



Making the case for synchronization of issue trackers



Table of contents

1. Working with external teams & subcontractors

- The challenge;
- The solution;

2. Local Collaboration in the same issue tracker

- The challenge;
- The solution;

3. IT Outsourcing

- The challenge;
- The solution;

4. Migrating to another issue tracker

- The challenge;
- The solution;

5. The Centralized Master Instance

- The challenge;
- The solution;

Synchronization - the answer to a multitude of problems



Summary



In this article we will elaborate on common use cases that require synchronization:

1. [Collaborating with external teams or subcontractors](#)
2. [Local collaboration across different projects](#)
3. [Working in remote instances from within your own issue tracker](#)
4. [Live migration of data to a remote issue tracker](#)
5. [The centralized Master instance](#)

And more. (you can find additional use cases on our documentation page [here](#)) Whether you're a developer, administrator, project manager or customer-facing problem solver, you're familiar with the need for issue trackers like Jira, ServiceNow or [HP QC/ALM](#). And the benefits they bring to your role;

- Improved visibility and prioritization
- Increased speed of issue resolution
- Happier customers

But increasingly, issue trackers are becoming a victim of their own success and your teams are being faced with a new set of challenges. You will often work on an assortment of projects, across a number of teams, who are likely to be working with their own issue tracking instances or across different projects inside one issue tracker. The result? You are faced with disjointed issues and inconsistent workflows. This is where we see issue tracker synchronization adding a new dynamic! Described in simple terms... ***Issue tracker synchronization tools ensure that relevant issue information is available across multiple systems - in real time. Allowing teams to collaborate consistently on issues within their own environment.*** In turn, this:

- Reduces the chance for human or system error
- Allows you to safely migrate your data to another system
- Grants you full control and autonomy over your own system

Issue tracker synchronization is valuable because it creates a better way of working together across a huge variety of scenarios. **Let us explain...**

1. Working with external teams & subcontractors

Your **customer-facing team** logs issues in one system and external subcontractors do the same in another system. These subcontractors play an important role in resolving these issues and you need to collaborate regularly.

working with teams with disconnected systems

The challenge;

How can your team deal with multiple issues efficiently when some of the issue information is maintained into 3rd party systems? There is the risk that you miss details, get confused by different reference numbers, and need to manually interact to resolve every issue. Opening up your own issue tracker to collaborate creates additional problems, like having to configure privacy configuration and workflows.

The solution;

jira to jira integration
The above is an example of a Jira to Jira integration in action. The solution is to synchronize all relevant issue data from your subcontractors into your own environment, giving you a complete view of progress status without affecting your own workflow and environment. And keeping your confidential data secure.

[TRY OUT OUR SYNCHRONIZATION TOOL EXALATE](#)

2. Local Collaboration in the same issue tracker

You have multiple teams working in one single issue tracker across multiple projects. For example, a customer facing team making tickets and an issue-resolution / development team resolving issues.

The challenge;

When working with different teams in the same issue tracker, each team will most probably have different workflows and responsibilities. Communicating on progress status takes time and leaves room for human error. Your teams are currently collaborating on issues in one of two ways; ~~different workflows across issue trackers~~

- *tracking each issue with an all-encompassing workflow to capture the tasks every team member has to undertake?*
- Have different workflows that track the work for each member?

In the first scenario it is difficult to get an accurate view of what is happening, and in the second scenario, there is the need for manual communication when an issue is resolved.

[TRY OUT OUR SYNCHRONIZATION TOOL EXALATE](#)

The solution;

~~local synchronization~~ Instead of communicating through email and/or manually sharing data, synchronize all of your development issues and support tickets into one view. Use customization to keep your own workflow and choose what fields need to be shared. Gain a full overview of progress status and new information.

3. IT Outsourcing

You're an organization which offers **outsourced IT consultancy** to a number of clients. All of these clients are using their own issue tracking technology. ~~separated issue trackers~~

The challenge;

As your client base grows, so does the number of issue trackers you are expected to work with. This requires huge investments in time and training to adopt clients' issue-trackers, processes and workflows. Time which could be used on resolving the issues and customer queries themselves. Prioritization is almost impossible as there is no single view of information and task deadlines.

[TRY OUT OUR SYNCHRONIZATION TOOL EXALATE](#)

The solution;

Synchronize all relevant issue data from your clients into your own environment. This allows your IT consultancy teams to organize client cases from their own trusted environments, and prioritize accordingly. One effective environment which serves both the IT consultancy and their clients. ~~Synchronization tool~~

4. Migrating to another issue tracker

You're switching issue trackers, and you need to migrate all of your data over to a new system.

The challenge;

Implementing a traditional migration and duplicating data from one issue-tracker to the other inevitably requires downtime. This downtime is something your team cannot afford, along with the risk of losing key data.

The solution;

Live migration. Use real-time synchronization so you get to keep usage of the source tracker at the same time as switching over to the new instance. A smooth transition without any downtime whatsoever. You're in full control at all times. Once all users have moved over to the new system and all required project data is synced, you can safely stop the synchronization.

[TRY OUT OUR SYNCHRONIZATION TOOL EXALATE](#)

5. The Centralized Master Instance

Your working environment consists of a master parent issue tracker and several child issue trackers. Issues and tickets get raised in your parent issue tracker, then delegated downstream towards development teams. These separated teams have their own responsibilities and tasks, each one preferring to work in their own issue tracker instance, managing their own sets of issues.

Master Instance synchronization

The challenge;

Communication. When creating issues and assigning tasks through the 'master' instance, there is no live view of progress status and new data. Once progress is made, manual reporting is required which leaves room for error, information getting lost in translation and delays. Jira integration type unknown Often, managers using the centralized instance need to be able to have a full view of all issues in order to be able to set priorities for the issues.

The solution;

When you synchronize the centralized issue tracker with all child instances, you see one live “master view” of all progress. Allowing you to easily prioritize tasks, but allowing all teams to keep autonomy of workflows and processes. synchronized by master instance

[TRY OUT OUR SYNCHRONIZATION TOOL EXALATE](#)

Synchronization - the answer to a multitude of problems

We know that the concept of syncing multiple instances of issue trackers is not without concern. But most of the presumptions we come across can be easily resolved. ***Will you lose time or require extra work to be completed by implementing synchronization? Will workflows remain accurate and consistent? Is acceptance from internal teams and external subcontractors, suppliers and clients a valid concern?*** With tracker synchronization there will practically be no training needed for issue teams, and very little time needed to implement. Plus you'll quickly see results;

- Infinitely better cross-team collaboration
- Vast time savings
- Significant reduction in error
- Better control over a previously uncontrollable environment.

So you know it's the right thing for you, but how do you ensure buy in from the right stakeholders? The best synchronization tools are autonomous across existing systems and vast improvements to project visibility and issue resolution times can be made with minimal effort. There doesn't need to be disruption to their current processes, and this better way of working will be quick to see. If you're aiming to scale up your capabilities you *need* to consider synchronization as part of your competitive advantage. Why? **Separated project data inevitably leads to: 1) more errors 2) more time wasted 3) more confusion = more problems!** Remember, the very concept itself is to turn complexity into something simple and flexible. By adopting a synchronization

tool, everyone involved in the process is gaining access to relevant, customised and accurate information. **We call it the issue tracking utopia.**

Summary

Exalate is the most flexible issue synchronization tool on the planet, and is the right fit for the most advanced use cases concerning synchronization across multiple trackers. Our goal is to lead the way for organizations who need to be agile and scale their issue tracking capabilities. We aim to achieve this through our vision to enable a global network of connected companies to work together at the flip of a switch. We can help you address the need for better internal processes, whilst supporting your clients, external suppliers and subcontractors. Curious how synchronization can help your use-case? Have a look at [Exalate](#) or [get in touch](#). ***Did not find the use case you were looking for? We've covered a lot more use cases on our documentation site [here](#).***

[TRY OUT OUR SYNCHRONIZATION TOOL EXALATE](#)