

Proposal: CloudStack Primate UI

Summary

I have been working on a new, modern role-based UI for Cloudstack (project Primate: <https://github.com/shapeblue/primite>) I demoed this for the first time at CCCNA19 last week and it was very well received. It was discussed, at length, as an item in the hackathon and the general consensus there was that this could become Cloudstack's new UI. We discussed a plan to achieve that and now I'm bringing that plan to the list for discussion.

Background

The current CloudStack UI has grown technical debt over time and it has become harder to extend, develop, maintain in the long run, it is also difficult for new contributors to learn and get started. Since late 2018, I started working on a side-project with the aim to create a modern progressive and role-based declaratively-programmed UI for CloudStack, called Primate. Before creating Primate, I set out to create a list of core requirements of what would give us an extensible, modern UI that was easy to develop now and in the future. These are the requirements I came up with:

- * designed from ground up to be a complete replacement for our combined user/admin UI
- * to respect all entities in cloudstack and not make assumptions on use-cases of cloudstack
- * data-driven and auto-generation of UI widgets and to be easy to learn, develop, extend, customise and maintain.
- * declarative programming
- * support for API discovery and parameter completion like CloudMonkey
- * support for custom roles

I looked at existing Cloudstack UI projects but none of them fully satisfied all these requirements and started Primate.

Project Primate

For the implementation, I compared a couple of opensource JS and UI frameworks and decided to use VueJS (<https://vuejs.org>) which is a JavaScript framework and AntD (<https://ant.design>) which is a UI design language with a well-defined spec, styling guide, and an implementation-specific to VueJS. VueJS was selected because among a few other JS frameworks I surveyed it was the easiest (for me) to learn and get started. I also surveyed a few UI frameworks and selected AntD because it came with a well-defined spec, styling guide and VueJS specific implementation which gives several re-usable components out of the box.

During the development of Primate, I used my previous experience from CloudMonkey and another PoC angular-based UI ProjectX, and it currently supports:

- * role-based UI based on API autodiscovery
- * auto-generated action/API forms with parameter completion
- * declarative component-driven views
- * modern programming methodologies (hot reloading, npm based build/run/compile etc.)
- * decoupled from core Cloudstack code
- * dynamic translation (most/many of old translation files ported)
- * includes dashboards, async job/API polling, all list views/tables per the old UI
- * browser history and url/route driven navigation
- * support for mobiles/tables/desktop screens
- * configuration driven UI customisation (of navigation, icons, APIs etc)

To get to this point, I've had some valuable help from Anurag and Sven et al at EWerk.

The development strategy to support all APIs out of the box in a data-driven way gives a functioning UI and scope to customise views/components over time. Support to declaratively add actions on all views and some action/customisation (ex. vm/zone deployment wizard) is in progress. The work was recently discussed and demo-ed at CCCNA19, the slides of my talk are here: <https://rohityadav.cloud/files/talks/cccna19-primite.pdf>

Discussion

With this discussion thread, I want to propose the idea hatched in the recent hackathon for Primate to become the next UI for CloudStack. I'd like to seek everybody's thoughts, feedback, comments, and reviews on that idea.

Obviously, part of any new UI project would mean that the existing UI development would eventually get abandoned. In the hackathon we discussed a potential plan to formally deprecate the old UI. I therefore think that this is something that needs discussion here and a vote if its going to happen.

In summary, that plan is as follows (with proposed timeframes in [brackets]) :

- * allow the community to download & experience the UI
- * discuss and gather consensus around Primate [now]
- * take a vote on whether to take this plan forward [next few weeks]
- * create a new repository under Cloudstack and accept Primate codebase, work on initial documentation with getting started, styling guide etc (i.e. make this part of the cloudstack project) [after the vote, if agreed]
- * collaborate to get primate to be functionally on-par with current CloudStack UI, gather feedback, issues, iterate development/testing etc. [2-3 months before winter LTS release]
- * release Primate with winter LTS 2019 release as a technical preview and also give notice of deprecation plan for existing UI
- * define a release approach (separate repo/releases with separate rpm/deb repo, or use git subrepo/subtree/submodule with CloudStack repository?) [Before winter LTS release?]
- * optionally stop taking feature FRs in old UI and create upgrade path/plan documentation from old UI to Primate
- * Release Primate GA in summer LTS 2020 release as the new UI, but still ship old UI with a final deprecation notice
- * old UI reaches EOL in Winter LTS 2020, old UI codebase is removed from codebase (this gives one year period from winter LTS 2019 Primate technical preview for community and users to discuss/comment/upgrade)

Per the proposal, the last release shipping the old UI would be in summer 2020 LTS. According to our LTS schedule, this would be supported under LTS until summer 2022 (but, with no new UI functionality after summer 2020)

The repository is currently available here: <https://github.com/shapeblue/primate> and has instructions on getting started – I'd encourage everybody to look at the work to date.

If anybody is keen to contribute in the meantime, I can receive bugs/requests/feedback as new issues: <https://github.com/shapeblue/primate/issues>

Links:

Git repository: <https://github.com/shapeblue/primate>

Issues: <https://github.com/shapeblue/primate/issues>

Screenshots: <https://twitter.com/rhtyd/status/1171224978714288129>

CCCNA19 talk slides: <https://rohityadav.cloud/files/talks/ccna19-primate.pdf>