

The journal of Apple technology.

Volume Number: #

Issue Number: #

Column Tag: Community

USENIX != WWDC

An impressionistic report on two very different technical conferences

by Rich Morin

Background

Apple (www.apple.com) and USENIX (www.usenix.org) both run excellent technical conferences, but the events are extremely different in approach, scope, size, and general flavor. So, although I can easily recommend their conferences to MacTech readers, the specifics of the recommendations will vary significantly.

Both organizations are both quite experienced at running conferences, having done so for more than two decades. Apple's WWDC (Worldwide Developers Conference) is much larger than USENIX's Annual Conference, but USENIX runs several events each year. So, both organizations do a good job of crafting schedules, picking talks, etc.

USENIX (co-)sponsors more than a dozen events (www.usenix.org/events) each year, covering a wide range of topics. This year's offerings, for example, include gatherings on autonomic computing, distributed systems, electronic voting, mobile computing, virtual execution environments, and several subtopics of system administration (e.g., computer security, large installations).

The Annual Conference is thus a smorgasbord of refereed and invited papers on hot topics in software design. Not accidentally, it is also a gathering place for researchers who want to trade ideas, argue about approaches, etc. A few brief tutorials may slip into the main track, but most are sequestered into intensive "tutorial tracks".

WWDC, in contrast, is Apple's annual opportunity to bring interested techies up to speed on new, Apple-specific technologies. So, WWDC sessions tend to be a mixture of presentations and tutorials. Although there may be an occasional nod in the direction of Unix and/or Open Source, the focus is primarily on Apple products: iPhone, Mac OS X, Safari, etc.

USENIX Impressions

The first USENIX conference I attended was in the early 80's. Many vendors were rushing to create Unix systems. I recall being quite intimidated, as a newbie, by the presence of the operating system's developers and early adopters. However, this was more than balanced by the excitement of hearing them present (and argue about) various design possibilities.

It was also amusing, on occasion, to watch the interactions of the business and technical folks. In one fabled interchange, a "suit" asked a pony-tailed man if he had been using Unix for a while. In a masterpiece of understatement, Dennis Ritchie answered "yes".

This Year's Sessions

Today's Annual Conferences are far less focused on kernel issues than in earlier years, but they are still a great place to hear about new and upcoming technologies. And, because most Open Source projects are developed on Unix-like operating systems, many of these projects get discussed, as well.

Even if you can't get to a specific conference, you can still take advantage of its presentations. Recordings and proceedings are often available from the USENIX web site. Here, by way of example, are some highlights from this year's conference...

Guido van Rossum (Google) gave an update on the "Python Programming Language". Steven C. Johnson (MathWorks) led a discussion about the disconnect between current programming languages and parallel (e.g., multi-core) computing environments. Tom Christiansen's "Advanced Perl" session contained several useful tricks about using regular expressions.

This year featured several sessions on large-scale computing, with presentations from Amazon, Google, Linden Lab, LiveJournal, SiCortex, Tellme, and others. There were also a large number of refereed papers (generally quite specialized) from researchers.

I particularly enjoyed Luis van Ahn's talk on "Human Computation". As one of the folks behind the CAPTCHA puzzles, he felt guilty about the amount of cognitive effort that is being wasted in making humans recognize distorted text. So, he has been devising ways to capture some of this effort (to assist OCR efforts) and writing games (e.g., www.espsgame.org) that capture useful work from the players.

One Laptop per Child

My favorite session, however, was "Crossing the Digital Divide: The Latest Efforts from One Laptop per Child". Mary Lou Jepson, a key hardware developer on the project, described some of the tricks she and others have used to produce a truly extraordinary laptop.

The OLPC laptop (laptop.org) has many features that I'd like to see more generally adopted. First, it is amazingly robust and safe. It can be used in the rain or at high temperatures (e.g., 50C!), dropped several feet onto concrete without damage, and fed by a wide range of power sources. If something breaks, it can be repaired in the field (often by children). It contains no mercury and the battery electrolyte burns at 100C, rather than the 1000C of typical laptop fires.

It also runs on a measly two (!) watts, provides opportunistic "mesh" networking, and has a screen that can be read in direct sunlight. Finally, of course, it costs less than \$200. Where do I sign up?

Where's Apple?

Apple is arguably the world's largest Unix vendor. However, aside from the prevalence of Apple laptops, the company was completely invisible at this event. They provided no speakers, held no BOFs (Birds Of a Feather meetings), and were not listed as sponsors. [Ed. note: Apple has sent speakers to Usenix in the past.]

I can understand why Apple might not wish to discuss unannounced or proprietary technologies, but why not present some of their shipping, Open Source innovations? Apple has introduced a substantial number of these, in areas ranging from device driver design through dynamic configuration to power management and web standards. Someone is missing a bet...

WWDC Impressions

Due to the restrictions of Apple's WWDC Non-Disclosure Agreement (NDA), I'm unable to talk about unannounced information, let alone the specifics of the presentations. However, I can (and will) talk about the presentation style, amenities, etc. I hope this gives you a useful idea of the "flavor" of the event. Speaking of which...

Noxious Noshes

In previous years, one of the benefits of attending WWDC was the presence of plentiful, high-quality food. Hot lunches were provided, generally with a selection of entrees. Tasty snacks were available between sessions and free "espresso bars" were on hand to provide hot drinks. Unfortunately, the food service this year was considerably degraded from that of previous years.

I tried one "mocha" at the espresso bar; it was made up from a packet of pre-mixed powder, much of which was still present in the drink. It was so unpalatable that I discarded it after two sips. The lunches were better, but over-cooked chicken and refrigerated sandwiches are not to my taste; I opted to pay for food at a nearby restaurant. I realize that complaining about the food may sound petty, but given the substantial cost of the event, Apple really should do better.

Solid Sessions

Apple's annual WWDC conference is a mix of overview and tutorial talks, generally presented by Apple employees (primarily engineers, with some managers and "evangelists"). As you might expect, the production values are solid, if sometimes a bit monotonous. All of the presenters use Keynote, most of the demonstrations work, and presentations follow a rather predictable pattern.

Because Apple publishes the WWDC session titles (at developer.apple.com/wwdc), I can tell you which topics I particularly enjoyed. In many cases, a bit of Googling will bring you to sites that give more details than I can (safely) disclose here.

The "State of the Union" talks are always worthwhile. Even if I'm not planning to work with a particular technology, attending its SotU session will give me a good overview of where it is headed. SotU sessions also serve as useful road maps to more detailed sessions.

I'm not a big fan of Objective-C; as a scripter, I think it makes the programmer write way too much code. So, I was happy to see that Apple is moving forward with its "new and improved" version, "Objective-C 2.0". Careful borrowing from scripting languages could make ObjC code easier to write and less subject to programming oversights.

Xcode 3.0 also looks very interesting as a scripting environment. I've been using TextMate and Terminal to develop Rails applications. Xcode's support for Ruby and the Xcode Organizer will provide an interesting, GUI-based alternative. And, of course, I'm delighted to see official support for Cocoa programming in Python and Ruby.

I've been a fan of DTrace (www.sun.com/bigadmin/content/dtrace) since Sun first announced it. Having a scriptable way to monitor kernel, library, and application activities is immensely useful. So, having Apple add DTrace to Leopard is a big win for Mac OS X developers.

However, Apple didn't stop there. Borrowing extensively from GarageBand, they produced XRay, a really slick GUI-based front end to DTrace. XRay provides dozens of "instruments" that can be attached (via DTrace) to various parts of the system. It then allows the developer to view the results, drill down for more information, etc.

I dislike the fact that widgets don't live in the same "space" as normal Cocoa apps, because this keeps me from cutting and pasting, etc. However, many folks find widgets to be extremely useful. Apple's new development tools (DashCode, WebClip) will allow many more (and better) widgets to be created.

For advance information on Xcode, XRay, and DashCode, see Apple's "teaser" page

(www.apple.com/macosx/leopard/developer) and/or fire up your favorite search engine...

Pernicious Privacy

Apple's well-known penchant for privacy makes the sessions less useful than they might otherwise be. Because cameras and recorders are prohibited and no printed materials are provided, attendees are left trying to take notes (or simply remember) the "fire hose" of material that each presentation provides.

And, because Apple doesn't provide any NDA-safe forums, it is difficult for attendees to follow up on presented technologies after the conference. More generally, non-disclosed ADC members have no legal way to trade notes on bugs, features, or programming techniques. Apple has known about this problem for years; it's really time they fixed it...

Non-Apple material is very sparse at WWDC, consisting of end-of-talk Q&A periods and a small number of feedback sessions. There are a few informal lunchtime sessions and some evening events at the nearby Apple Store, but there are no BOFs. Nor, sadly, were there any booths this year for third-party vendors, Open Source projects, etc. These were an interesting part of earlier WWDCs; I'd like to see them return.

One conspicuous (and delightful) exception this year was a scientific "poster session" that took place in a conference hallway. Several dozen presenters stood next to large (roughly 3' x 5') posters that described their work. Topics ranged from ArchImage (architecture) to WeatherScope (meteorology).

Conclusion

In summary, WWDC is a large, narrowly-focused, unidirectional multicast from Apple to its developers; the back-channel is informal and low bandwidth, at best. That said, WWDC is by far the best way for prospective Mac developers to get started and for experienced Mac developers to learn about new technologies and fill in gaps in their knowledge.

If you can't go in person, watching the recorded versions of the talks is reasonably effective. Unfortunately, Apple does not make these generally available. Given that only 0.5% of the ADC membership got to this year's conference, I think that Apple is missing another bet...

Rich Morin has been using Unix on Macs (e.g., A/UX) for two decades. As you might expect, he's very happy with Mac OS X and the way it has been accepted by the Apple community. Rich does contract technical editing and writing, programming, and web development for a living (and, between contracts, for fun). You can reach him at rdm@cfcl.com.