

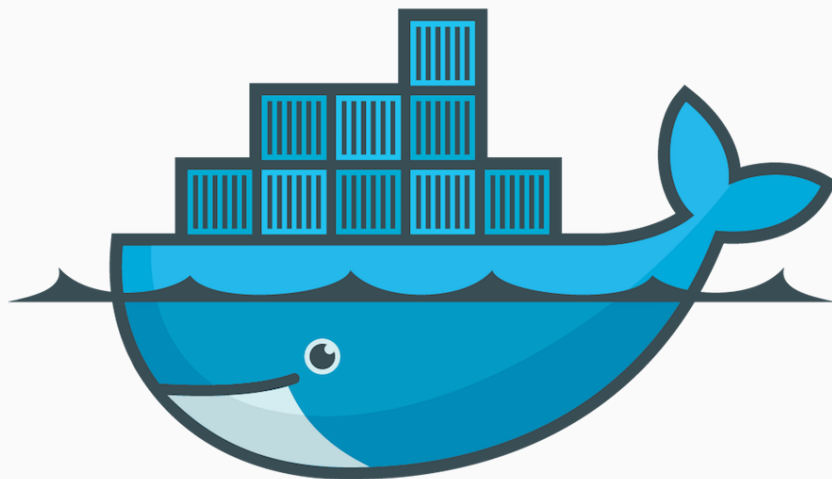
A periodic newsletter with links to blogs and industry news for subscribers to the Architecting IT blog.

[View this email in your browser](#)

This is newsletter #24, up to 30 June 2016.



# architectingIT



# docker

## Thoughts on DockerCon 2016

This year I thought I'd try something different when attending conferences and booked to attend DockerCon in Seattle. I chose to pay for it personally (although I probably could have been given a media badge), mainly because I wanted to experience the event as an actual end user. Aside from the lack of workspace and power that I normally expect (with the privilege of blogger lounges), I have to say that I think the event was a great success.

There's a palpable sense of optimism from the Docker team and attendees, in the way they are taking container technology forward that obviously springs from youth and the desire to want to change the way applications are traditionally deployed. That said, there are still gaps in the way in which Docker can support the Enterprise, not least of which is the ability to manage long term persistent data.

As a case in point, Casey Bisson from Joyent presented on “Persistent Data Patterns: Dockerize Your Database for Ultimate Power and Flexibility” on the Ecosystem Track. His recommendation for MySQL was to ignore the use of Docker volumes, keep data in multiple replica containers (presumably on different hosts) and use a Joyent plugin to archive updates into an object store. This approach clearly isn’t either scalable or practical from a performance perspective.

In [Brian Goff](#)’s presentation (available [here](#) to watch [here](#)) on storage, he talks about persistent storage purely from the point of Docker plugins, without regard for how security, mobility or data protection will be achieved. In fact, taking a question from the floor on backup, his response was simply “storage is hard”. There’s clearly some way to go in resolving this part of the infrastructure and unfortunately the ability to persist data reliably is a tough job in a world where applications have increasing mobility.

Looking at some of the other features presented over the two day conference, we see a focus on expanding the developer ecosystem to make applications both easy to write and deploy. New Docker for Windows and Docker for Mac allow applications to be built and run locally, crucially without having to install a vast array of tools to the local machine. Instead, these features run inside local containers that are in effect a self-contained development environment. This may not seem powerful at first glance, but imagine having to maintain many versions of Python, Java and other languages on the same development machine. This benefit will be more readily appreciated as we move to Windows container-based application development that have significant dependencies on .NET and DLL images.

Speaking of Windows, Microsoft demonstrated Windows running containers on Windows Server 2016 TP5, the latest preview release of the next version of Windows for servers. At this stage the container deployment process is nowhere near as smooth as that for Linux. There are at least 20 processes running in each container and the registry has to be replicated for each new container spawned. The Nano image for Windows is 4GB with a Server Core image at 9GB, making it pretty heavyweight for a standard download. That said, Microsoft seems to be investing lots of money to bring containers on quickly, and thats a good thing. You can watch the Microsoft presentation from DockerCon [here](#).

I’d recommend watching the two keynotes from the start of each day of DockerCon. Day one is more about the developer, day two has more on the enterprise deployment of containers into the IT ecosystem. There are some

good presentations on how to use the new Docker Swarm in release 1.12 and on fixing security bugs and using Swarm to do a rolling upgrade of an application. This more than anything speaks to the power of Docker and containers and that's the ability to ship code in the most lightweight and efficient way possible. This ability for me at least, is much more powerful than the ability to cut down the size of VMs compared to containers. It's all about agility.

- [DockerCon Keynote Day 1](#)
- [DockerCon Keynote Day 2](#)

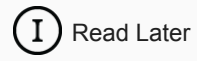
What do you think?

Chris Evans

---

### New Architecting IT Blog Posts - May/June 2016

- [Are DataCore's SPC Benchmarks Unfair?](#) (22 June 2016)
  - [Docker for Mac and Windows - Changing The Developer Paradigm](#) (21 June 2016)
  - [Docker & Storage - Solving the Problem of Data Persistence](#) (19 June 2016)
  - [SolidFire Introduces New "FlashForward" Purchasing Model](#) (7 June 2016)
  - [Torus - Because We Need Another Distributed Storage Software Solution](#) (5 June 2016)
  - [Object Storage: Validating S3 Compatibility](#) (31 May 2016)
  - [Storage Field Day 10 Preview: Exablox](#) (24 May 2016)
  - [Storage Field Day 10 Preview: Hedvig](#) (23 May 2016)
  - [Storage Field Day 10 Preview: Primary Data](#) (24 May 2016)
  - [Storage Field Day 10 Preview: Nimble Storage](#) (23 May 2016)
  - [Storage Field Day 10 Preview: Datera](#) (22 May 2016)
  - [Storage Field Day 10 Preview: Tintri](#) (20 May 2016)
  - [Storage Field Day 10 Preview: Pure Storage](#) (19 May 2016)
  - [Storage Field Day 10 Preview: Cloudian](#) (16 May 2016)
  - [Storage Field Day 10 Preview: Kaminario](#) (17 May 2016)
  - [Long Term Evaluation: Scale Computing HC3](#) (9 May 2016)
  - [One \(Storage\) Protocol to Rule Them All?](#) (13 May 2016)
  - [EMC Unity or VNX3? What's In a Name?](#) (4 May 2016)
- 
-



---

*Copyright © 2016 Brookend Ltd, All rights reserved.*

Want to change how you receive these emails?  
You can [update your preferences](#) or [unsubscribe from this list](#)