PeeringDB General Meeting 2024 – Product Committee Update

The product committee is responsible for improving PeeringDB's software as a whole, from bug fixes and small tweaks through the planning execution of large changes to the web ui, API, peeringdb-py. The committee sits between users, who ask for improvements, and the developers, who implement them.

People changes

Big thanks to Steve McManus for chairing the committee until earlier this year, and for helping me get off the ground. Jeff Bartig, Paul Hoogsteder, and I joined the committee late in 2023. Thanks also to Leo Vegoda for helping me get up to speed in the Chair position.

Key Numbers and Process Issues

We deployed **12** releases in 2023, addressing **76** issues, **11** of which were bugs.

We currently have a queue of **54** issues fully defined and ready for development.

We are getting ready to roll out automatic data syncs from the prod site to **beta.peeringdb.com** as part of a set of changes to improve the quality of deployed features, in part by encouraging more users to use the beta site for their day-to-day work.

New Objects

We introduced two new items to the object model in 2023:

- The **Carrier object** describes networks providing high capacity links between facilities. A new Carrier object permission just arrived this month.
- The **Campus object** defines a set of **facilities** where inter-building cross-connects are available. The HOWTO for facilities now explains how to create a campus.

As of this AM there are **49** campus objects and **161** carrier objects defined.

Data Access

We introduced new search functionality, backed by ElasticSearch, which is now the default. It allows users to search for things like "in a city name" from simple search. The Advanced Search features remain and continue to be developed.

We also started publishing data defining all interconnection facilities as a daily .KMZ dump, which has been very popular. We are planning the ability to export Advanced Search results as a .KMZ download soon.

Data Quality

We introduced a way for an internal source of truth to propose updates to PeeringDB. Because it only makes proposals, it doesn't need to be trusted with credentials. An authenticated human user then chooses which updates to accept or reject. This lowers the friction of keeping PeeringDB current and accurate.

Web UI

We are nearly ready to deploy a new web front end, which is intended to retain the simplicity we have now while incorporating better support for the myriad use cases and data growth we've seen. The new setup will be available at **preview.peeringdb.com**, so users have an opportunity to test and provide feedback before we deploy to the main site.

Volunteer developers

We're very glad that volunteers can contribute to PeeringDB! We improved the development quickstart environment a couple of years ago, and volunteers use it - I myself was able to get a PDB instance up and running, complete with real data, in about 20 minutes. We'd like to acknowledge 2023 contributions from:

- Carlos Aguado
- Daniel Van Allen
- Todd Crane
- User "kiraum"

New volunteer developer this month: Arturo Baldo

Thanks to all PC members

…without which none of this would be possible: Jeff Bartig, Yan Berthier, Yolandi Cloete, Matt Griswold, Martin Hannigan, Peter Helmenstine, Paul Hoogsteder, Laurent Jarbinet, Steve McManus, Arnold Nipper, and Leo Vegoda.

Questions?