Why self-host JIRA and Confluence?

When buying Atlassian products, you have a choice: host JIRA and Confluence on your own hardware, or let Atlassian do the hosting:

Should you self-host, or go with Atlassian Cloud hosting?

It is worth considering for a moment how unique this is. The industry has moved strongly to a SaaS, subscription model. Virtually every new product management app these days is SaaS-only. Companies *love* it when they have your data, a steady income stream, and fewer environments to support. Customers mostly like SaaS too - who wants to spend their time maintaining software?

So why do **we strongly recommend self-hosting ?** In short: if you don't self-host, you miss out on the plugins, extensibility and integration possibilities that make Atlassian products really great.

More Plugins (aka Add-ons)
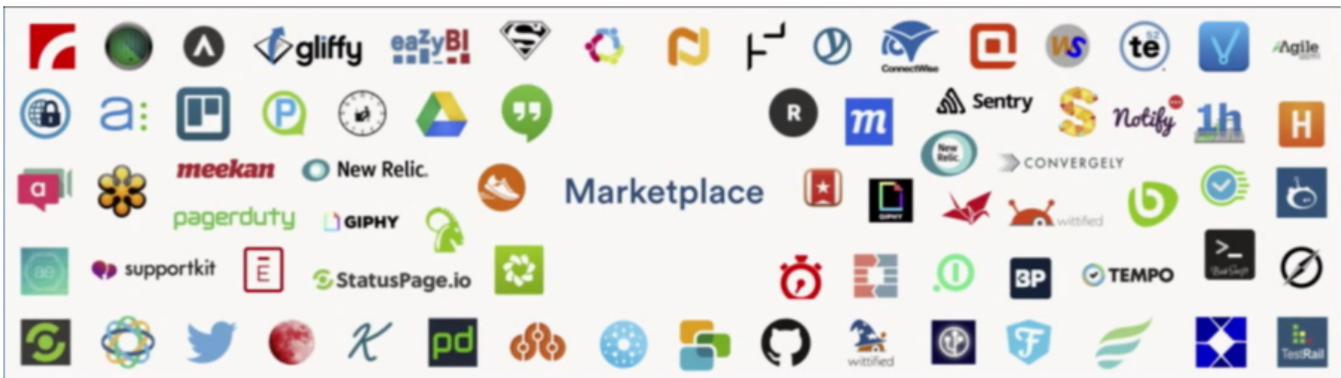
JIRA and Confluence have a thriving plugin ecosystem, visible at
https://marketplace.atlassian.com.

Plugins are why you can't go wrong with JIRA and Confluence: you're not just buying a product, you're buying into a platform upon which hundreds of companies are madly building and making lots of money.

The dirty secret is this: **most plugins don't work on Atlassian's Cloud-hosted** version. Here are logarithmic graphs showing plugins by popularity (as per marketplace.atlassian.com stats), excluding unmaintained plugins incompatible with the past 3 releases.

JIRA Plugins by popularity, Jan 2016
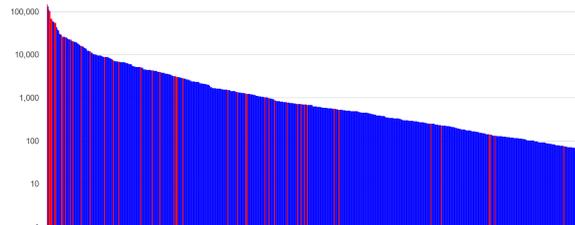
Confluence Plugins by popularity, Jan 2016
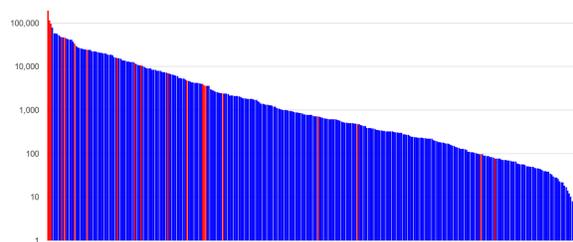
Blue plugins are self-hosted only, red are cloud-ready, and yellow are cloud-only.

There is not a lot of red.

Things are improving for Cloud customers. Atlassian is pushing it's new cloud-compatible 'Connect' plugin framework, and in the 2015 Summit, Mike notes that Cloud-compatible plugins have grown from 50 to 300 in a year. If you consider only the top 50 plugins, the situation looks better:

JIRA Top 50 Plugins by popularity, Jan 2016

Confluence Top 50 Plugins by popularity, Jan 2016

However, given that:

- migrating a plugin to the Connect framework requires a complete re-architect
- Connect plugins are severely limited in what they can do
- Connect plugins require a vendor-hosted component, putting a burden on the plugin vendor

it seems likely that outside the money-making top 50, the "long tail" comprising the vast majority of plugins will never be Connect-enabled.

More Integration Possibilities

This applies especially to Confluence. Your wiki does not exist in a vacuum.

- Want to reuse user accounts for your corporate Active Directory or LDAP? No can do, though to be fair, Cloud-hosted does offer Google Apps integration.
- Got other SQL databases? By running JIRA and Confluence, you already have two. Did you know that Confluence makes an awesome ad-hoc reporting platform? Have a look into the SQL, PocketQuery, and Play SQL plugins for Confluence, and Database Values and Arsenale Dataplane plugins for JIRA. For instance, let's make a live 'Top 50' chart from a plugin stats database, using Pocket Query:

app	cloud	name
Change Count	Vendor	

  This is only possible if you self-host.
- How about embedding HTML or Javascript, e.g. an iframe of your latest build results, or widgets on the web

A Javascript clock widget	Javascript pulling JSON from the web to render a table
&nbsp;	**Latest product versions**

More extensibility

- The second most popular JIRA plugin, ScriptRunner, is not available for Cloud. Script Runner (aka Groovy plugin) adds a fantastic degree of extensibility to JIRA. One can script up new custom fields, new behaviours (e.g. show field X if field Y has *this* value), run arbitrary code on workflow transitions, expose custom REST endpoints, and so forth.

> ⚠️ Adaptavist now have a 'Cloud' version of ScriptRunner, but don't be fooled: the Cloud version is virtually worthless, lacking proper scripted fields, Behaviours, REST endpoints, and workflow conditions/validators.

- The whole wonderful world of Confluence user macros is denied to Cloud-hosted customers. User macros are a powerful mechanism whereby administrators can create custom wiki macros, such as the Timezone macro above.
- There is no customizing Confluence appearance with themes, either via simple tweaks or via complete custom themes (such as this website).
- No custom HTML in JIRA or Confluence (particularly useful embedded in JIRA field descriptions).

Backups

Most data loss is caused by user error - an admin accidentally deletes the wrong space or project, or performs a bulk update on Jira issues that goes wrong.

If you self-host, you'll probably have backups, and can restore whatever needs restoring.

If you are on Cloud and delete a space or delete a project, you're out of luck: Atlassian have backups, but they're not going to restore them for you. Is there a difference between a backup you can't restore and no backup at all?

Avoid lock-in

With self-hosted, *your data is yours*:

- With self-hosted, your data behind the company firewall, on your own domain name (

  🔖 **CLOUD-6999** - Allow custom domains for Cloud apps `IN PROGRESS`) With Cloud, your instance is public at a https://yourcompany.

  atlassian.com URL, no 2FA, not even restricted to an IP range (🔖 ~~**CLOUD-2636**~~ - Allow customers to configure IP allowlist `CLOSED`).

  You are one guessed password, zero-day exploit or fat-fingered permissions mishap away from being compromised. Atlassian have a good security track record, but still issue frequent security advisories. It would be naïve to believe hackers don't have a few more up their sleeves. If your Cloud-hosted site was compromised, would you ever know? You have no access to the server logs, and Atlassian won't know or care if hundreds of requests start coming from China.
- With self-hosted, Atlassian licenses are perpetual. If one day you decide to never upgrade JIRA or Confluence again, you never need to pay Atlassian another cent. With Cloud-hosted, you will be paying Atlassian forever.
- With self-hosted, you can access the database directly. This benefit is difficult to explain, but it is huge. You can, with standard SQL, theoretically perform any operation that JIRA or Confluence can. You can search your own data with SQL in ways Atlassian never dreamed of. For instance, "show me every Confluence page containing a particular macro", or "Generate invoice line items from JIRA worklogs in this project". When things break, you can see why in the database, and often fix the problem (Atlassian support frequently ask customers to run custom SQL). Your data is yours not only theoretically, but practically, not restricted by Atlassian's UI or REST API. Relational databases are the great leveller, returning control from application programmers to end users.
- With self-hosted, you get to decide when maintenance is done. You decide when to upgrade to the next release, rather than being on the bleeding edge.

Note that it is possible (with some effort) to move from self-hosted to cloud, and cloud to self-hosted

Conclusion

All these Cloud hosting limitations are itemized on the Restricted functions in JIRA Cloud applications page (and children), but the implications are not spelled out. Essentially, new Atlassian Cloud customers don't know what they're missing. They get the products, good in themselves, but without the extensibility and ecosystem that makes them truly great. Frustrating limitations are encountered one by one, by various users, and a desire for change never overcomes corporate inertia.

Put another way: Cloud has undeniable benefits and shortcomings, but these are asymmetrically felt: the benefits are financial and up-front (reduced IT burden), liked by managers, whereas the shortcomings are unclear, discovered over time as daily frustrations and paper cuts, felt by dis-empowered end users who have no say in purchasing.

If your company wants the full flexibility of self-hosted, the costs come at four levels:

- **Infrastructure** – $40/month for a VPS if you don't already have your own servers
- **Installation** – the one-off cost of server setup, including tasks like establishing off-host backups, monitoring and documenting disaster recovery procedures
- **Basic systems administration** – keeping a server running, responding to downtime alerts and so forth. Companies such as such as Contegix and many others offer such "managed hosting" for Atlassian apps, but in truth, JIRA and Confluence rarely cause problems once set up, and a typical company's IT department can handle these tasks quite well.
- **Product upgrades** – these can indeed be painful when plugins and external integration are involved (which is why Atlassian forbids them in Cloud-hosted). Most upgrades go smoothly, but every now and then some critical problem arises when a really deep knowledge of the product is needed.

- **Specialist support** – Very rarely things do break badly, and one needs a cross between a systems administrator and Atlassian product expert on-hand to fix.

Red Radish offers services which we like to think gives you the best of both worlds:

- **Installation** – we can set your server up well from the beginning, *fast* because we've done it before, saving you money (if you value your time). If you're already on Atlassian's Cloud hosting we'll be happy to handle the transfer process to minimize downtime.
- **Basic systems administration** – we provide runsheets and optional training that let regular IT people do daily administration. In addition our Knowledge Base is there to help IT administrators.
- **Product upgrades** – Red Radish Consulting offers bi-yearly, quarterly or monthly upgrade services.
- **Specialist support** – Red Radish Consulting offers support contracts, where for a monthly fee we are available for anything from low-level systems emergencies (e.g. restoring backups to a new system following hardware failure) to product usage questions.

Please contact us if you'd like to discuss, or find an Atlassian Expert near you.