Crowdsourcing triage: A useful pattern for Jira issue ownership

When a new issue is created, who owns it? The traditional answer is 'nobody, until it is claimed'. Here we suggest a better model, where ownership is collective until claimed.

The traditional support queue

Say we are using Jira for an internal IT helpdesk. To respect the nomenclature, let's refer to issues as tickets, and the workers as agents.

When a new ticket is raised, who gets assigned initially? More broadly, **who initially 'owns' a new ticket?**

The most common answer is: **Nobody owns new tickets** until they are assigned. There is a support queue, sorted by urgency or some metric, and agents pick their next ticket off the top.

This is the mode Jira Service Desk assumes. At Atlassian I worked as a support engineer following this model for about 5 years.

Group ownership

The alternative answer is: **The whole group owns new tickets**. It works as follows:

- Create an email distribution list for your helpdesk team, so email to helpdesk@yourcompany.com goes to all agents.
- Create a Jira account called `helpdesk` , with email address helpdesk@yourcompany.com.
- Make `helpdesk` get assigned new tickets in the IT project.
- Ensure that the Assignee gets notified of Issue Created and Issue Commented events.

So now all agents will be emailed when a ticket is initially created. The ticket is notionally owned by the whole group. So any agent is free to comment on the ticket, to clarify the problem or ask for more details, *without implicitly claiming ownership.*

When an agent does commit to 'owning' the ticket, **they reassign the ticket to themselves**. Emails are no longer sent to the `helpdesk` group.

That's it.

Advantages of initial group ownership

There are a few reasons why this assign-to-a-group pattern is IMHO better than the standard assign-to-nobody pattern: **knowledge sharing**, **avoiding fear of claiming ownership** and **email-centricity**.

Let's imagine we have 3 employees dedicated to support. A new ticket comes in, describing a tricky problem: a product bug is occasionally causing weird stacktraces.

Luckily, Sally the Support Agent has seen this before: she spent all last week helping another user isolate the problem.

Now what happens to our ticket?

In the standard queue pattern, agents pick tickets as they become free. If we're lucky, Sally is free, assigns the ticket to herself and solves the problem in record time. If we're unlucky (as is likely), Jimmy the intern will be free next, and he will spend the next week re-discovering what Sally already knows.

What happens if we implement the 'group ownership' pattern?

When the ticket is created, everyone on helpdesk@ gets the email. All agents are expected to at least skim-read new ticket emails. While reading, Sally gets that *deja-vu* sense. She jumps online, comments "I think I saw this bug last week. Did you use the frobniz feature? Can you send us the so-and-so logs?", and then gets back to her currently-assigned ticket.

Although Sally has commented, the ticket is still assigned to the group. She hasn't claimed ownership (she can't right now, being busy). The other two agents get emailed her comment, and learn (by example) that this problem has been seen before, what diagnostics to ask for. Even if Sally never gets assigned the ticket directly, she has directed enquiries on the right path.

Sally was assigned other tickets. She only commented on the new ticket because she was free to do so without committing. **By embracing the lack of initial ticket ownership**, we empower people to contribute without commitment, share knowledge, and collectively respond faster.

Another nice feature: people only peripherally involved, like IT managers or senior developers can be subscribed to helpdesk@. Perhaps Bob the Developer actually fixed the bug a month ago – he can instantly recognise the symptoms and chime in. IT managers and developers live in email – they won't be monitoring a support queue. **Because initial triage relies on email, peripherally involved people who can get involved**.

Summary

In short, we crowdsource the initial triage of tickets, leading to natural knowledge sharing, faster resolutions and some quick wins. If the wisdom of the crowd doesn't deliver, we fall back to the usual process of agents assigning themselves tickets.