



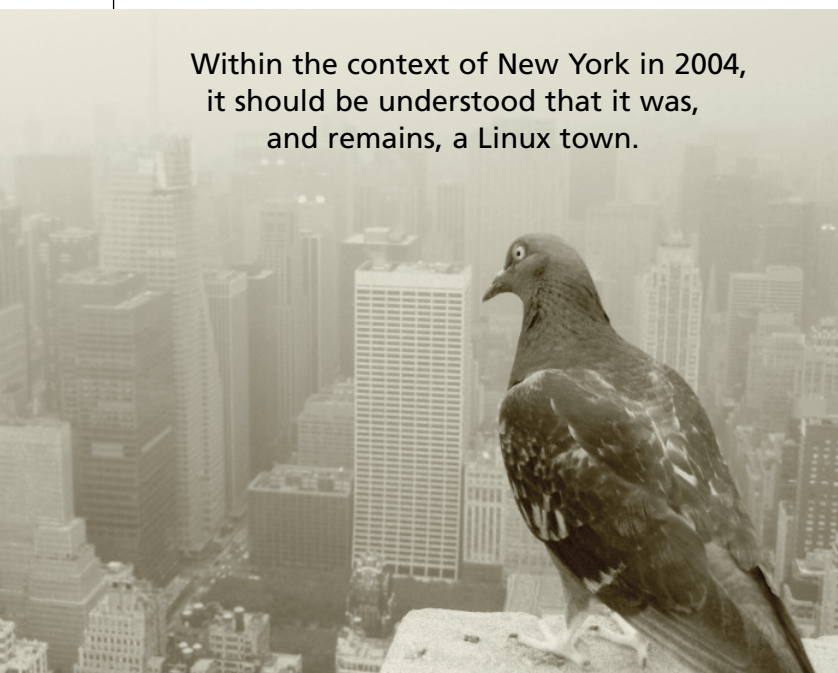
# Starting a BSD User Group

**A handful of like-minded BSD users launched the New York City BSD User Group (NYC\*BUG) in 2003. NYC\*BUG meets monthly and hosts multiple mailing lists, while managing to provide services to the BSD projects and other user groups.**

The user group's chronology is filled with the waves and troughs that any technical user group faces and to summarize instructive lessons is difficult. However, I will attempt to review the life of NYC\*BUG and present some useful conclusions for others interested in starting a BSD user group (BUG) of their own.

In December 2003, several of us planned to officially launch in January 2004 at what was to be the last Linux Expo in New York City. It was understood that there would be opportunities to engage the broader open-source community at Linux Expo and conduct a birds-of-a-feather ses-

Within the context of New York in 2004, it should be understood that it was, and remains, a Linux town.



sion (BoF), culminating in our first meeting in early February.

Some 50 people attended the BoF, and our early meetings were large and engaging. Discussions continued into the early morning hours, and we began to witness how a technical user group could corral people and begin to have an impact.

## ***The Unwelcoming Committees***

Not everyone in the local community was enthusiastic about our launch. NYC always entertains a myriad of open-source user groups, from broad tents with heavy corporate influence to others that verge on cult status. Yes, cult status.

The local Unix group, in particular, felt we should have run the idea of a BSD group by them before proceeding. We sought neither permission nor forgiveness.

Somewhat more humorously even before the BoF, one post on January 13, 2004, to the NetBSD NYC regional list dismissed us: <http://http://mail-index.netbsd.org/regional-nyc/2004/01/13/0003.html>.

"i wonder if beer is involved? there's no mention of it in that posting."

"they are just doing what the linux group does... probably same ppl too."

Ultimately, this longtime BSD developer did a meeting for us and became one of our biggest advocates.

## ***Starting with What We Didn't Want***

In retrospect, we did dwell in a state of utter confusion. But we were clear on what we didn't want.

A number of us had been around other technical associations and user groups, which all seemed to fall into one of two categories.

First, there were professional associations that lacked any substantial content and seemed more like resume line-items. The respective sector's vendors—such as information security—dominated the presentations and materials, and the passive attendees touched keyboards only for hiring and firing purposes.

Second, technical user groups earned well-

deserved reputations as gathering places for the socially awkward, where one or more cliques congregated to exchange the odder ends of technical culture. And like the professional associations, presentations could still be drenched in sales pitches.

We wanted neither scenario. We sought a loose grouping that focused on the BSDs in the production environment.

### ***All about the Asterisk***

Another critical element was to be agnostic in relation to BSD projects. While many people in NYC\*BUG might strongly favor one BSD over the others, we remain officially neutral. This is not some naive notion. We witness the same occasional mailing list flame wars or nasty interview comments that the rest of the world sees. But fundamentally, we also think the BSDs have more in common than not.

The license, in whichever simple form, remains the clearest overreaching principle. Some may view it as developer- or corporate- or freedom-friendly, but we all are comfortable in having a license that doesn't require a few years of law school to comprehend. If one of the projects imploded tomorrow, we think that the remaining projects would likely provide reasonable new homes for the particular developers and users.

Contrasting the BSD community to other open-source communities also further binds the BSDs together. While criticisms might flow free between the projects within the confines of NYC\*BUG, it is all in the spirit of open discussion and without the nastiness characteristic of online flame wars.

Over time, it also became apparent that regular physical meetings undercut the environment for trolling or even making unjustified comments about other BSD projects. The online world allows the worm to imagine itself as a king. But when that worm has to look other worms in the eye, delusions of royalty tend to dissipate.

One meeting in the past year provided an overview of a particular BSD project. There were no other users from that project in the room, and we were curious to see how the discussion would evolve.

The outcome was worth a thousand words. One longtime developer and Unix veteran criticized the project for lack of a focus, but it was done in a manner that more likely pointed to the problems in that person's own BSD project. Another longtime developer humbly spoke about the problems faced by their third BSD project, presenting instruction to the speaker as an open mea culpa. The modesty evoked by face-to-face meetings remains

one of NYC\*BUG's strengths. All open-source projects are glass houses while online interactions tend to obscure this reality.

### ***Sales People Not Welcome***

Another principle that quickly developed was avoiding salespeople, and even implicitly discouraging their attendance. NYC\*BUG meetings and mailing lists were no place to preach about some "disruptive" new product, nor a pool for technical recruiters to fish for resumes. Yes, NYC\*BUG meetings are free and open to all, but it's the BSDs in production that provide the overarching principle.

There was one particular early trial for this. We asked Apple corporate people in New York for an engineer to speak about Darwin, the BSDs and open source. We remain appreciative and well-aware of the contributions Apple has made to the BSDs, but this particular salesperson insisted such a topic would be nothing more than a 15-minute presentation and implored us to allow the circulation of sales materials. We stuck to our approach, and frankly threatened to end the meeting if the technical presentation became a sales pitch.

The meeting was technical with standing-room-only for over 50 people, with almost no one leaving the room during a two-hour presentation. This confirmed our view. Particularly with so many of us being veterans of the Dot com era, no one was going to voluntarily listen to a sales presentation without an open bar, some expensive giveaways, and maybe a decent steak. We were not looking to occupy a free night each month in our schedules. Rather, we wanted to create an environment where the attendees are active participants and where meeting topics reflected what people were actually doing. It was, and remains, the yardstick by which we measure NYC\*BUG's successes and failures.

### ***The Right Relation to Sponsors***

Within the context of New York City in 2004, it should be understood that it was, and remains, a Linux town. Until the 2008 crash, the bellwether industry in New York City was finance, often as Linux shops migrated from Solaris. There were a few significant BSD shops in NYC and we set out to engage those shops.

Larger trends matter. The adoption of Linux in the finance industry mattered. But years later, the advent of systemd in the Linux distributions created a new audience for us. Apple's loud support of Unix with the launch of OS X gave us a larger hearing. Noting those larger issues goes a long way in reaching out to broader audiences beyond

# Starting a BSD User Group

the usual suspects.

Probably the most fruitful relation built with a BSD shop has been with New York Internet. They had moved from Solaris to FreeBSD years earlier, and were consistently rated one of the best public data centers. While we didn't have the quantity/size of the Linux audience, they were quick to offer us support due to our quality. In one owner's words (paraphrased), "You seem like good people to be around." Since then, NYI has provided us with a full cabinet hosting dozens of projects and has been a consistent sponsor. They have also contributed greatly to the community beyond NYC. FreeBSD's U.S. East Coast infrastructure resides in their New Jersey facility, and both the BSD community and NYI are better off with this relationship. Projects such as the BSD Certification Group, pfSense, to BSD.lv and mandoc, not to mention the many hosted mirrors, all have benefited from

---

**"We are asked frequently for advice on starting a user group. The first piece of advice is don't attempt to replicate us."**

---

the NYI relationship.

It's important to connect the notion of "good people to be around" with our attitude to sales presentations, etc. Potential sponsors might want to have free reign on mailing lists and presentations, and even conference attendee lists. But a user group that allows such access probably doesn't attract the types of people that sponsors want to talk to in the first place. Yes, we might make it difficult, but it's well-worth the efforts if our parameters are respected.

## ***Coordinating without Assets***

The sponsor question raises an issue of funds and assets. We have conducted five conferences and managed to make a profit on the last four. We maintain the cabinet at NYI. But ultimately, we have no assets, nor do we pursue them. Any profits gained from the conferences are donated back to the BSD projects.

Maintaining a nonprofit entity, or even just a NYC\*BUG bank account, demands a higher level of responsibility and accountability than we want. And having such entities means brewing potential conflicts. Being asset-poor means having nothing to fight over. It's safer to be a starfish than a spider, in reference to that pop sociology book. Fork us, copy us, but we are too loose to break. NYC\*BUG is more like viscous mud than a sand castle. Cut up a spider into two parts and it dies. Cut up a

starfish into two parts, and you have two starfish.

We have maintained a certain informality from the beginning. We strive to allow a fluid membership, which means we can't conduct any meaningful voting. Who is qualified to vote? Mailing list members? What about those who attend one meeting a year? Or those who rarely miss a meeting?

Once the formalisms required by elections are implemented, too many other procedural questions arise. While we explored creating a nonprofit early in our history, we quickly realized that the stakes would be higher, and that being asset poor and as informal as possible was a better route.

Six of us walked into a meeting with a technology-oriented NYC agency that was excited to migrate us into a nonprofit structure. Five of us were on board. But five of us walked out of the meeting against creating a nonprofit, and content to continue without any formal structure.

Outside of a few responsibilities performed by the NYC\*BUG admin group, no one is really compelled to do anything. One might attend a meeting when the topic is interesting. Or engage in a mailing list thread when it's relevant to a job or interest. But the regular appearance and reappearance of people became natural, and maintaining this low barrier to entry (and exit) shapes the atmosphere for the better.

But then how can the admin group be a useful body? Admin's purposes are simple. It is not a privilege to be on admin. The central function is to resolve the organizational questions inappropriate for the talk@ mailing list. Discussions might revolve around a colocation issue in the cabinet, or determining upcoming meeting topics. Admin membership evolved from a few of those taking on the roles of coordinating NYC\*BUG activities, yet is unelected and there are no formal terms.

For most BUGs, an admin group isn't just unnecessary, it's really a dangerous barrier. If five people are meeting regularly, having two of them decide organizational questions on a separate mailing list is useless. And it will only discourage the other three people from taking an active role. Besides eternally seeking to infuse technical solutions to organizational problems, technical people also tend to overbuild organizational infrastructure as if they are scaling a startup.

Once the admin ball was rolling, we attempted to make it selective in choosing members, yet easy for members to leave admin without any discomfort. It hasn't always been successful, but overall, admin continues to function and provide a decent amount of direction. On occasion, someone from admin will note that some particular individual would be a good addition. We determine their

inclusion by consensus, since the chemistry of admin is critical.

One noteworthy incident illustrates admin's role at its best.

Like the broader BSD community, NYC\*BUG faces an ugly dearth of diversity, particularly in terms of gender. We are quite conscious of it, and it is a regular discussion. Part of our approach to rectify this is to make it clear that sexism and patronizing attitudes toward women are unacceptable.

In one case, a NYC\*BUG talk@ poster used a disgusting and sexist signature in his emails. As this individual had already been a problem on other levels, many people around NYC\*BUG had already been sending his posts to /dev/null. But when his signatures appeared, several people not on admin emailed demanding action.

He was promptly removed from the mailing list and the reason was communicated. But it was indicative that other NYC\*BUG participants, not on admin, had quickly reacted and believed it was their responsibility to address the situation as well.

### **Warning: Don't Try This at Home**

We are asked frequently for advice on starting a user group. The first piece of advice is don't attempt to replicate us. We are an unusual example. We are in a dynamic, global city filled with technical people and resources. Our meetings frequently include some of the "who's who" of the Unix world. If you're located in a small college town, or in a rural, dispersed area, attempting to photocopy NYC\*BUG will lead to endless frustrations.

The overriding lesson is that we started appreciating the context in which we operate. We knew the NYC technical scene. We knew the user group scene. We knew what we didn't want to do. And most importantly, we started modestly with goals we could accomplish reflecting our situation and the resources at hand.

Most of the time when someone wants to initiate a BSD user group, there is one individual driving the effort with the time, energy, and ideas. And that leads to the first problem. There shouldn't be one individual who owns the project. There needs to be at least a handful of those who share a common vision and who can take responsibility for getting the BUG operational, and maintaining it into the future.

On that note, the first step might be initiating a mailing list and publicizing it. Having a launch meeting usually translates into one or two people being viewed as the organizers, and the rest of the attendees passively voting with their involvement, or lack of involvement. A mailing list is a good vehicle for gathering people to take on active involvement

from the beginning. Finding common interests and topics is a critical first step.

If you're a handful of people in an hour's radius of, say, Rzeszow in Poland, meeting monthly is difficult. Maybe a quarterly day-long event makes more sense. And why not have everyone engaged in organizing the event? In that case, the mailing list is likely the ideal platform for the group's activities.

Our meetings can range from a dozen to dozens of attendees. For others, four or five is a worthwhile accomplishment. And a meeting on, say, Apache, doesn't have to be presented by a relevant BSD port maintainer. This is a normal mistake. If people in and around the BUG are doing something with the BSDs, there are topics and presenters available. Relying on "names" and topic authorities means the well of speakers dries up quickly. Having actual participants present means keeping the barrier to entry low for everyone. There are a number of speakers at BSDCons whose first presentations were at NYC\*BUG meetings. We created the environment that was comfortable and interactive and which ultimately benefits the BSD community beyond NYC.

### **Come On In. The Water's Great!**

Another function that BUGs can play is providing a conveyor belt for individuals to be involved in BSD projects. There is a tendency to view entry into the BSD projects as some virtual meritocracy in which the good rises above the crud, and tomorrow we'll have our next layer of core developers.

Reality is quite different. BUGs can be the place in which meeting developers and other users, face-to-face, can introduce more isolated users into the BSD culture. One would imagine that a lot of eager potential developers find that there's a very valid reason why bash isn't the default shell, or that testing ports on even esoteric architectures does matter.

While NYC\*BUG could never take credit for creating developers, a good number of people have gone from being isolated BSD end-users into developers in the environment fostered. Contributing ports, for instance, is a relatively easy path into contributing to the BSDs, and we've attempted to provide introductions that are easy-to-grasp for more users. From there, the slippery slope of spending those few free waking hours quickly evaporates into hacking makefiles and checking dependencies.

### **Fun, Fun, Fun**

Our golden rule remains a phrase from an early USENIX participant: are you having fun?

It is difficult to be motivated to deal with the hassles, the meetings with no-show speakers, etc., if you aren't having fun. Make it fun. If meetings become a chore, maybe make them quarterly. Or



## Starting a BSD User Group

just keep the mailing list. Or an IRC channel. Ultimately user groups do not turn into wildly successful IPOs and result in the creation of brilliant new technologies. They are really just social organizations with a purpose.

### **BUGs Matter**

BUGs can provide a useful framework for the community, as an alternative to being grounded in the corporate world or with isolated developers. BUGs can better reflect what actual users are doing, with their needs, desires, and frustrations. Unlike corporate donations, BUGs can provide no-strings-attached contributions. Ultimately, a broader layer of people can support the projects, which in turn also gives a deeper sense of ownership of the projects.

BUGs can provide a useful and grounded picture of production BSD usage in a way surveys can never manage. We spend a lot of time determining meeting topics and speakers, and sometimes we find our choices fall flat. Other times, we find that being creative on meeting topics piques interest no one realized existed. Nothing is more flattering than watching other user groups in NYC, and beyond, replicate our meeting topics and approaches.

NYC still can't really be considered a "BSD City," but the flag we planted certainly fosters that notion.

The BSD community is not traditionally known for its advocacy. But if we had a world with a few dozen BUGs that could maintain mirrors for each BSD project, donate funds and hardware regularly, connect with BSD-using firms in their area, and so on, the BSD community as a whole could be significantly stronger.

NYC\*BUG, for instance, created a place where the BSD-curious can dip their toes into the water, and maybe explore how the BSDs could work for them. Such an approach can also work far beyond the confines of New York City. ●

---

**GEORGE ROSAMOND** is a founding member of the New York City BSD User Group. By trade a systems administrator, George maintains a small colocation firm for technical clients, in addition to contributing to an array of BSD projects large and small, with a particular focus on privacy-enhancing technologies. For almost two years, his central focus has been the Tor BSD Diversity Project (<https://torbsd.github.io/>), which looks to extend the usage of the BSDs into the Tor public anonymity network (<https://www.torproject.org/>).

## ZFS experts make their servers **ZING!**

Now you can too. Get a copy of.....

**Choose ebook, print or combo. You'll learn:**

- Use boot environments to make the riskiest sysadmin tasks boring.
- Delegate filesystem privileges to users.
- Containerize ZFS datasets with jails.
- Quickly and efficiently replicate data between machines
- Split layers off of mirrors.
- Optimize ZFS block storage.
- Handle large storage arrays.
- Select caching strategies to improve performance.
- Manage next-generation storage hardware.
- Identify and remove bottlenecks.
- Build screaming fast database storage.
- Dive deep into pools, metaslabs, and more!

Link to: **<http://zfsbook.com>**

WHETHER YOU MANAGE A SINGLE SMALL SERVER OR INTERNATIONAL DATACENTERS, SIMPLIFY YOUR STORAGE WITH **FREEBSD MASTERY: ADVANCED ZFS**. GET IT TODAY!

