.

An overview of the Retrospective Review process

In order for Retrospective Reviews to be effective, the process has to help teams to easily identify and fix root causes. The exact methods you use depend on your team culture. At Atlassian, we've discovered a winning combination of several methods:

- **Single-point accountability** for Retrospective Review results can clarify responsibilities. The assignee of the Retrospective Review ticket is always the person accountable.
- Use **face-to-face meetings** to speed up analysis, quickly create a shared understanding, and align on what needs fixing.
- **Review and approval** by our engineering leadership team can encourage the right level of rigor and priority.
- Significant mitigations have an agreed **Service Level Objective (SLO)** for completion (eight weeks in most cases), with reminders and reports to ensure completion.

This Retrospective Review process isn't complete without setting up a Retrospective Review meeting, capturing actions, getting approval, and communicating the outcome.

Steps for the Retrospective Review owner to take

1. Schedule the Retrospective Review meeting. Invite the delivery team, impacted teams, and other stakeholders using the meeting invitation template.
2. Use the "Five Whys" technique to traverse the causal chain and discover the underlying causes of the escalation. Prepare a theory of what actually happened compared to the ideal sequence of events. List some proposed mitigations for each cause you identify.
3. Meet with the team and run through the agenda. Check out "Retrospective Review meetings" below.
4. Follow up with the responsible engineering managers to get a time-bound commitment to actions that were agreed upon in the meeting.
5. If the participating teams haven't done so already, raise a Jira issue for each action in the backlogs of the team(s) that own them. Link from the Retrospective Review issue.
6. Add the engineering manager(s) from the appropriate group(s) to the "Approvers" field on the Retrospective Review.
7. When you feel the Retrospective Review ticket is ready for approval and publication, select the "Request Approval" transition to alert the nominated approvers. An automated comment in the issue provides instructions for approvers.
8. Follow up with approvers and other stakeholders as needed. This often involves answering questions and driving decisions.
9. Once the Retrospective Review is approved, we want to multiply its value by sharing what we learned with the whole company. To this end, we have automation that drafts a blog post in Confluence when the Retrospective Review ticket is approved. Follow the "Draft Blog Post" link on the Retrospective Review ticket to edit and publish the blog post.

Retrospective Review meetings

The Retrospective Review lead, usually the escalation manager, organizes a Retrospective Review meeting after filling out the issue details and hypothesizing about the causes of the escalation and its potential mitigations.

At Atlassian, we've observed that in-person or face-to-face meetings foster quicker and more effective communication, deeper analysis, and more meaningful learning. We encourage teams to meet in person whenever feasible. However, due to geographical or scheduling constraints, it might not always be possible to gather everyone simultaneously. In such cases, consider holding multiple sessions or gathering input asynchronously.

While we use the term "face-to-face," these meetings often occur via video conference to accommodate our distributed teams, and we may include a broad audience of stakeholders.

Standard agenda for a Retrospective Review meeting

1. Remind everyone that Retrospective Reviews are blameless, and explain the purpose of the review. Refer to the section "Why should retrospective reviews be blameless?"
2. Review the timeline of events, making adjustments as necessary.
3. Share your theory on the causes and potential mitigations to prevent future escalations. Engage with attendees to develop a collective understanding of what occurred, why, and how to take action.
4. Encourage "open thinking" (e.g. "What could we do to prevent this type of escalation?") over "closed thinking" (e.g. "Why weren't we monitoring system X?"). Open thinking promotes innovative ideas by not limiting discussions to the current state.
5. Pose questions like "What went well?" or "What could have gone better?" or "Where did we get lucky?" to identify near-misses, learnings, mitigations, and actions beyond the immediate scope of the escalation. Our aim is to derive as much value as possible from the escalation experience.

Conclude by expressing gratitude for everyone's participation and insights.

Template for inviting participants to the meeting
Please join me for a blameless Retrospective Review of <link to escalation>, where we <summary of escalation>.

The goal of this Retrospective Review is to maximize the value of an escalation by understanding all contributing causes, documenting the escalation for future reference and pattern discovery, and implementing effective preventative actions to reduce the likelihood or impact of recurrence.

In this meeting, we'll analyze the escalation, identify its key causes, and decide on actions to mitigate them.

A productive Retrospective Review meeting typically generates numerous mitigations and actions. While this is beneficial, Retrospective Review meetings may not be ideal for significant prioritization decisions due to a lack of context (e.g., other ongoing work competing for resources).

Therefore, it's often more practical to discuss the identified actions with responsible managers after the meeting, in order to secure firm, time-bound commitments. These commitments should then be documented in the Retrospective Review issue.

How are Retrospective Review actions tracked?
For every action that comes out of a Retrospective Review, we:

- Raise a Jira issue in the backlog of the appropriate team. We track all of our Retrospective Review actions in Jira.
- Link the actions from the Retrospective Review issue as long-term or short-term actions to the customer escalation ticket.
- Track long-term actions in our Atlas tool for visibility.

We use Jira issue links to track actions because our teams use many different instances of Jira Cloud and Server. We track and report on this data using custom tooling that spiders remote issue links and populates a database with the results. This allows us to track the state over time and create detailed reports.

Adding your insights and feedback

We strongly encourage everyone to contribute their experiences and learnings so we can evolve this Handbook. By sharing your knowledge, you play a vital role in keeping this Handbook relevant and effective, fostering a culture of continuous improvement and collaboration. This approach aligns with our core value of "Open company, no bullshit," which promotes transparency and wisdom. Help us transform this guide from a static document to a living reflection of our collective expertise and growth.

Thanks for reading

You've just learned Atlassian's secrets to effective escalation management. We hope you found this Handbook useful. Keep calm and carry on!

If you have any feedback or suggestions, please send us an email at escalation-handbook@atlassian.com.

Discover the best ways to resolve your toughest issues. Learn how Atlassian manages escalations and get insights on creating a process that works at your organization.

Learn more at
Atlassian.com

ATLASSIAN