

A large, abstract graphic of a cloud shape composed of glowing, multi-colored lines (blue, purple, cyan) and a network of nodes, set against a dark blue background with a starry pattern. The cloud has a downward-pointing arrow shape integrated into its base.

Distribution of Container Images

From tiny deployments to massive analysis on the grid

Enrico Bocchi
CERN IT, Storage Group

“Build, Ship, Run, Any App Anywhere”



Build

Develop an app using Docker containers with any language and any toolchain.



Ship

Ship the “Dockerized” app and dependencies anywhere - to QA, teammates, or the cloud - without breaking anything.



Run

Scale to 1000s of nodes, move between data centers and clouds, update with zero downtime and more.

© Docker Inc.

“Build, Ship, Run, Any App Anywhere”



Build

Develop an app using Docker containers with any language and any toolchain.



Ship

Ship the “Dockerized” app and dependencies anywhere - to QA, teammates, or the cloud - without breaking anything.



Run

Scale to 1000s of nodes, move between data centers and clouds, update with zero downtime and more.

© Docker Inc.

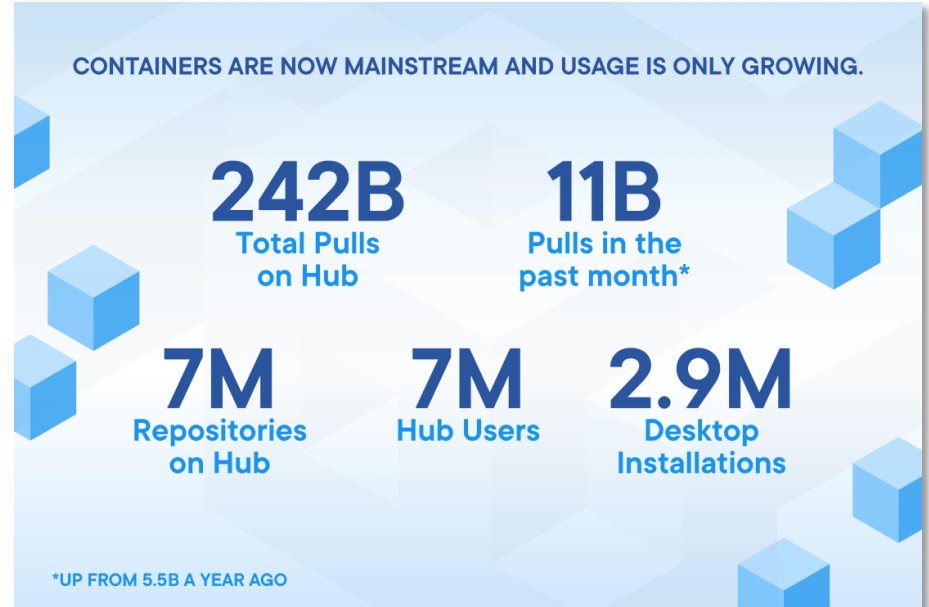
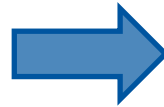
- **Container Registry:** Specialized repository to store container images
 - Distribution of images by uploading (`docker push`) and downloading (`docker pull`)
 - Public, private, self-hosted

```
docker run -d -p 8080:8080 --name myregistry registry:2.7.1
```

The Docker Hub Registry

- Most popular public registry – Docker's default

Docker Index
30 July 2020



The Free Lunch Is Over



The screenshot shows the Docker blog header with navigation links: Why Docker?, Products, Developers, Pricing, and Company. The main article title is "Scaling Docker's Business to Serve Millions More Developers: Storage". The author is Jean-Laurent de Morlhon, and the article was published on August 24, 2020.

- 150 M images
- 15 PB storage

- 4.5 PB idle images from free accounts



Starting on November 1, images [...] not pushed or pulled in the last 6 months will be removed.

Containers at CERN

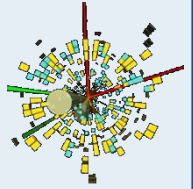
Images for Service Deployment

- Small images (< 1 GB)
- Run on few nodes
- Re-use from upstream



Images for Scientific Analysis

- Immutable unit for reproducibility
- ~10 GB per image
- Run on the Worldwide LHC Grid (potentially thousands of nodes)



Containers at CERN

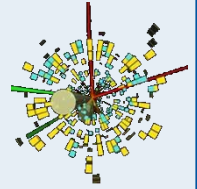
Images for Service Deployment

- Small images (< 1 GB)
- Run on few nodes
- Re-use from upstream



Images for Scientific Analysis

- Immutable unit for reproducibility
- ~10 GB per image
- Run on the Worldwide LHC Grid (potentially thousands of nodes)

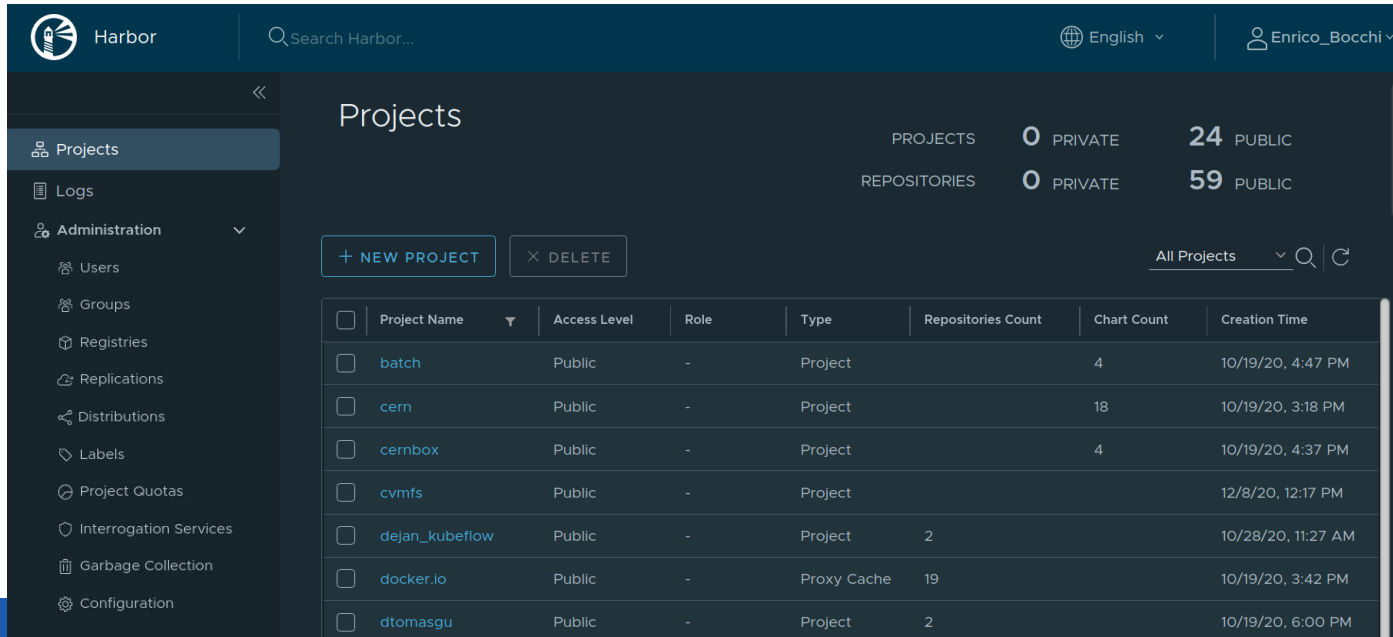


▪ Currently using **GitLab Container Registry**

- ✓ Tight integration with CI pipelines, Registry storage associated to GitLab project
- ✗ No Garbage Collection of unreferenced blobs, No support for OCI artifacts



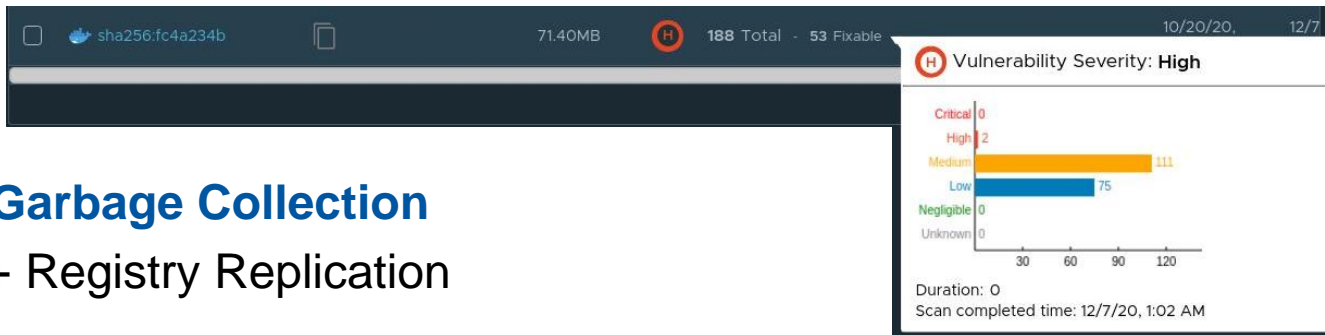
- Active Community project, CNCF Graduated
- Storage of images and Open Container Initiative artifacts (e.g., Helm charts)



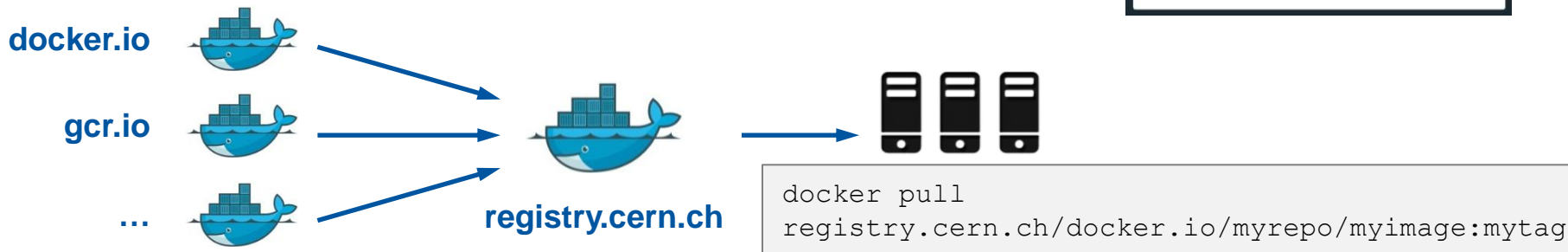
The screenshot shows the Harbor web interface. The top navigation bar includes the Harbor logo, a search bar, language selection (English), and user information (Enrico_Bocchi). The left sidebar contains navigation options: Projects, Logs, Administration (Users, Groups, Registries, Replications, Distributions, Labels, Project Quotas, Interrogation Services, Garbage Collection, Configuration), and a search bar. The main content area is titled 'Projects' and shows a summary of project counts: 24 Public Projects, 0 Private Projects, 59 Public Repositories, and 0 Private Repositories. Below the summary are buttons for '+ NEW PROJECT' and 'X DELETE'. A table lists the projects:

<input type="checkbox"/>	Project Name	Access Level	Role	Type	Repositories Count	Chart Count	Creation Time
<input type="checkbox"/>	batch	Public	-	Project		4	10/19/20, 4:47 PM
<input type="checkbox"/>	cern	Public	-	Project		18	10/19/20, 3:18 PM
<input type="checkbox"/>	cernbox	Public	-	Project		4	10/19/20, 4:37 PM
<input type="checkbox"/>	cvms	Public	-	Project			12/8/20, 12:17 PM
<input type="checkbox"/>	dejan_kubeflow	Public	-	Project	2		10/28/20, 11:27 AM
<input type="checkbox"/>	docker.io	Public	-	Proxy Cache	19		10/19/20, 3:42 PM
<input type="checkbox"/>	dtomasgu	Public	-	Project	2		10/19/20, 6:00 PM

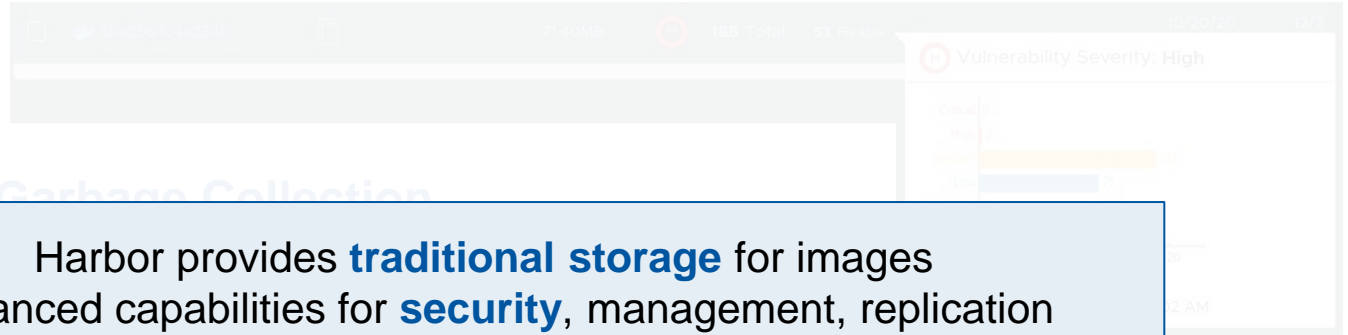
- **Artifact Signing:** Ensure trusted source for artifacts being installed
- **Vulnerability Scanning:** Based on external plugins



- Non-blocking **Garbage Collection**
- Proxy Cache + Registry Replication



- **Artifact Signing:** Ensure trusted source for artifacts being installed
- **Vulnerability Scanning:** Based on external plugins



- Non-blocking Garbage Collection
 - Prox
- Harbor provides **traditional storage** for images
Advanced capabilities for **security**, management, replication
- How to distribute multi-GB images to thousands of nodes?

docker.io

gcr.io

...

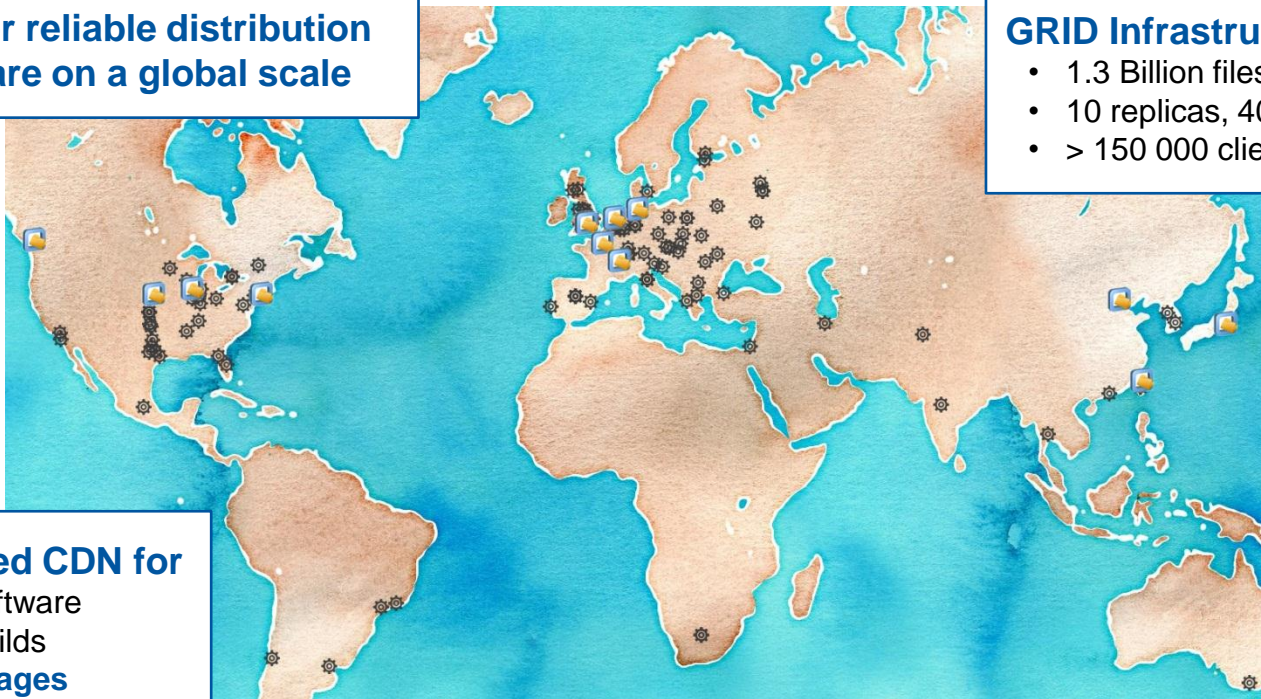


```
docker pull registry.cern.ch/docker.io/myrepo/myimage:mytag
```

**Service for reliable distribution
of software on a global scale**

GRID Infrastructure

- 1.3 Billion files
- 10 replicas, 400+ caches
- > 150 000 clients



Well-established CDN for

- Production software
- Integration Builds
- **Container Images**

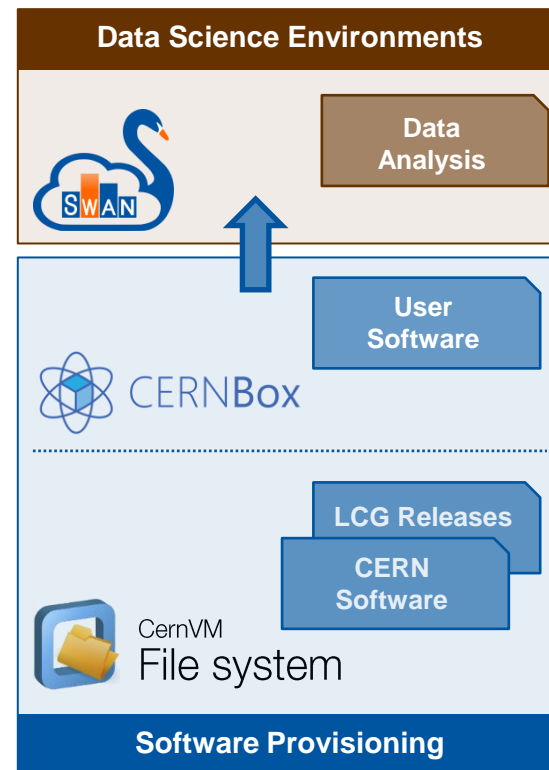
CS3 2020 Talk

CVMFS for Science Environments



- CVMFS delivers software to SWAN
 - **LCG Releases** – Frozen set of compatible packages
 - Step towards reproducibility
 - Software updates decoupled from analysis platform
- Diverse sciences and use-cases

Forget 2020: Building an integrated AARNet Cloud Ecosystem in 2021	Gavin Charles Kennedy	09:00 - 09:15
SWAN, Rucio, and Jupyter	Mario Lassnig	09:15 - 09:30
Running Parallel, Distributed ROOT Analysis with PyRDF on Public Cloud - AWS Lambda Case Study	Mr Jacek Kusnierz	
Data access and integration challenges in a distributed computational environment for medical research	Piotr Nowakowski	
JupyterLab for Earth Observation applications with HTCondor scaling and Voilà dashboarding	Mr Davide De Marchi	10:00 - 10:15



Server: Ingestion of existing images

- Extraction of layers into flat root filesystem
- Efficient file-based deduplication
- Publication into CVMFS repository

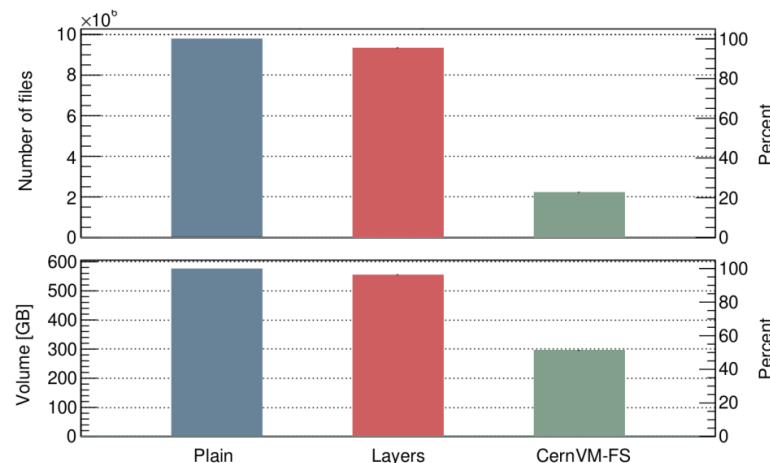
Client: Efficient pulling and caching

- On-demand fetching of required files
- No need to store the entire image locally
- Self-managed local cache

▪ unpacked.cern.ch:

First CVMFS-powered Container Hub

- 700+ container images from major experiments
- Efficiency in deduplication
- Benefits from existing CDN and CVMFS clients for large-scale execution



CVMFS for Container Images

Server: Ingestion of existing images

- Extraction of layers into flat root filesystem
- Efficient file-based deduplication
- Publication into CVMFS repository

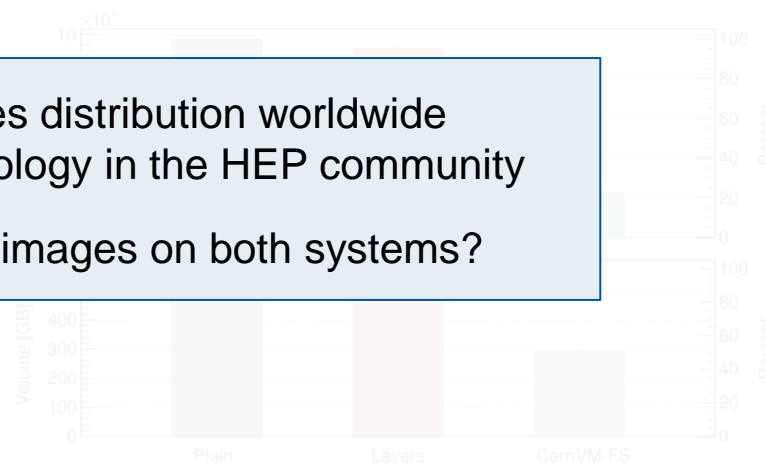
Client: Efficient pulling and caching

- On-demand fetching of required files
- No need to store the entire image locally
- Self-managed local cache

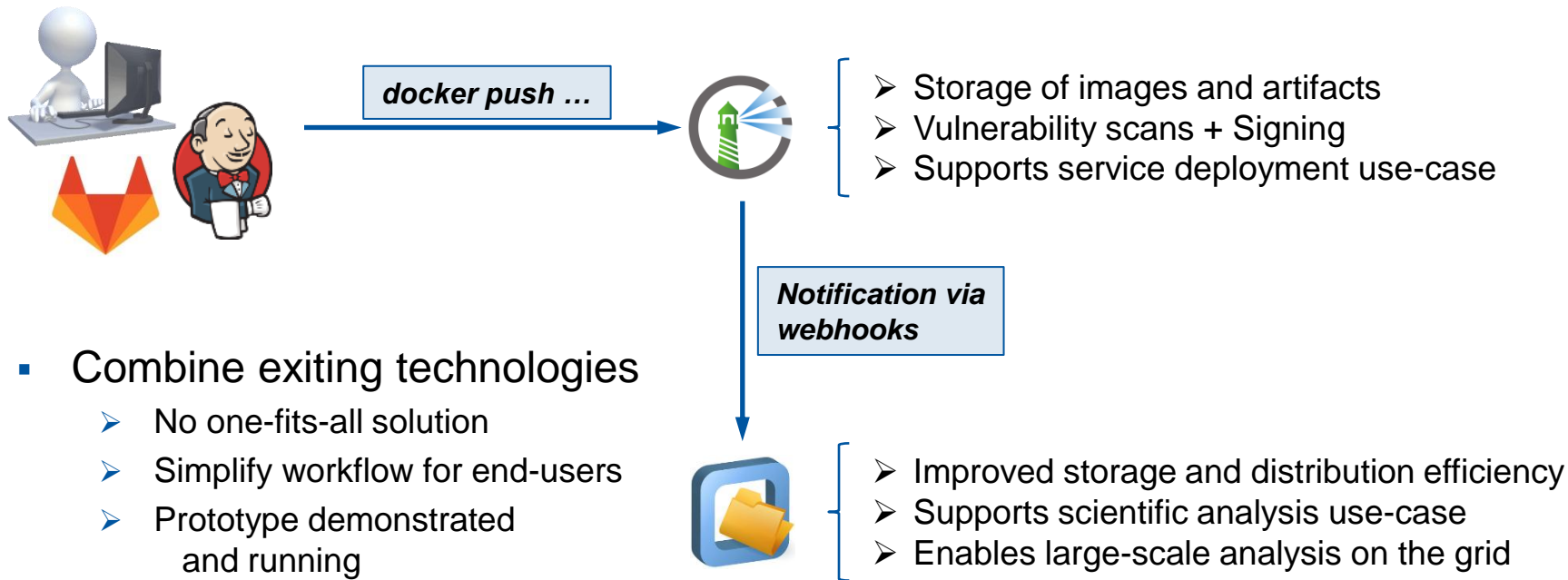
CVMFS enables **efficient** images distribution worldwide
Leverage on **widely-adopted** technology in the HEP community

- How to integrate publication of images on both systems?

Benefits from existing CDN and
CVMFS clients for large-scale execution



Streamlined Management and Publication of Images

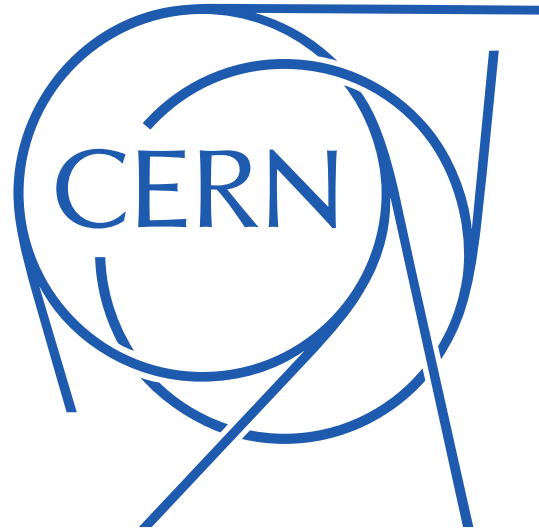


Thank you!

Questions? || Comments?

Enrico Bocchi

enrico.bocchi@cern.ch



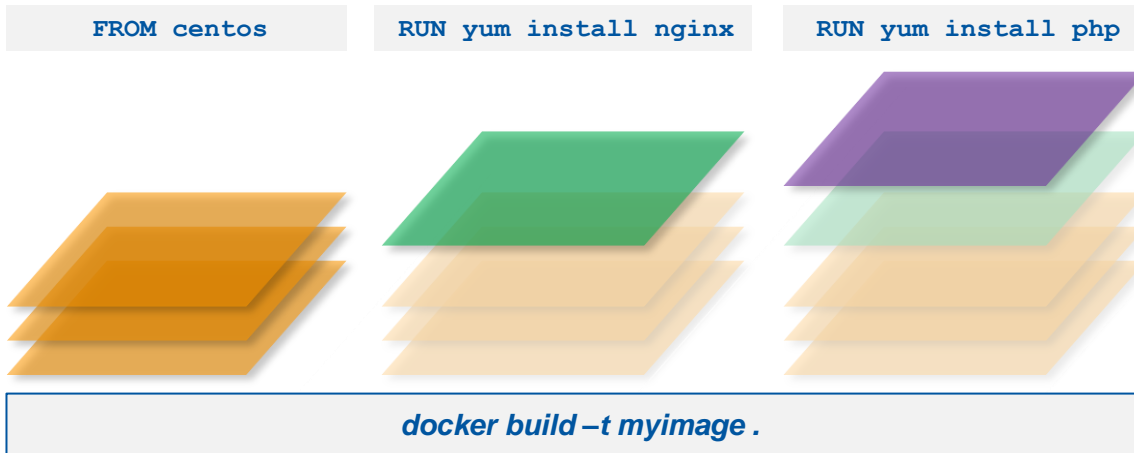
Backup

Recap on Containers Nomenclature

- **Container:** Runtime instance of an image and its execution environment
 - Provides isolation from the host environment (and from other containers)
 - Can access external resources – Network, volumes, host devices, ...

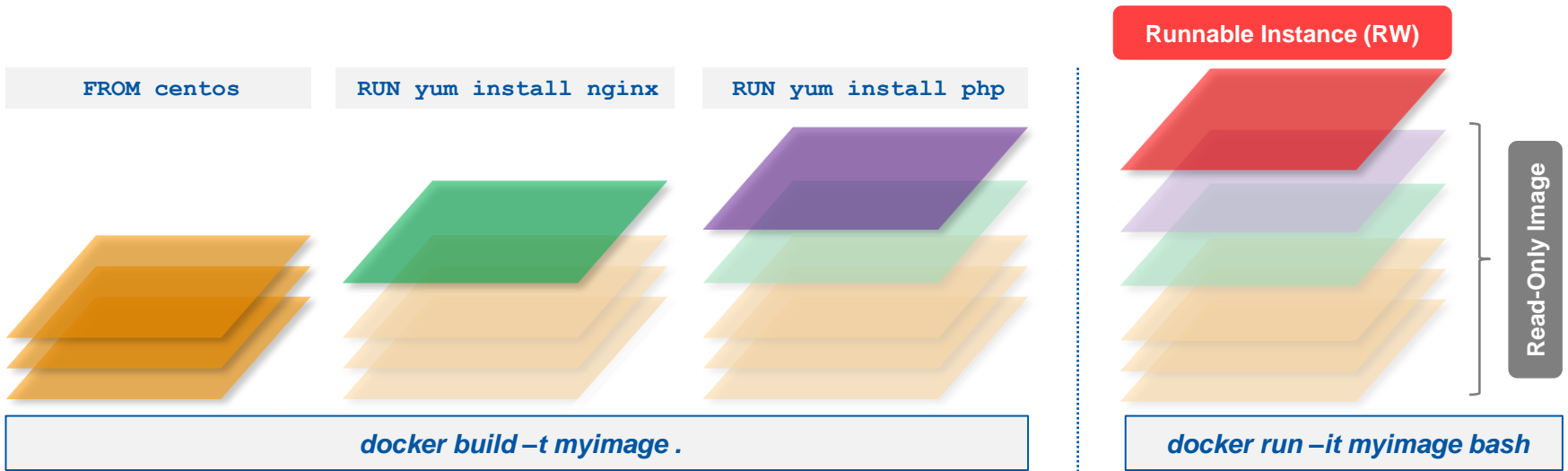
Recap on Containers Nomenclature

- **Container:** Runtime instance of an image and its execution environment
- **Image:** Self-standing portable package of software
 - Embeds all is needed to run an application (software, dependencies, settings, ...)
 - Union of several layers (tar files) stacked together



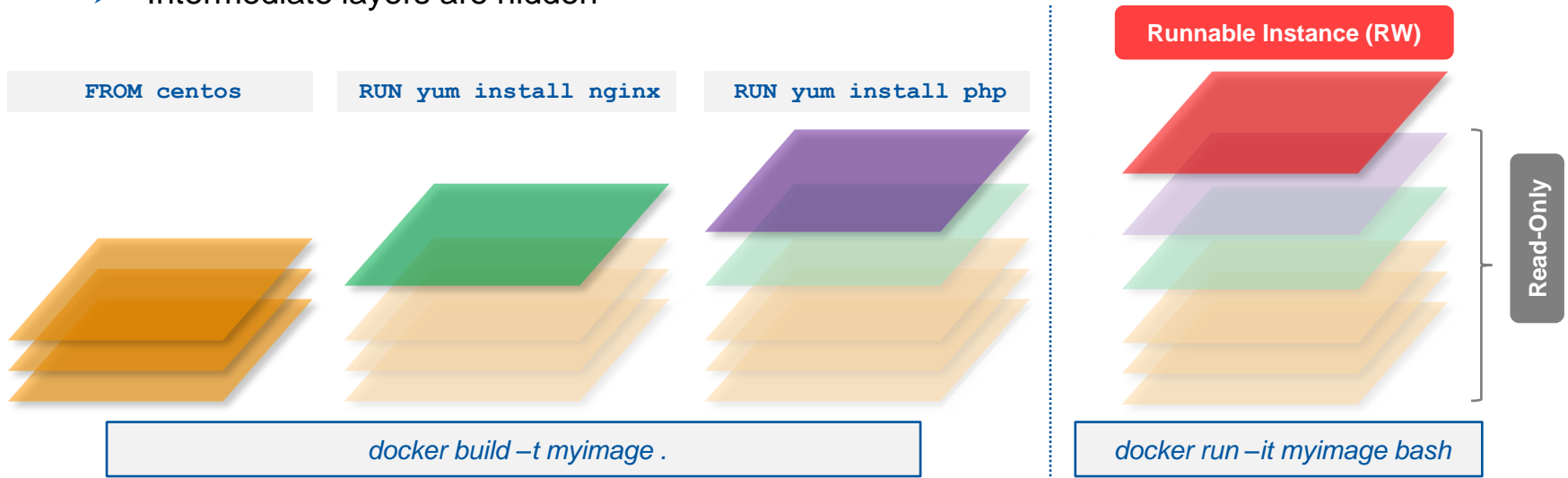
Recap on Containers Nomenclature

- **Container:** Runtime instance of an image and its execution environment
- **Image:** Self-standing portable package of software



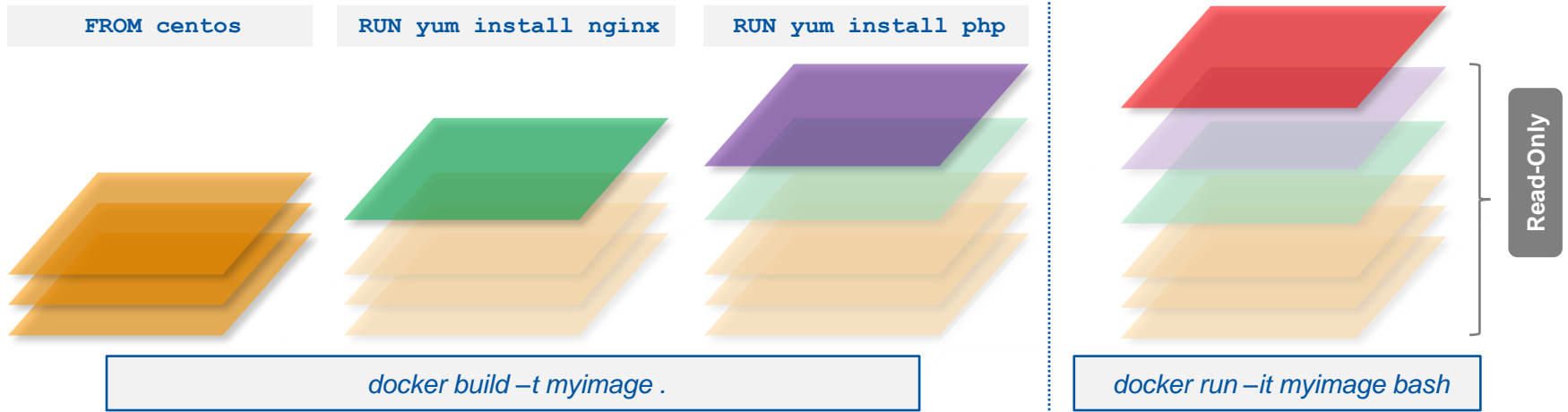
Quick Recap on Containers Images

- Image: Read-only template with instructions for creating a container
 - Produced as several layers (tar files) stacked together
 - Layering is used to improve storage utilization (can be reused)
 - Intermediate layers are hidden



Quick Recap on Containers Images

```
[root@ThinkPad-X1]# docker history myimage
IMAGE                CREATED              CREATED BY          SIZE
75cc2375258a        4 seconds ago      /bin/sh -c yum -y  66.9MB
e779b8a4024f        9 seconds ago      /bin/sh -c yum -y  77.8MB
470671670cac        4 days ago         /bin/sh -c #(nop)  0B
<missing>           4 days ago         /bin/sh -c #(nop)  0B
<missing>           7 days ago         /bin/sh -c #(nop)  237MB
```



Container Registries

- **Public:** Docker Hub
- **Private in cloud:** Amazon ECR, Microsoft Azure, Google, IBM, Alibaba
...
- **On Premise:** Red Hat Quay



Alibaba Cloud
Container Registry

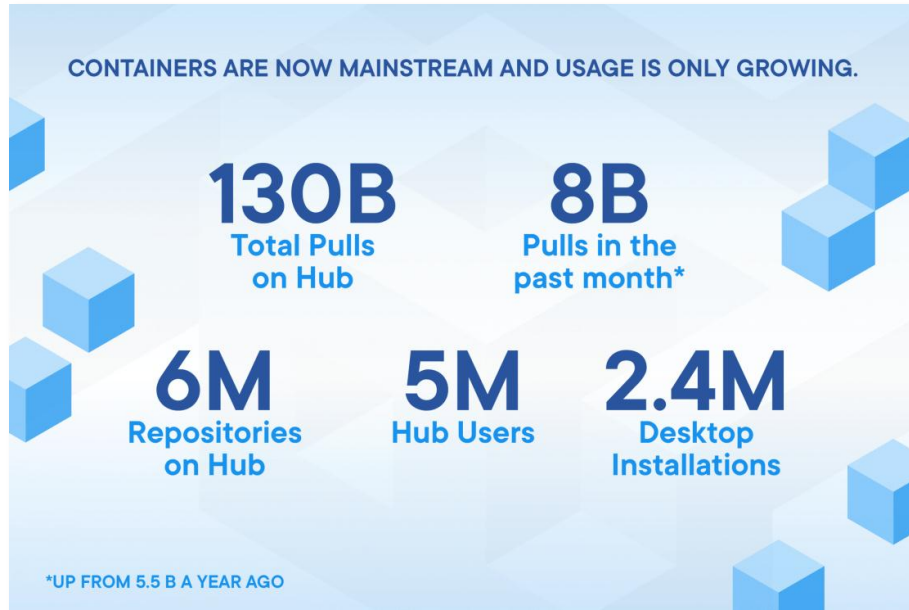


IBM Cloud
Container Registry



The Docker Hub Registry

- Most popular public registry – Docker's default

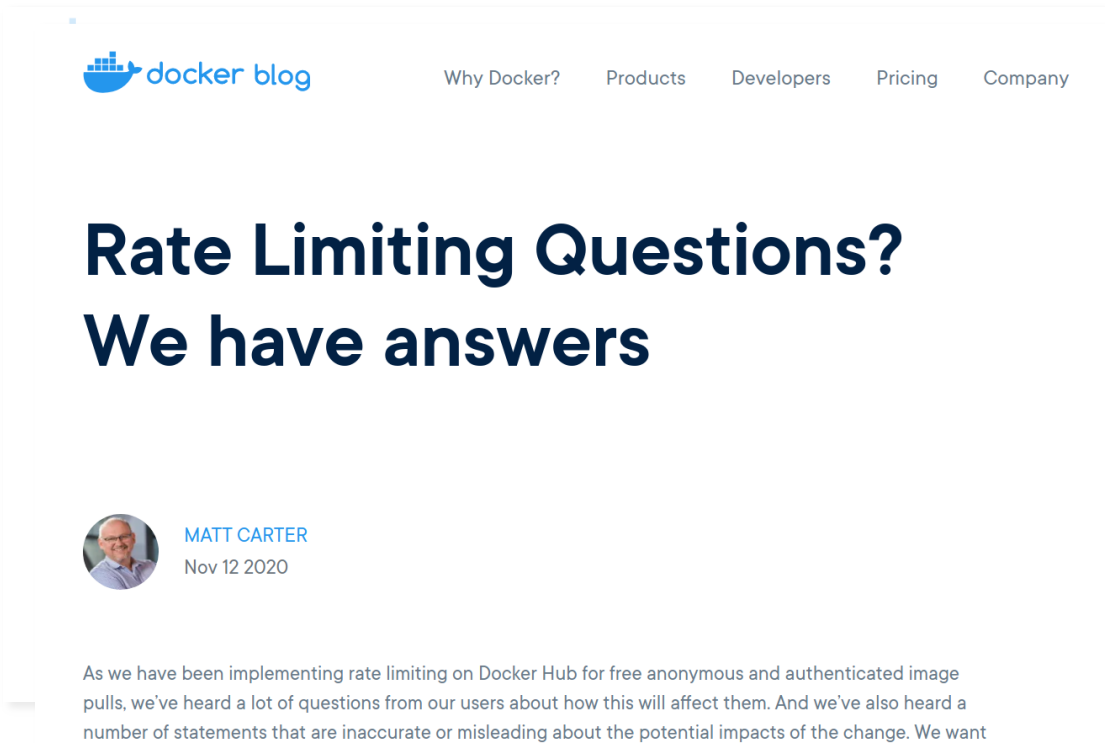


Docker Index

04 February 2020

<https://www.docker.com/blog/introducing-the-docker-index/>

The Free Lunch Is Over



The screenshot shows the Docker Blog header with navigation links: Why Docker?, Products, Developers, Pricing, and Company. The main heading is "Rate Limiting Questions? We have answers" in large, bold, dark blue text. Below the heading is a circular profile picture of Matt Carter, his name "MATT CARTER" in blue, and the date "Nov 12 2020". At the bottom of the visible text, it begins with "As we have been implementing rate limiting on Docker Hub for free anonymous and authenticated image pulls, we've heard a lot of questions from our users about how this will affect them. And we've also heard a number of statements that are inaccurate or misleading about the potential impacts of the change. We want

- Unauthenticated:
100 pulls / 6 hrs
- Free accounts:
200 pulls / 6 hrs
- Mirroring to private registries recommended

Containers at CERN

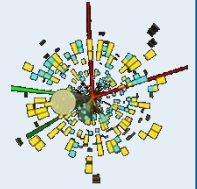
Images for Service Deployment

- Small images (< 1 GB)
- Run on few nodes
- Re-use from upstream



Images for Scientific Analysis

- Immutable unit for reproducibility
- ~10 GB per image
- Run on the Worldwide LHC Grid (potentially thousands of nodes)



▪ Currently using **GitLab Container Registry**

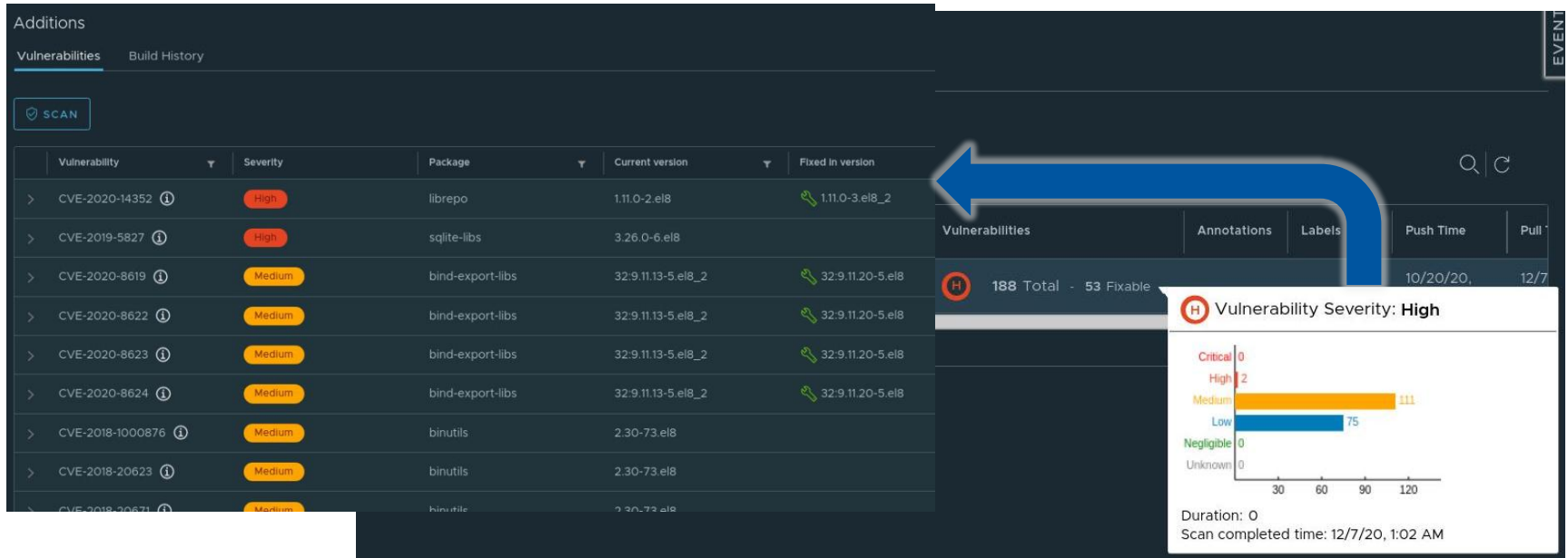
- ✓ Tight integration with CI pipelines, Registry storage associated to GitLab project
- ✗ No Garbage Collection of unreferenced blobs, No support for OCI artifacts

▪ Requirements:

- Vulnerability (CVE) scans, Storage for artifacts, GC
- Efficient storage and distribution of images at scale



- **Artifact Signing:** Ensure trusted source for artifacts being installed
- **Vulnerability Scanning:** Based on external plugins (e.g., Clair, Trivy, Sysdig)



The screenshot shows the Harbor web interface for vulnerability scanning. On the left, a table lists detected vulnerabilities with columns for Vulnerability ID, Severity, Package, Current version, and Fixed in version. A blue arrow points from the table to a summary card on the right. The summary card displays '188 Total - 53 Fixable' vulnerabilities and a bar chart showing the distribution of severity levels.

Vulnerability	Severity	Package	Current version	Fixed in version
CVE-2020-14352	High	librepo	1.11.0-2.el8	1.11.0-3.el8_2
CVE-2019-5827	High	sqlite-libs	3.26.0-6.el8	
CVE-2020-8619	Medium	bind-export-libs	32.9.11.13-5.el8_2	32.9.11.20-5.el8
CVE-2020-8622	Medium	bind-export-libs	32.9.11.13-5.el8_2	32.9.11.20-5.el8
CVE-2020-8623	Medium	bind-export-libs	32.9.11.13-5.el8_2	32.9.11.20-5.el8
CVE-2020-8624	Medium	bind-export-libs	32.9.11.13-5.el8_2	32.9.11.20-5.el8
CVE-2018-1000876	Medium	binutils	2.30-73.el8	
CVE-2018-20623	Medium	binutils	2.30-73.el8	
CVE-2018-20671	Medium	binutils	2.30-73.el8	

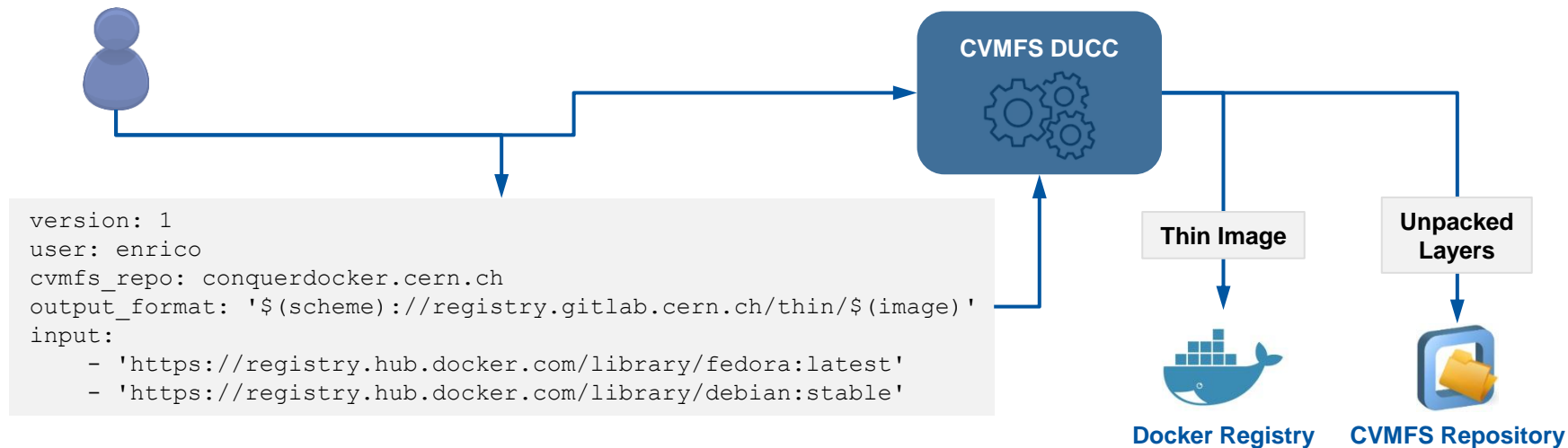
Vulnerability Severity: High

Severity	Count
Critical	0
High	2
Medium	111
Low	75
Negligible	0
Unknown	0

Duration: 0
Scan completed time: 12/7/20, 1:02 AM

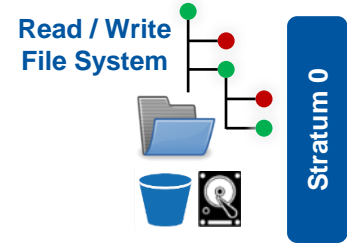
CVMFS ingesting Docker Layers

- **DUCC: Daemon to convert and publish unpacked layers**
 - Based on wishlist of Docker images to be ingested
 - Automatic generation and publication of thin image and unpacked layers

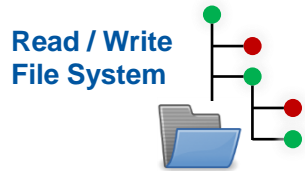


CVMFS Stratum 0s

- `cvmfs_server` package for repository management



```
# cvmfs_server transaction myrepo.cern.ch
# cd /cvmfs/myrepo.cern.ch && tar xvf myarchive.tar.gz
# cvmfs_server publish myrepo.cern.ch
```



Transformation

- Create file catalogs
- Compress files
- Calculate hashes

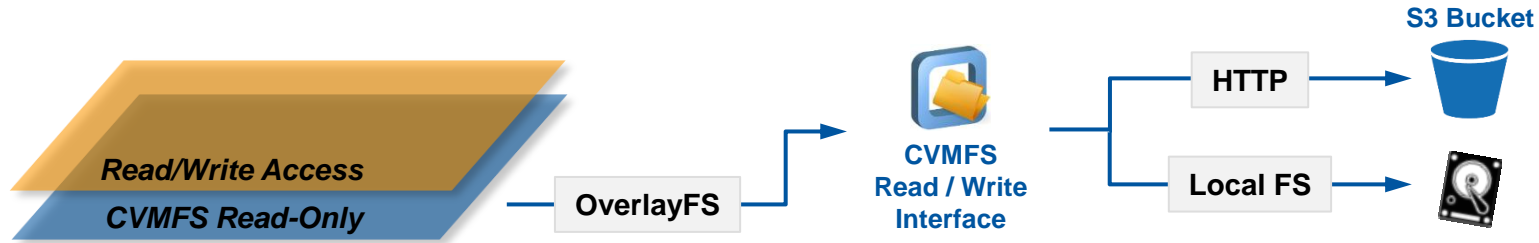
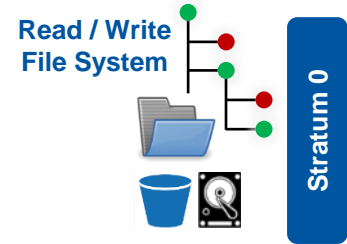


Content-Addressed Objects,
Merkel Tree

- Implicit file de-duplication via content-addressable objects
- Directory structure and file metadata stored in file catalogs

CVMFS Stratum 0s

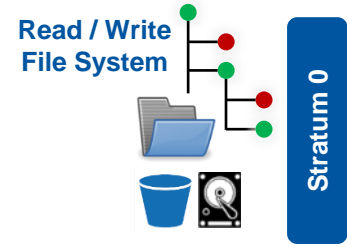
- `cvmfs_server` package for repository management
- Authoritative storage for repository content
 - Local file system
 - S3 compatible storage system (e.g., Amazon, Ceph)



- Updates applied by overlaying a copy-on-write union file system volume
- Changes are accumulated in the volume and synchronized afterwards

CVMFS Stratum 1s

- Stratum 1 servers in Europe, US, Asia
 - Reduced RTT to caches and clients
 - Improved availability in case of Stratum 0 failure



- RESTful CVMFS GeoAPI service
 - Clients submit request with desired resource and Stratum 1s list
 - Stratum 1 returns sorted list of Stratum 1s
 - Based on MaxMind IP database

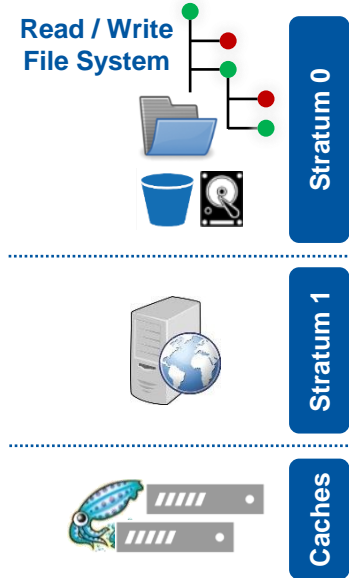


Stratum 1

```
HTTP GET
http://s1.cs3.org/cvmfs/<desired_resource>/api/v1.0/geo/<list_of_known_stratum1s>
```


Site Caches

- Off-the-shelf HTTP caching software
- Squid-cache as forward proxy
 - Recommended for clusters of clients
 - Reduced latency to clients and load on Stratum 1s
- Take advantage of cloud based CDNs
 - OpenHTC on CloudFlare
 - Helix Nebula Cloud (RHEA, T-Systems, IBM Cloud)



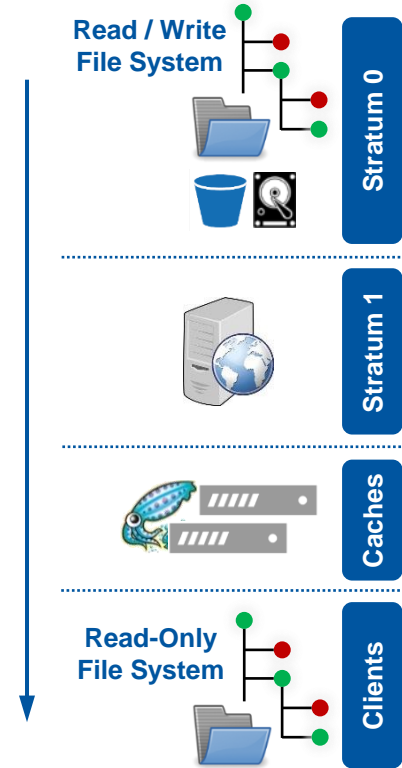
OpenHTC.io

HELIX
NEBULA
THE SCIENCE CLOUD

CLLOUDFLARE®

CVMFS Clients

- Client package for `/cvmfs` access
 - Runs on HPC, Supercomputers, end-users laptops
 - Docker support with in-container mount or CSI driver
 - Dynamic mounting of repositories with autoofs
- Local caching
 - Local file system (soft limit enforced)
 - Tiered: In-memory and disk
 - Alien: Cluster and Network file systems
- Embedded tools for troubleshooting and FS verification



Conclusions

- Containers are widespread and customary
 - For service deployment by practitioners in IT
 - For scientific analysis in the High Energy Physics community
- There is no one-fits-all solution for management and distribution
 - Harbor features advanced capabilities
 - CVMFS enables efficient distribution at scale
- ➔ Combine existing technologies
 - Simplify publication and management of images for end-users
 - Prototype demonstrated and running