

Defining Development Goals: LaunchPad complements SourceForge

Egon Willighagen 

Published February 2, 2008

Citation

Willighagen, E. (2008, February 2). Defining Development Goals: LaunchPad complements SourceForge. *Chem-bla-ics*. <https://doi.org/10.59350/nks4k-n7e69>

Keywords

Cdk

Abstract

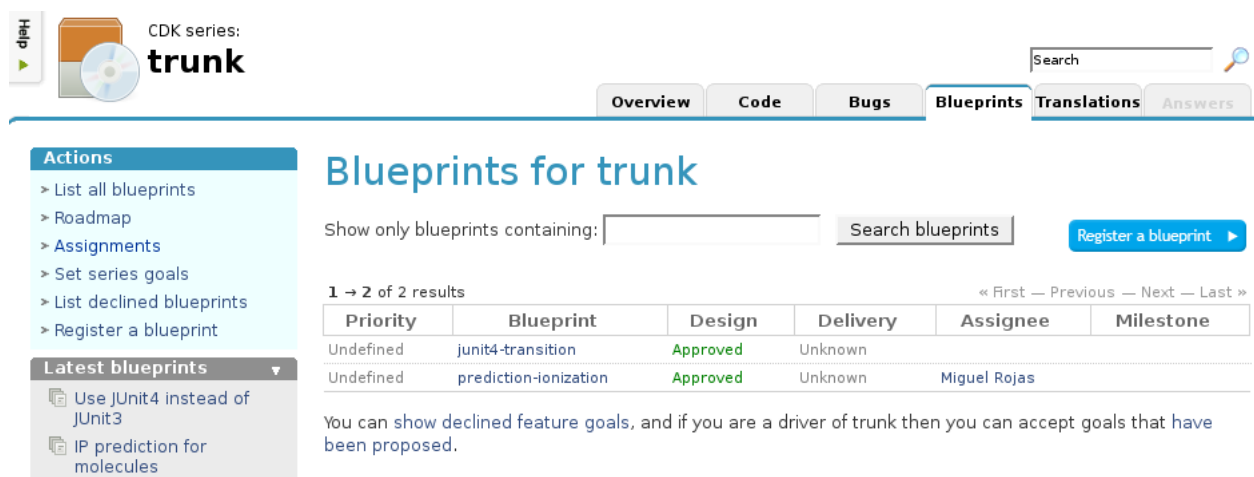
Today, Miguel (who made the 10000th CDK commit) and I gave LaunchPad a go, because it offers a nice GUI for planning and monitoring source code development. We have set up a CDK team and a CDK project. LaunchPad has overlap with SourceForge functionality, but the idea is not to duplicate functionality. Moreover, we do not translate the CDK either, so that LaunchPad functionality is not useful either. Not for the CDK at least;

Copyright

Copyright © Egon Willighagen 2008. Distributed under the terms of the [Creative Commons Attribution 4.0 International License](#), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Today, Miguel (who made [the 10000th CDK commit](#)) and I gave [LaunchPad](#) a go, because it offers a nice GUI for planning and monitoring source code development. We have set up a [CDK team](#) and a [CDK project](#). LaunchPad has overlap with [SourceForge](#) functionality, but their idea is not to duplicate functionality. Moreover, we do not translate the CDK either, so that LaunchPad functionality is not useful either. Not for the CDK at least; maybe for [Jmol](#) and [Bioclipse](#)?

However, we are interested in the task management system of LaunchPad. While the CDK project is currently maintaining a [Project Maintenance Tasks](#) tracker, it does not have the feature richness of the LaunchPad equivalent. The latter allows us to link tasks with series goals. We currently basically have two series: the `cdk1.0.x/` branch, and `trunk`. Miguel and I have been working on getting the [ionization potential prediction](#) in `trunk` working, which involves about all the code Miguel wrote during his PhD thesis with [Christoph](#). And, this is one of the goals of the next stable CDK series (replacing the 1.0.x series). This is something we can easily define in LaunchPad:



CDK series: **trunk**

Overview Code Bugs **Blueprints** Translations Answers

Search

Actions

- > List all blueprints
- > Roadmap
- > Assignments
- > Set series goals
- > List declined blueprints
- > Register a blueprint

Latest blueprints

- Use JUnit4 instead of JUnit3
- IP prediction for molecules

Blueprints for trunk

Show only blueprints containing: Search blueprints Register a blueprint

1 → 2 of 2 results « First — Previous — Next — Last »

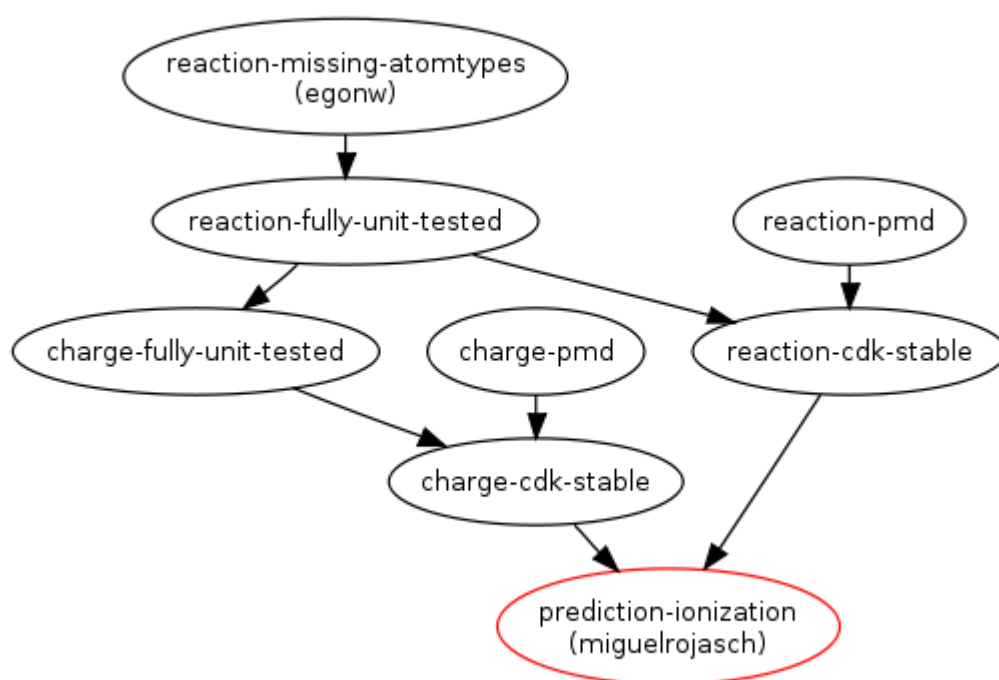
Priority	Blueprint	Design	Delivery	Assignee	Milestone
Undefined	unit4-transition	Approved	Unknown		
Undefined	prediction-ionization	Approved	Unknown	Miguel Rojas	

You can show declined feature goals, and if you are a driver of trunk then you can accept goals that have been proposed.

Getting the IP-prediction code updated for the new CDK atom types and other changes, and making it CDK stable involved quite a long list of tasks, which shows dependencies. For example, I can't continue [cleaning up the partial charge prediction code](#), before the [resonance structure generator in the reaction module](#) is working properly again. This in turn depends on me adding missing radical and charge atom types, which in turn depends on expected atom types, which Miguel had to implement. And this last is actually what he was committing around the 10000th commit.

Now, Miguel and I will try to manage this development in `trunk` using LaunchPad. It allows us to define all these smaller tasks, but, more importantly, the dependencies between them:

Dependency tree



* Blueprints in grey have been implemented.

As such, LaunchPad gives us the means to manage this complex development. It shows up what we're facing, how far we have progressed, and much, much more:

Actions

- > Edit title and summary
- > Change people
- > **Change status**
- > Change priority
- > Edit whiteboard
- > Propose as goal
- > Target milestone
- > Request feedback
- > Subscribe yourself
- > Subscribe someone else
- > Link to bug report
- > Link branch
- > Offer mentorship
- > Add dependency
- > Show dependencies
- > Propose for meeting agenda
- > Mark superseded
- > Retarget

Change status

Definition Status (Required)

Approved

The current status of the process to d

Implementation Status (Required)

Unknown

- Unknown
- Not started
- Deferred
- Needs Infrastructure
- Blocked
- Started
- Slow progress**
- Good progress
- Beta Available
- Needs Code Review
- Deployment
- Implemented
- Informational

chem-bla-ics

This goes well beyond what SourceForge has to offer; this will be an interesting experiment. I do not anticipate dropping SourceForge at all (just in case you were wondering...); they have served as generally very, very well; and completely free too! (LaunchPad is free too) As far as I can see, they form a perfect complement. Like a ligand and an enzyme, like opensource and [open notebook science](#), or like a Mammoth and an ice field.

Speaking about ONS... [Jean-Claude](#), not sure if LaunchPad would be open to projects without source code too...