This (DRAFT) document is to capture issues facing LocationTech projects related to incubation.

This topic is especially important as LocationTech is approaching its 3rd birthday. So many things about the LocationTech community and working group have been hugely successful. A great collection of projects & especial for geoprocessing, the Tour, the Intern program, the newsletters, collaboration between projects & members, membership growth, highly innovative members joining, very high profile members joining, and more. As some of of the project leads said, LocationTech “is killing it!” (i.e. doing extremely well).

However, as of January 2016, no projects have released software from LocationTech.org yet. The optics of this are problematic and especially for advancing everyone’s desire to have the LocationTech brand be strong and synonymous with high quality spatially aware software. Addressing this is the #1 priority for LocationTech in 2016.

It is important to note the project communities have been growing and seeing increasing contributions and adoption.

This document was crafted by Andrea Ross, Director of LocationTech. Others are welcome to contribute by editing, commenting, or providing feedback.

In no particular order, the issues are:

The fog of Eclipse process

The automation & techniques developed by the Eclipse Foundation to help manage workloads when nurturing 300’ish projects helps when efficiently taking on the 301’st project with experienced project leads and committers. As one would reasonably expect, incoming projects (ones not already familiar with Eclipse process) are very dependent on mentorship. Unfortunately the formal mentors for LocationTech have been pretty much completely absent. This might not be that surprising as mentors must be from the architecture council, which has been populated by developers from the eclipse platform, modeling, and other areas that aren’t connected in any way to LocationTech.

At times, without clear awareness of the path ahead, some LocationTech projects have been somewhat paralyzed and unsure what to do. This has slowed progress and eroded moral.

Recommendations:

Mentor-the-mentors from within LocationTech. Jim, Rob, and Jody are joining the Architecture Council, and will be eligible to be mentors for LocationTech projects. Helping them ramp up as gracefully as possible is a priority. Support them and treat them well if they take on mentoring other projects.

We are hiring the first dedicated employee for LocationTech in June. An important part of their duties should be to help guide projects through the onboarding process.

Staff & mentors, please remind ourselves often that newcomers don’t know what they don’t know so they can’t ask about it. A proactive rather than reactive approach may help a great deal. For example, a well timed hint/suggestion/overview/primer makes a huge difference and is worth the time to offer it proactively and save on wasted effort or a reactive response later.

The Eclipse Foundation has had staff do bootcamp sessions at EclipseCon over the years. Doing one at FOSS4G NA in May would be beneficial for LocationTech projects, leads, committers.

It might be worth considering that new projects require much more staff assistance than old and adjusting accordingly.

Estimate the very difficult to estimate

The management teams of developers working on LocationTech projects are asking for a clear start date, end date, and rough estimate effort involved to incubate. Developers, and especially those new to LocationTech, have no idea because they may not have given rigorous thought to their dependency chain before, and can't predict if issues will be found. The fog of process inexperience is a big factor here too. For these and other reasons, estimates have no credibility.

The IP team cannot know what they're facing until they dig into the code and look at things. They cannot provide estimates for completing IP review with any kind of credibility.

Members and people bringing projects to LocationTech are unable to reliably estimate the effort and time it will cost to graduate incubation.

Recommendations:

TBD: This is tricky! Ideas we very welcome.

Noted that IP is a huge part of this.

In time, a large archive of already approved libraries for LocationTech will be available. It is already becoming the case today with hundreds of CQ's reviewed and approved. As this happens, it becomes an invaluable resource to the community.

The PMC will become an increasingly valuable body to work with & advise on potential libraries the community should use.

For what it's worth, the Eclipse Foundation's Executive DIrector Mike Milinkovich has shared in the past that no project that wanted to graduate has every failed out. Yes, there can be much work to do such as replacing a rejected library with another, but many many projects graduate and do releases all the time. At times, LocationTech projects needed to be reassured as they felt there was a chance they might not ever graduate.

Suggestion: Ability to identify key choke-points (or "decision point") CQs with project leads and IP team. Due to the nature of transitive dependencies some CQs are further up the food chain and any any feedback will have a knock-on-effect invalidating dozen's of CQs. The ability to identify these choke-points could be of strategic value in reducing workload for both incubating projects and the IP team.

Ships passing in the IP process

Developers from the LocationTech community have tended to ask for blocks of time from their employers to be allocated to work on incubation. These blocks of time are hard won and precious. See the previous section about the challenges for estimating time required.

The IP team has a huge backlog of CQs, and blocks of time to work on a specific project's CQ's is hard won and precious. The same challenges for estimating affect them.

When the blocks of time for the developers do not align with the blocks of time for the IP team, sometimes bad things can happen. The IP team hits the wall without answers to things and needs to move on to other projects. Coming back to a project later is hard as they've lost the current knowledge and familiarity. For developers they raise a bunch of CQ's, only to see them sit because the IP team is (justifiably) focused on something else. After a while they are stale and no longer reflect current reality and need to be replaced with a new CQ. Frustrations reach a boiling point.

Recommendations:

TBD, thinking about this as it is tricky, and very important.

The stable branch suggestion from below may help here too.

Hiring the extra person for the IP team helped a great deal. The team recently lost a person and that will have an impact in the short term.

LocationTech has been experimenting with choosing to focus on certain projects, even though it meant leaving others to wait. This was the right call and has enabled GeoMesa, Spatial4J, and a few others to get into the position to do a release. This likely won't the the long term mode of operation and it does spend precious goodwill with the projects that get stuck waiting.

Coding in the fast lane

Re-using a library rather than writing something yourself just makes a heck of a lot of sense. There's a cost that can sneak up if a library has a large transitive dependency chain, or nasties lurking somewhere in the dependency chain that cause it to be rejected.

In areas of particularly high innovative velocity, the services and libraries underpinning projects change fast. Keeping current is essential. This can be overwhelming from an IP review perspective.

Estimates both from developers and the IP team regarding how long it will take to IP review a project are often not credible. This causes serious grief to the management teams of developers.

Recommendations:

It is worth taking the time to give a sober consideration and once over to try and assess libraries before using them. This is a good task for the PMC to help with. Best case is to talk with the PMC proactively. In time, it is likely LocationTech developers will get good at factoring this. We'll also get a larger library of stuff we've approved previously.

Consider using a stable branch. When development gets to a reasonable pause point, push to the stable branch, and do IP review and a release based on that branch. The development or unstable branch can go fast & furious. More often than not, customers want to use a stable branch and upgrade carefully at fairly spaced intervals.

We have taken action upstream to extend the stable branch of GeoTools from 6 months to one year - specifically for the LocationTech incubation process.

Where it makes sense, consider if some technologies are appropriate for being classified as exempt prerequisites. For example, Hadoop, Accumulo, Spark were declared exempt prereqs. This means the IP team does not need to scan each version and variant. This approach eased workload tremendously.

When you’re not at rock bottom

If your technology project is very low level with few dependencies below it, you generally have an easy go of IP review and thus pretty good autonomy for when you do releases. However, so much of the open source software world simply isn’t like this. If you happen to be higher up in the technology stack, so many things are moving beneath you. You can pick your dependencies, but their dependencies are largely beyond your influence. If the entire technology stack from bottom to the top is hosted at LocationTech, than life is much easier. But this will likely be very rare for any Eclipse Foundation working group, and certainly as they are starting out.

A dependency approved for use on Orbit is not immediately available in a Maven repository. Our ability to handle IP subsets is vastly reduced.

Recommendations

Encouraging other projects, and especially your dependencies to consider joining & hosting with us helps bigtime.

Pick your libraries super carefully and mindful of the costs associated as discussed above.

While we have an approach in mind for dealing with maven and IP subsets it is not yet proven in practice.