AI: Some History And A Lot More About Matrices

Microservices Deployment Architecture With Kubernetes Clusters

₹120
ISSN-2456-4885

OpenSource ForYou

Volume: 10 | Issue: 09 | Pages: 100 | July 2022

THE COMPLETE MAGAZINE ON OPEN SOURCE

EFY GROUP Publication

Databases In Demand

HeatWave: Making MySQL Smarter

Blockchain Based Databases: Decentralised And Secure

Choose The Right Database For Your Project

YugabyteDB: The Reasons Why It's Popular



Why Companies Need To Set Up An Open Source Program Office

Interview:

Gerald Venzl, Senior Director, Product Management, Oracle Corporation



www.opensourceforu.com

Wanna Be Your Own Boss?

DO OPEN SOURCE.



Demand for Open Source is sky rocketing. Be it for managing IT infrastructure or development of software--Open Source solutions are what customers are seeking.

All you need to do is develop expertise in an Open Source stack, and then build a team around it!

And, Open Source For You can be your friend and a guide through this journey.

TO READ OUR PRINT EDITION Visit: https://subscribe.efyindia.com

TO READ OUR EZINE EDITION Visit: https://ezine.lfymag.com

WORLD'S LEADING PUBLICATION ON OPEN SOURCE

Looking for marketing solutions to engage with cutting edge techies?
Contact us at growmybiz@efy.in OR call us at +91-9811155335.



Does Your Antivirus Solution Provide You With Complete Protection?

Don't RELY on 20 years old technology to FIGHT current date viruses



TOP 10 things that your Antivirus solution should provide to tackle today's threat...

Ransomware File Protection	ATP- Advance Threat Protection	EDR- Endpoint Detection And Response	
Application Blocking	Deep Learning Malware Analysis	Exploit Prevention	URL Blocking
Disk and Boot Record Protection	Peripheral Control	Respond Investigate Remove (Root Cause Analysis)	

If it doesn't, contact us for a solution which can

Contact: Santosh on 9971696319 or Email at santosh.gupta@itsipl.com

ITS
A Total Solutions Company

I. T. Solutions India Private Limited
D-88/5, Okhla Industrial Area, Okhla Phase I, New Delhi -110020
Ph: 011-47695000 • Email: sales@ itsipl.com • www.itsipl.com

- Mumbai
- Jaipur
- Chandigarh

SOPHOS 2020 BEST NEXTGEN PARTNER | Delhi/ NCR
SOPHOS 2019 BEST NEXTGEN PARTNER | Delhi/ NCR
SOPHOS 2018 BEST SI PARTNER | Delhi

CONTENTS

JULY 2022 | ISSN-2456-4885

FOR U & ME

31 Why Telecom Service Providers Should Opt for Open RAN

47 Exnovation: Another Side of Innovation

49 AI Hardware that's Tailored for Computer Vision

53 How One Book Can Change Your Life's Trajectory

74 Where is DevOps Headed?

FOCUS

64 Are You Familiar with These Popular Open Source Databases?

66 Blockchain Based Databases: Decentralised and Secure

69 YugabyteDB: The Reasons Why It's Popular

71 Choose the Right Database for Your Project

DEVELOPERS

76 Julia and Python: Which Language is Quicker?

84 The AI and ML Services that Microsoft Azure Offers

89 How to Analyse Sentiments Using Machine Learning

ADMIN

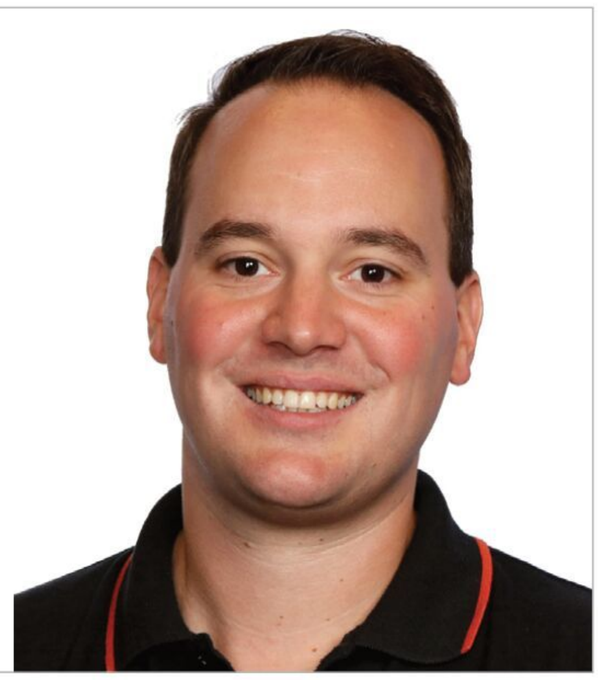
93 How to Run a Kubernetes Cluster Locally with Kind

97 Get Ready to Embrace the AIOps Era

14

"I wish the industry would not follow this ever increasing hype cycle for new stuff"

Gerald Venzl, senior director, product management, Oracle Corporation

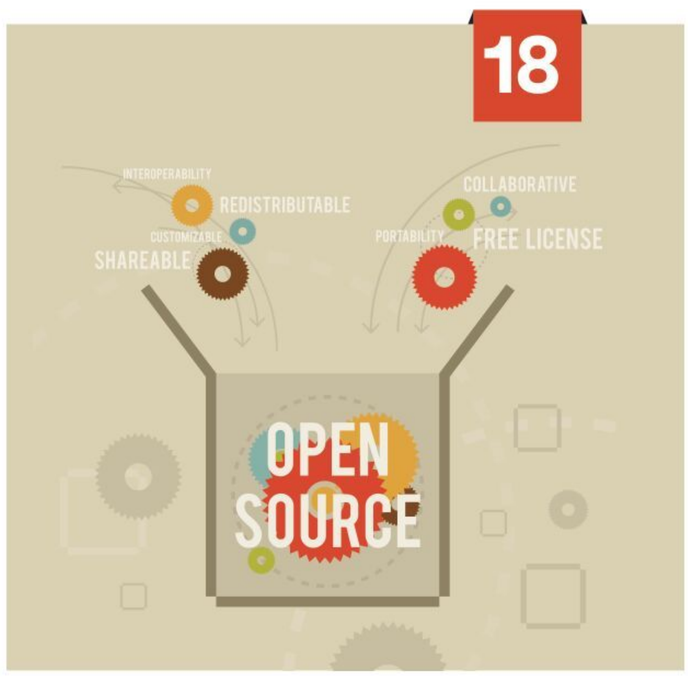


The Design Odyssey

22

Microservices Deployment Architecture with Kubernetes Clusters





Compliance: Why Companies Need to Set Up an Open Source Program Office



NFTs – Not Frivolous Tech

REGULAR FEATURES

07 FossBytes

CALLING EXPERTS

Share your knowledge at the 19th edition of
India's most exciting convention on Open Source!

19th Edition

OPEN SOURCE INDIA

29-30 SEP 2022

NIMHANS CONVENTION CENTER, BENGALURU

The Who's Who of Open Source Speak Here...

Open Source India (OSI) is the premier Open Source conference in Asia targeted at nurturing and promoting the Open Source ecosystem in the subcontinent. Started as LinuxAsia in 2004, OSI has been at the helm of bringing together the Open Source industry and the community.

Scan to Register

OR visit:
www.opensourceindia.in

www.OpenSourceIndia.in

EDITOR
RAHUL CHOPRA

EDITORIAL, SUBSCRIPTIONS & ADVERTISING
Delhi (HQ)
D-87/1, Okhla Industrial Area, Phase I, New Delhi 110020
Phone: +91-9811155335
E-mail: info@efy.in

MISSING ISSUES
Phone: +91-9811155335
E-mail: support@efy.in

BACK ISSUES
Phone: +91-9811155335
E-mail: support@efy.in

NEWSSTAND DISTRIBUTION
Phone: +91-9811155335
E-mail: efycirc@efy.in

ADVERTISEMENTS
NEW DELHI (HEAD OFFICE)
Phone: +91-9811155335
E-mail: efyenq@efy.in

MUMBAI
E-mail: rmowest@efy.in

BENGALURU
E-mail: rmosouth@efy.in

CHINA
Worldwide Focus Media
E-mail: china@efy.in

GERMANY
pms Plantenberg Media Service GmbH
E-mail: germany@efy.in

JAPAN
Tandem Inc.
E-mail: japan@efy.in

TAIWAN
J.K. Media
E-mail: taiwan@efy.in

UNITED KINGDOM
ASA Media
E-mail: uk@efy.in

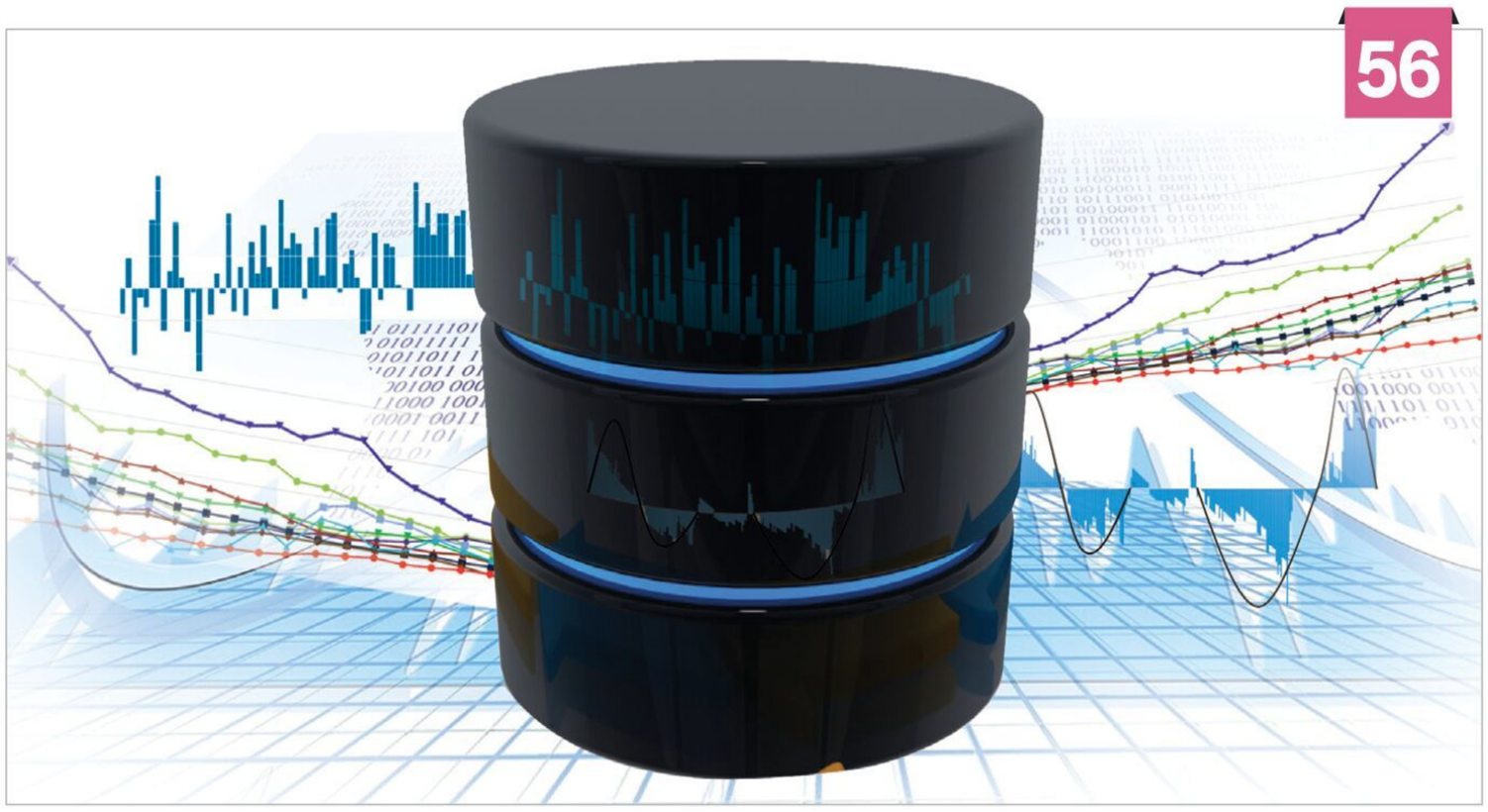
UNITED STATES
E & Tech Media
E-mail: usa@efy.in

Printed, published and owned by Ramesh Chopra. Printed at Tara Art Printers Pvt Ltd, A-46,47, Sec-5, Noida, on 28th of the previous month, and published from D-87/1, Okhla Industrial Area, Phase I, New Delhi 110020. Copyright © 2021. All articles in this issue, except for interviews, verbatim quotes, or unless otherwise explicitly mentioned, will be released under Creative Commons Attribution-NonCommercial 3.0 Unported License a month after the date of publication. Refer to http://creativecommons.org/licenses/by-nc/3.0/ for a copy of the licence. Although every effort is made to ensure accuracy, no responsibility whatsoever is taken for any loss due to publishing errors. Articles that cannot be used are returned to the authors if accompanied by a self-addressed and sufficiently stamped envelope. But no responsibility is taken for any loss or delay in returning the material. Disputes, if any, will be settled in a New Delhi court only.

SUBSCRIPTION RATES			
Year	**Newstand Price (₹)**	**You Pay (₹)**	**Overseas**
Five	7200	4320	—
Three	4320	3030	—
One	1440	1150	US$ 120

Kindly add ₹ 50/- for outside Delhi cheques.
Please send payments only in favour of EFY Enterprises Pvt Ltd.
Non-receipt of copies may be reported to **support@efy.in**—do mention your subscription number.

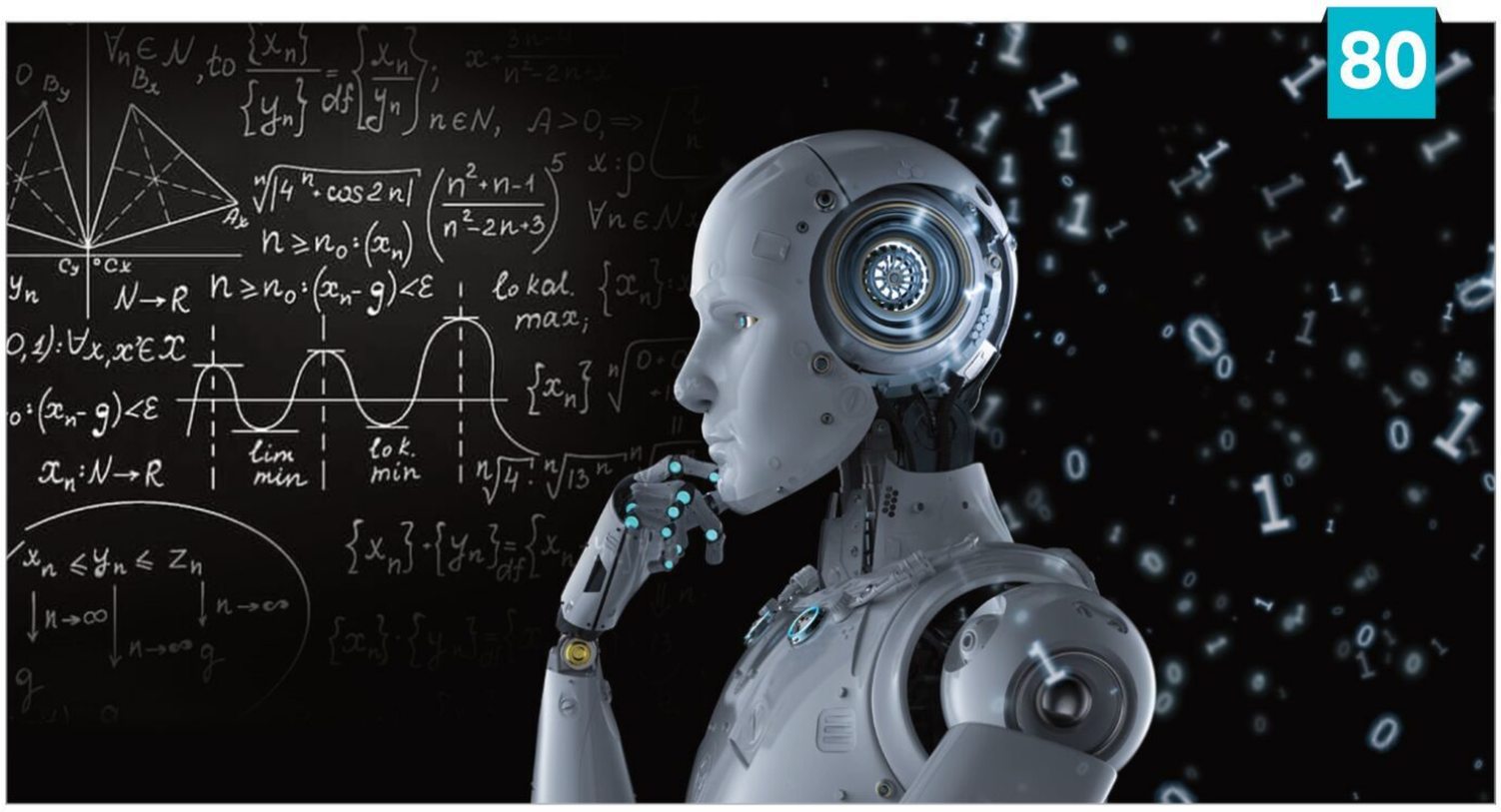
CONTENTS



Why We Need Time Series Databases for Site Reliability Engineering



HeatWave: Making MySQL Smarter



AI: Some History and a Lot More about Matrices

Google open sources PSP Security Protocol



Google has open sourced an encryption scheme it had developed to protect traffic between its data centres. PSP, which stands for PSP Security Protocol, was created to relieve Google's processors of the growing burden of software-based encryption, according to the company. PSP has been hailed as a success in the company's own environment, and the company has stated that it is "making PSP open source to encourage broader adoption by the community and hardware implementation by additional NIC [network interface card] vendors." PSP offloads encryption to NICs, which was previously possible with existing encryption schemes, but not at the scale or with the traffic coverage required by Google.

"At Google's scale," the company wrote when announcing its decision, "the cryptographic offload must support millions of live transmission control protocol (TCP) connections and sustain 100,000 new connections per second at peak."

Existing security protocols, according to Google Cloud's Amin Vahdat and Soheil Hassas Yeganeh, had flaws. "While TLS meets our security requirements, it is not an offload-friendly solution because of the tight coupling between the connection state in the kernel and the offload state in hardware. TLS also does not support non-TCP transport protocols, such as UDP," they stated.

Google added a custom header and trailer to the standard User Datagram Protocol (UDP) encapsulation to create PSP. PSP is currently implemented in three ways: one for Google's Andromeda Linux virtualisation kernel; one for its Snap networking system; and an application-layer version, SoftPSP, created so Google Cloud customers could use PSP on computers with traditional NICs.

Iterative releases MLEM, an open source machine learning tool

Iterative, an MLOps platform that raised US$ 20 million in a Series A round almost a year ago, has launched MLEM, an open source Git-based machine learning model management and deployment tool. The goal here, according to the company, is to bridge the gap between ML engineers and DevOps teams by utilising the Git-based approach that developers are already familiar with. Developers can use MLEM to store and track their machine learning models throughout their life cycle. As such, it

Kubecost develops OpenCost to rein in K8s costs

Kubecost's tool for monitoring and optimising spending on Kubernetes clusters has been released as an open source project. OpenCost is now available as open source software, according to Alex Thilen, head of business development at Kubecost, and the company has already submitted it to the Cloud Native Computing Foundation (CNCF) for approval as a sandbox-level project. Adobe, Armory, Amazon Web Services (AWS), D2iQ, Google, Mincurv, New Relic, and SUSE are among the project's founding members, in addition to Kubecost.

Although OpenCost is designed to run within a Kubernetes cluster, no data is sent outside of the cluster without user permission. It can collect data in real-time after only a few minutes of installation.

The primary issue addressed by OpenCost is overprovisioning of Kubernetes infrastructure. Many developers overprovision infrastructure to ensure maximum application performance. The problem is that much of that infrastructure is unused, and costs rise steadily as each new Kubernetes cluster is provisioned. According to Kubecost, organisations can cut Kubernetes-related cloud spending by 60-80 per cent without sacrificing application performance.

As the percentage of workloads running on Kubernetes clusters grows, it is more likely that those platforms will be managed centrally by an IT operations team. These teams are graded based on how well they optimise cloud infrastructure usage. They must also demonstrate to development teams how much Kubernetes infrastructure is consumed by individual applications.

Adobe releases a toolkit to contain visual misinformation

Adobe envisions a Web littered with photos and videos labelled with information about where they came from. The company's primary goal is to reduce the spread of visual misinformation, but the system could also benefit content creators who want to keep their names associated with their work.

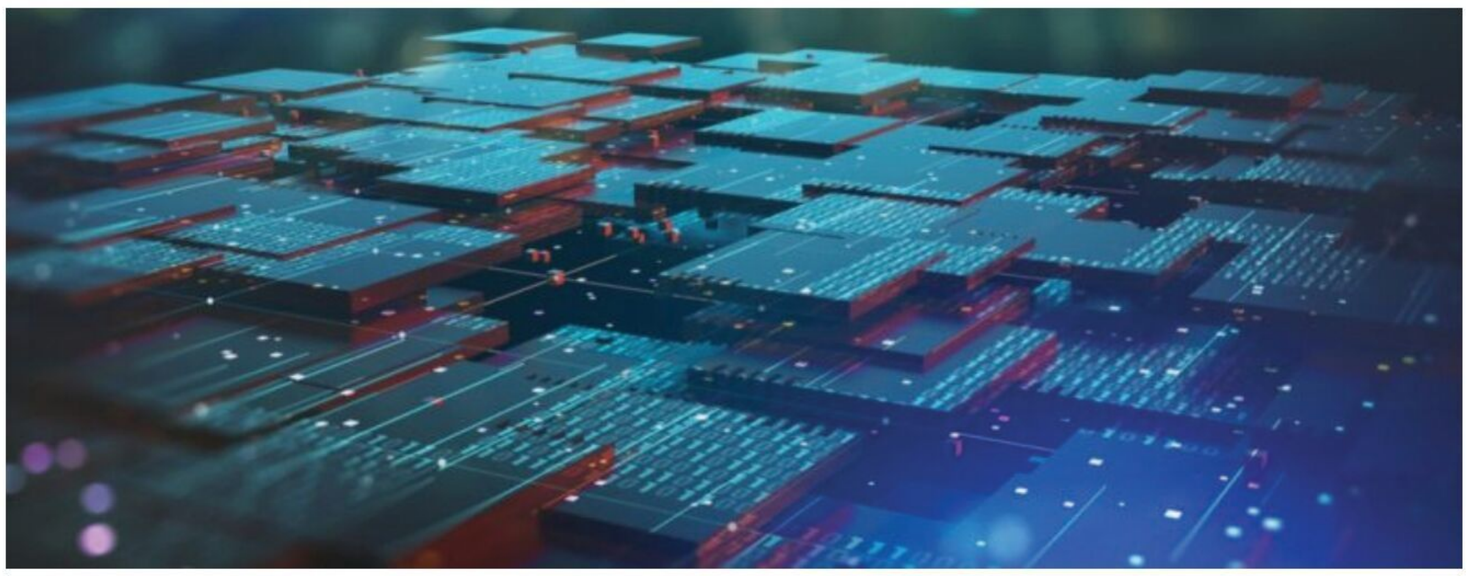
Adobe's Content Authenticity Initiative (CAI) project, first announced in 2019, has since released a whitepaper on a technology to do just that, integrated the system into its own software, and partnered with newsrooms and hardware makers to help universalise its vision.



The company is now announcing the release of a three-part open source toolkit to get the technology into the hands of developers and out into the wild. Adobe's new open source tools include a JavaScript SDK for developing ways to display content credentials in browsers, a command line utility, and a Rust SDK for developing desktop apps, mobile apps, and other experiences for creating, viewing, and verifying embedded content credentials.

In the same way that EXIF data stores information about aperture and shutter speed, the new standard also records information about a file's creation, such as how it was created and edited. And if the company's shared vision comes true, that metadata, which Adobe refers to as "content credentials," will be widely viewable across social media platforms, image search platforms, image editors, and search engines.

complements Iterative's open source GTO artefact registry and DVC, the company's data and model version control system.



A system like this allows for easier model sharing between business units and teams, and makes it easier for ML teams to collaborate with their DevOps teams. It also provides a single source of truth for determining the lineage of a given model in highly regulated industries. Iterative provides a hosted platform that does all of these things via its Iterative Studio service for collaborating on ML models, tracking experiments and visualisations, and hosting a model registry.

"Model registries simplify tracking models moving through the ML life cycle by storing and versioning trained models, but organisations building these registries end up with two different tech stacks for machine learning models and software development," said Dmitry Petrov, co-founder and CEO of Iterative. "MLEM as a building block for model registries uses Git and traditional CI/CD tools, aligning ML and software teams so they can get models into production faster."

Red Hat tests the 'NVK' Nouveau open source Vulkan Driver

Following the recent news about Nouveau reorganising code to allow its shader compiler to be used outside of Nouveau Gallium3D, Red Hat's Karol Herbst, a longtime Nouveau developer, has been posting patches for his new 'NVK' Nouveau Vulkan driver effort. NVK is a brand new, yet-to-be-merged open source Vulkan driver for NVIDIA graphics hardware. This Mesa based driver is currently being worked on primarily by Karol Herbst, who joined Red Hat several years ago and has since continued to work heavily on Mesa, including in the areas of OpenCL compute and other features. Aside from NVK, he has recently begun working on Rusticl, a Rust-based OpenCL implementation for Mesa.



Jason Ekstrand of Collabora, as well as David Airlie of Red Hat, have been making early contributions to NVK. NVK can at least run vulkaninfo, but it is still a work in progress, with the initial code only being committed two weeks ago. Apart from performance issues with newer generations of NVIDIA graphics cards, the lack of an open source NVIDIA Vulkan driver has been a major roadblock, given that most Linux games these days are Vulkan-native, and even Steam Play is mostly Vulkan with VKD3D-Proton/DXVK. This NVK driver will most likely be updated

in the future to support the open source NVIDIA kernel driver as an alternative to the Nouveau DRM driver. The original NVK open source Vulkan driver code is available on Nouveau's GitLab repository.

Spotify launches 'Basic Pitch', an audio-to-MIDI file converter

The R&D team at Spotify has released Basic Pitch, a free, open source tool for converting audio files to MIDI format. Basic Pitch, which uses machine learning to transcribe musical notes, can convert a recording of almost any instrument — including voice — to a MIDI version. Over the last 40 years, musicians who use computers to compose, produce, and perform their music have primarily used MIDI, a digital standard that acts as sheet music for computers, describing which notes to play when in an easy-to-edit format. Despite its widespread use, creating a composition from scratch with MIDI can be difficult. MIDI notes are typically generated by musicians using an interface, such as a keyboard designed for that purpose, or by manually typing the notes into their software.



Live performances with real instruments are frequently difficult for a computer to interpret, which is a problem for singers who are unfamiliar with piano keyboards or complex software. Even for musicians who are familiar with MIDI composition, it can be a time-consuming process.

"To solve this problem, researchers at Spotify's Audio Intelligence Lab teamed up with our friends at Soundtrap to build Basic Pitch — a machine learning model that turns a variety of instrumental performances into MIDI," said a company release.

Basic Pitch has advantages over other note-detection systems, such as pitch bend detection and tracking multiple notes across multiple instruments at the same time. It is also low on resources, allowing it to run faster than other systems of its type on most modern computers. Basic Pitch aims to provide musicians and producers with the "power and flexibility" of MIDI without the need for specialised equipment, allowing them to record whenever inspiration strikes and edit their compositions later.

Keyfactor unveils its community

Keyfactor, the machine and IoT identity platform for modern enterprises, has launched the Keyfactor Community. The community will provide knowledge and open source tools to developers, and operations and engineering teams to help them implement the best security solutions for their use case or product.



To build, deliver, and run applications securely, engineering and operations teams are increasingly relying on public key infrastructure (PKI) and machine identities. PKI provides developers

JFrog announces Pyrsia to secure software

JFrog Ltd, the liquid software company that created the JFrog DevOps platform, has announced Project Pyrsia, an open source software community initiative that uses blockchain technology to secure software packages from vulnerabilities and malicious code. Project Pyrsia, which is now accepting sign-ups, is an open source based, decentralised, secure, build network and software package repository aimed at assisting developers in establishing a chain of provenance for their software components, thereby increasing confidence and trust.

Open source software is an essential component of nearly every technology we use today. Nonetheless, there is no doubt that the volume, sophistication, and severity of software supply chain attacks have increased in the last year. The JFrog Security Research team has tracked over 20 different open source software supply chain attacks in recent months, two of which were zero-day threats. While open source components are intended to improve development efficiency, not knowing where your software comes from makes it difficult to identify risks, sowing doubt and uncertainty about its safety.

As a result, JFrog and other open source technology leaders such as Docker, DeployHub, Futureway, and Oracle have collaborated to create the Project Pyrsia network for validating the source and security of open source software packages. Pyrsia enables developers to use open source software with confidence, knowing that their components have not been compromised, without the need to build, maintain, or operate complex processes for securely managing dependencies.

Samsung and Red Hat collaborate to build advanced open source solutions

Samsung Electronics and Red Hat, Inc., a provider of open source solutions, have announced a broad collaboration on software technologies for next-generation memory solutions. The collaboration will focus on the development and validation of open source software for existing and emerging memory and storage products, such as NVMe SSDs, CXL memory, computational memory/storage (HBM-PIM, smart SSDs), and fabrics, with the goal of creating a large ecosystem for tightly integrated memory hardware and software.

The exponential growth of data driven by AI, AR, and the rapidly approaching Metaverse is causing disruptive changes in memory designs, necessitating more sophisticated software technologies that better integrate with the most recent hardware advancements.

This is the first time Samsung has collaborated with an open source software company to foster engagements across the IT marketplace. As part of this strategic partnership, Samsung will launch the Samsung memory research cloud (SMRC), where the two companies will develop and test software solutions on a variety of server platforms. The SMRC will serve as an open collaboration hub, allowing customers and partners to evaluate new software products in optimal combinations with memory hardware.

In addition, the companies will participate in open source communities such as the Linux Foundation to ensure that their software technology is fully supported in Red Hat Enterprise Linux and other open source software stacks. Samsung intends to launch its SMRC platform in the second half of this year, with the goal of serving as a continuous catalyst for innovation in future IT systems.

with a flexible and scalable security tool on which to build and integrate business-critical processes and infrastructure. The Keyfactor Community will make available to all users open source community editions of the company's leading products for cryptography, PKI, and digital signing.

Users of the Keyfactor Community will have access to community editions of EJBCA, SignServer, and Bouncy Castle, as well as free trials of enterprise editions on Azure and AWS. The community software already has a large user base, with thousands of people acting as quality assurance agents, downloading, testing, and using each new release.

"Open source technology creates an opportunity for innovation through collaboration and encourages transparency to build digital trust in the enterprise," said Tomas Gustavsson, chief PKI officer, Keyfactor. "The new Keyfactor Community empowers users to work closely with industry peers, and offers educational resources and access to world-class knowledge through a secure setting."

Kill Bill and Wovenware develop plugin to integrate Adyen payments gateway

Kill Bill, the open source billing and payment platform, and Wovenware, a provider of custom AI and software engineering solutions, have announced the creation of an open source plugin that integrates the global e-commerce payments provider, Adyen, to the Kill Bill platform, allowing for more payment options and processing of credit/debit cards, ACH, PayPal, Venmo, and many other payment methods.

"We're pleased that through our partnership with Kill Bill, Wovenware is able to help connect Kill Bill's payments platform to some of the world's leading payments providers, such as Adyen," said Wovenware CEO and co-founder Christian Gonzalez.



The Kill Bill code is free, and its architecture is highly modularised, allowing users to tailor it to their specific business requirements. Users of Kill Bill can use the plugin to offer more payment options to their customers. Wovenware can provide turnkey customisation for organisations with limited IT resources. To get the open source Adyen plugin, go to *https://github.com/Wovenware/killbill-adyen-plugin*.

"Leveraging the plugin that Wovenware created to integrate Kill Bill with Adyen, Kill Bill users now have access to one of the world's largest end-to-end payments solutions," said Kill Bill co-founder Pierre Meyer. "Wovenware's deep expertise and success in expanding the reach of the Kill Bill platform to new users reaffirms our decision in making the company our plugin partner of choice."

Exadel upgrades its facial recognition service

Exadel, a Silicon Valley based software consultancy and engineering firm, has updated its CompreFace open source facial recognition service with a slew of new features aimed at improving the customer experience and broadening its integration with other systems.

CompreFace, a face biometrics app that uses deep neural networks, released an initial version in November 2020 for a variety of industries including retail, hospitality, and travel. Exadel has now released version 1.0 to improve the user experience with a new user interface (UI), no-code options, ease of deployment, and new integrations.

The updated user interface manages subjects and images to begin facial collection without the use of scripts or code. Exadel claims that it also aids in the resolution of facial recognition issues by allowing users to check for correct images saved in each subject.



With the open source library Double Take, CompreFace developers can now write integrations that allow CCTV facial recognition without writing any custom code. Frigate, a network video recorder with real-time local object detection, is also linked to CompreFace for facial recognition. The file server system unRAID can also be integrated using the CompreFace and Double Take templates.

Serhii Pospielov, head of the AI practice at Exadel, said, "With the support of the developer community, Exadel has continued to improve the CompreFace user experience. As AI practices continue to rise, we are seeing more use cases emerge for facial recognition. This open source service allows anyone to enter the facial recognition market and access highly reliable services without needing deep coding or machine learning experience."

Atom text editor will officially be terminated later this year

On December 15 this year, Microsoft's GitHub will shut down Atom, its open source text editor that inspired and influenced widely used commercial apps such as Microsoft Visual Studio Code, Slack, and GitHub Desktop. The social code company stated that it is doing so to focus on cloud based software.

When Microsoft acquired GitHub in June 2018, Nat Friedman, the CEO at the time, reassured the GitHub community that Atom was still alive and well.

Atom has come to a halt after four years of progress. Apart from maintenance and security updates, the project hasn't seen significant feature development in several years, according to GitHub. Community involvement has declined during this time, and the business of locally installed software now appears less appealing than the potential recurring revenue, vendor lock-in, and information gathering enabled by cloud based apps.

Crane Venture Partners offers funding to early-stage founders

London based Crane Venture Partners (CVP) is a venture capitalist firm focused on open source based businesses.

The company claims that its first 2019 US$ 90-million fund led the industry in seed to Series A graduation rates. In other words, it excelled at assisting new businesses in getting off the ground. CVP has now raised US$ 140 million in its second fund to support early-stage founders.

This new fund focuses on artificial intelligence, data, and developer tools in addition to open source. The firm collaborates with early-stage founders to create the platforms and tools that will power tomorrow's software infrastructure. Crane does not simply sign a blank check and wait to see if you sink or swim. Its partners serve as extensions of the founders' teams. They are especially useful in areas where many technology companies lack expertise, such as marketing, sales, and customer success.



Crane's first fund invested 60 per cent of its assets in the United Kingdom, 20 per cent in Germany, and 10 per cent in Swiss companies. Its second fund is expected to invest 50 per cent of its assets in the United Kingdom, with the remainder spread across Europe, Israel, and the United States. Crane's second fund has already funded nearly 20 companies. Encord, Encore, Novu, Nuclia, Qovery, Restack, SeeChange, and Veratrak are examples.

Samagra launches a GovTech mentoring course

Samagra, an India governance consulting firm, has launched 'Code for GovTech' — an open source GovTech mentoring programme for engineering students (C4GT). This summer coding programme will run from May to August 2022, and one can apply through Samagra's official website. Those selected as contributors will also receive a stipend and a job opportunity with prominent GovTech firms, according to the firm.

Students will be able to learn from and interact with GovTech practitioners both inside and outside the government as part of the programme. They will also receive hands-on 1:1 mentoring from ecosystem pioneers such as EkStep, Tekdi, NavaDhiti, and others. The C4GT will be held on GitHub.



Commenting on the C4GT launch, Gaurav Goel, CEO and founder, Samagra, stated, "In a country of more than 130 crore citizens, the importance of technology in ensuring last-mile delivery of public services in a standardised manner cannot be underestimated. The role of GovTech in solving the pressing governance challenges of our times is only going to increase. And therefore, there is a need for the best engineering talent in the country to become a part of and lead the next wave of GovTech innovation. Code for GovTech has been designed as a way to create a tech talent pipeline which can solve some of the most complex challenges facing India.''

Nonetheless, Atom is expected to operate past its December 15, 2022 decommissioning date. GitHub intends to archive the Atom repository, but the code is open source and available to anyone who wants to champion the project.

Credentials of thousands of open source projects are at risk

Thousands of authentication tokens and other security-sensitive secrets are being leaked by a service that assists open source developers in writing and testing software. According to a new report by security experts, many of these leaks allow hackers to access the private accounts of developers on GitHub, Docker, AWS, and other code repositories.



The availability of Travis CI's third-party developer credentials has been an ongoing issue since at least 2015. HackerOne, a security vulnerability service, reported at the time that a GitHub account it used had been compromised after the service exposed an access token for one of the HackerOne developers. A similar leak surfaced in 2019 and again last year.

Anyone who has access to the tokens can read or modify the code stored in repositories that distribute an uncountable number of ongoing software applications and code libraries. The ability to gain unauthorised access to such projects opens the door to supply chain attacks, which involve threat actors tampering with malware before it is distributed to users. The attackers can use their ability to modify the app to target a large number of projects that rely on it in production servers.

Despite the fact that this is a known security risk, researchers from the Aqua Security firm's Nautilus team report that leaks have continued. A series of two batches of data accessed by the researchers using the Travis CI programming interface yielded 4.28 million and 770 million logs from 2013 to May 2022, respectively. The researchers discovered 73,000 tokens, secrets, and various credentials after sampling a small percentage of the data.

GitHub Sponsors now available to developers in India

Today's world runs on open source, and open source developers build software that so many people and organisations rely on. Financially supporting the people who build and maintain open source is key to ensuring a healthy and sustainable open source ecosystem. That's why GitHub Sponsors is now available to all developers in India. No more waitlisting. You can sign up right away if you have a bank account in India! Developers who were already on the waitlist are now in the program.

India has a thriving community of open source creators. In fact, the number of developers on GitHub in India now totals more than eight million and is growing, making it the second-largest developer community on GitHub. India's developers have made over 200 million contributions to open source software projects in the last year, and more than 10 million software projects from around the world depend on packages created in India. These contributions show how powerful and invested India's developers have become.



Anyone around the world can now support India's talented community and the open source projects they maintain. Through the GitHub Sponsors explore tool, it has become easier to discover projects and maintainers. The goal is to make it easier for the global community to explore dependencies and decide who to support.

DeepMind's MuJoCo's codebase is now available on GitHub

DeepMind, an Alphabet subsidiary and AI research lab, acquired the MuJoCo physics engine for robotics research and development in October 2021. The simulator was to be open sourced and maintained as a free, open source, community-driven project. DeepMind claims that the open sourcing is now complete, with the entire codebase available on GitHub.



MuJoCo, which stands for Multi-Joint Dynamics with Contact, is a physics engine designed to aid research and development in robotics, biomechanics, graphics and animation, and other fields that require fast and accurate simulation. MuJoCo can be used to implement model-based computations for machine learning applications such as control synthesis, state estimation, system identification, mechanism design, data analysis through inverse dynamics, and parallel sampling. It can also be used as a standard simulator, such as for gaming and interactive virtual environments.

According to DeepMind, MuJoCo has a comprehensive simulator capable of simulating complex mechanisms; it is readable, performant, portable code; and its codebase is easily extensible. It also has extensive documentation, including both user-facing and code comments.

For more news, visit *www.opensourceforu.com*

K-9 and Thunderbird to work together on the new Android email app

Thunderbird, the open source email client, is coming to Android via the K-9 Mail Android email app project, which is merging with Thunderbird to form the Thunderbird Android email app. Thunderbird was moved to the Mozilla Foundation's subsidiary MZLA Technologies Corporation two years ago, in a similar ownership structure to Firefox under the foundation's subsidiary, Mozilla Corporation. With new features such as OpenPGP end-to-end encryption and a long-awaited mobile app, the Thunderbird project was able to forge its own path.



According to the Thunderbird team, discussions began in 2018 between Thunderbird product manager Ryan Lee Sipes and K-9's lead maintainer Christian Ketterer about a possible collaboration for a Thunderbird email app. In 2022, the two have decided that rather than Thunderbird developing its own app from scratch, K-9 should simply join Thunderbird.

"Many Thunderbird users have asked for a Thunderbird experience on the mobile, which we intend to provide by helping make K-9 amazing (and turning it into Thunderbird on Android). K-9 will supplement the Thunderbird experience and enhance where and how users are able to have a great email experience. Our commitment to desktop Thunderbird is unchanged; most of our team is committed to making that a best-in-class email client and it will remain that way," Thunderbird said.

“I wish the industry would not follow this ever increasing hype cycle for new stuff”



While new technologies can lead to innovations, hype often goes with the territory. ***Gerald Venzl, senior director, product management*** at ***Oracle Corporation*** speaks with ***Abbinaya Kuzhanthaivel*** about the risks of following the hype, and shares his thoughts on how to grow a career by contributing to open source projects.

Gerald Venzl, senior director, product management, Oracle Corporation

Q. Can you tell us a bit about your journey with open source?

A. I currently work as a database product manager looking after the Oracle Database. As part of my job, I help steer the direction of the product and help pinpoint priorities in the market that need to be addressed. Often people are surprised when I tell them that open source is an important aspect of that. For example, Oracle recently launched a Kubernetes operator for its database, which is fully open source and is available on GitHub. But we have also Docker build files on GitHub, and even some of our Oracle Database drivers are fully open source, such as our Node.js driver and the Python driver.

Q. Is Oracle your first venture into open source?

A. I was accustomed to open source in my personal life before joining Oracle. For me, open source is a natural part of my job.

Oracle has a long-standing commitment to open source with Oracle Linux, Java, MySQL and many other projects. Although the Oracle Database core is proprietary, we see the value of open source and having people to contribute and understand the technologies around it.

Q. What do you think about open source as an opportunity?

A. The thing that I like about open source is that it’s essentially a democracy and everybody gets to participate. Not that all participants are core contributors, but it connects people from all over the world. The exposure to projects and other developers is important for growing your skillset and your career.

In Oracle, we are currently focusing a lot on the Kubernetes operator. That’s an exciting project. In my free time, I also write little tools and open source them. I had



5243 ATTENDEES | **70+** SESSIONS HOSTED | **100+** SPEAKERS | **18** COUNTRIES REPRESENTED

18th Edition

OPEN SOURCE INDIA

Thank You

VISITORS, for making the event a huge success

SPEAKERS, for contributing your valuable thoughts

PARTNERS, for your support

Microsoft Azure | soda foundation | Google | aws

MySQL | HUAWEI | iTEXT | GitHub

SUSE | ENTREPRENEUR FIRST JOINEF.COM | TRUEFOUNDRY BY ENSEMBLE LABS | Application Library Engineering Group

openLooKeng | EDGE GALLERY | soda foundation meetupindia | BANGALORE CNCF MEETUP

www.OpenSourceIndia.in

For more details, call on +91-9667743666 or email us at info@opensourceindia.in

worked a little on a command line tool called 'csv2db', that allows you to load CSV data into a database. I see these projects as a chance to educate myself. You read something, and then you try to implement it and see whether you really understood it, whether it works and so on.

Q. How did this change happen from being a developer to a product manager?

A. My path was probably highly irregular. I am from Austria originally where I worked as a developer for an American company headquartered in New York. I was part of the performance engineering team. I was actually one of those guys who tried to profile the code and those skills eventually brought me to troubleshoot production systems at customer sites. I relocated to New York, and then a couple of years later Oracle asked me to join their pre-sales team as a result of our earlier interactions in running proof-of-concepts and performance tests on their systems together.

I liked the opportunity to learn by interacting with customers directly and discussing requirements. But for me, the turning point was moving back into development and specifically into the technical aspects as product manager.

Q. What are the top two 'must have' qualities of your role?

A. I think one must definitely be able to accept ever-changing requirements. But it's not just a reactive job. You also have to have a curious mindset. If you have a curious mind, are open to new ideas, and can think out-of-the-box, you can establish your priorities and grow into the role.

Q. What would you like to say about the challenges in your current role?

A. Product manager is a very diverse and ever-changing role. One must be highly adaptive because you need to go through steering the product, prioritising the internal roadmap and also accept changes depending on market demands and customer needs. So the key challenge, and also the fun part, is that no day really ever looks the same.

Q. Any important risk you have taken so far which you think might inspire others?

A. I think my shifting into product management was the biggest risk. The future is unpredictable and one must dare to take such risks. For open source projects, it's probably taking the step and actually wanting to contribute to a project. It can be a bit daunting in the beginning, especially to know where to start contributing for very popular active projects. Not all contributions have to be done in code; often people appreciate help with the documentation or testing. There are many ways to get started — the important thing is that you do get started. If you are passionate about something, you will have to go after it. You may have people not reacting the way you expect at first, but that's okay. It will essentially help to learn.

Q. What do you consider as top leadership qualities?

A. There are a couple of things that I think are important for leadership. And the first one is that it's okay to be wrong and acknowledge it. This also encourages people around you to freely share ideas. The other thing would be to take chances and get out of your comfort zone. I have never formally learnt product management. I just was intrigued by it, gave it a try and I thought, let's see how it goes. It has put me in a good space for growth.

Q. Anything you are not very happy about when it comes to open source?

A. I wish the industry would not follow this ever increasing hype cycle for new stuff. New doesn't automatically mean great. We may talk just about all the new things out there, but the world still runs on Linux and, remember, we still use the HTTP, TCP/IP and all those technologies today. The fundamental technologies that connect us around the globe have been there for a long time. Something doesn't have to be new to be great and often new technologies go just as fast as they came.

Q. What are the major risks in going ahead with new technologies?

A. A major risk is forgetting or not wanting to ask the "why" we need the new technology. In our industry we get excited very quickly about something that we then want to work with. Sometimes that means that we oversimplify some business requirements and kind of omit the downsides of a new technology, just so that we can use it for a new project. I agree that new things lead to innovations and there is nothing wrong with that. But I have equally seen just as many projects fail that tried to replace the legacy system because a new buzz technology is out there and looks attractive. I've seen a three or four year-long project fail because no one bothered to look at the 'why' when replacing the previous system; and although it solved the new requirements, people forgot to ask themselves what the old system did well and they just ended up with those old issues again.

Q. How does one keep away from the hype around new technologies?

A. I would say that, as a developer, don't just blindly follow the latest and greatest. If someone is telling you new stuff, well it's great to know. But it's no surprise to me that Linux runs the world and HTTP runs the Web, because those technologies are really well-designed.

Have an open mind and look at what's new, but think about whether this new technology actually will serve your needs. It's fine if it doesn't.

Q. How can a developer find projects to contribute to while keeping away from the hype?

A. There's nothing wrong with working on a hype project, but you have to make sure that you actually have interest. Think small — don't expect to become the main contributor in the next two weeks or in a short span.

Remember most of the open source projects really like non-code contributions just as much, such as testing or reporting bugs or lack of information in the documentation. Don't just go into it to write some code. Initially have some idea about which area you want to contribute to. It should be something that excites you. Go to GitHub and just read through project contributing guidelines. It will tell you what contributions the project needs and how to make them correctly.

Your work may be small, like adding a sentence to the doc or correcting a typo. But it will allow you to get familiar with the process and with the other people involved in these projects. Do not expect to jump into a project and change its core. Most likely, only an approved committee of committers can actually change that part of the code. Build some trust, show that you understand the project and over time your involvement will grow naturally.

Q. Any examples of projects you think were overrated because they had just used a new technology?

A. I have seen a few working in the database space. When you think about it, relational databases have been around for a very long time and are still going strong. To some extent, we have forgotten why relational databases became so popular. The goal was to organise the data in a way so that five years from now somebody who comes in and has no clue what the data looked like, can make sense out of it. For a while there was a general hype that you no longer need any database, whether relational or non-relational, because Hadoop will do everything. And it didn't. It actually just led to data cleansing issues for many folks. Don't get me wrong — there are companies that successfully run Hadoop clusters and there is nothing bad about the technology itself. But you have to understand what it is and when to use it.

Q. Oracle has recently introduced the new MongoDB API for its autonomous database. What was the reason behind it and how did it happen?

A. At Oracle, we follow the converged database methodology. This methodology focuses on bringing the algorithms and computation to the data, rather than the other way around. A very good analogy for converged database methodology is the smartphone, where you can do multiple use cases in one device, like taking a picture and sending it to a friend while being on a phone call, for example. In recent years, we have seen a proliferation of vendors pitching their technology to address an often simple use case For example, developers like working with JSON documents and MongoDB allows them to store and retrieve these documents. But it is one thing to store and retrieve these and another to analyse terabytes of them in real-time. We think SQL is a really good language for any kind of analytics and we have the best database for mixed workloads, i.e., allow real-time analytics while transactional workloads are running. Additionally, Oracle Database has been managing JSON documents natively since 2014.

Developers love the MongoDB API as it makes database interaction very natural for them. And we have the best database for analytics and mixed workloads that can also manage JSON documents natively. So we decided to give developers the best of both worlds — the same MongoDB API on top of the world's leading database.

Q. Will you say MongoDB is a new hype?

A. The JSON format is a very useful hierarchical format and is nothing new — it's been around since 2001. If you want to use JSON, then go ahead, by all means. Oracle has done a lot of work to introduce JSON operations into the SQL standard, and we see more and more databases supporting these standardised operations. You will find that if you want to work with JSON, retrieve JSON documents, query and manipulate data in JSON documents, you definitely don't need to have a document store anymore. MongoDB is a cool technology, but so were XML databases in their days. I think it is definitely not needed for data management aspects.

Q. Any hiring plans?

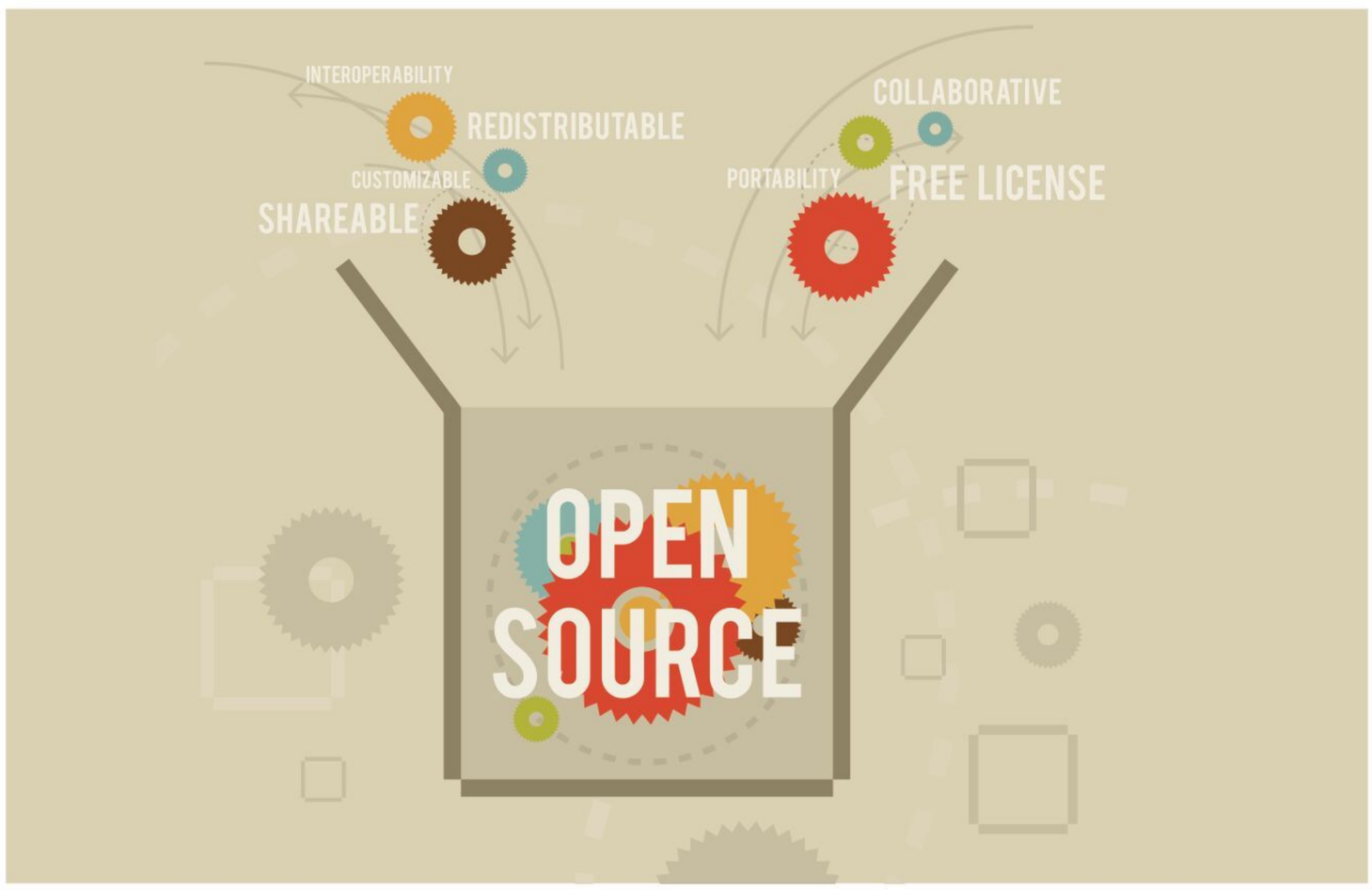
A. We are constantly hiring great talent and all our openings can be accessed on the Oracle careers page. We are looking for people in a variety of different roles, including engineering and product management. We are strong in diversity and have people from all around the world and all ages, including graduate students.

Q. Your message for our readers.

A. Programming is a universal language and it's great to be a developer and write programs. Don't be shy, be always ready to try something new and get out of your comfort zone to do things you are passionate about. END

Why Companies Need to Set Up an Open Source Program Office

Managing the use of open source software and decreasing compliance risks is key to the success of any software product. An open source program office can help an organisation do just that. Find out how.



Open source software (OSS) is integral to building a modern software solution. Be it an internal or a customer facing solution, organisations rely significantly on open source software today. OSS components are governed by their unique licence terms, and non-compliance with these can often expose organisations to security and intellectual property (IP) risks which eventually may hamper a company's brand value.

When development teams are delivering a software release, they are primarily trying to meet project deadlines. Therefore, the tracking of versions of components and libraries, or the third party code pulled into the project, is not as rigorous as it should be. This means that licences and vulnerable OSS components can enter the code base and be delivered to customers. This can be risky for both the customer and the company delivering the software solution.

Another increasingly challenging area is that of developers contributing to open source projects. Companies can reap numerous benefits if they do so. This includes keeping skills current, retention of staff, attracting developers to work for the organisation, and improving the image of the company. Many open source projects require developers to sign a contributor licence agreement. This states that any IP created by the developer belongs to the project and not to the contributing developer. In this scenario, organisations need to be careful that IP and trade secrets that are not open source are not being signed over to open source projects.

Developers need to be educated about open source licensing issues, determining what to leverage, when or how much they can contribute to the community, and what packages might bring risk to the organisation's reputation. All this can be streamlined by putting a strategic policy and operations in place. One way of doing this is by creating an entity that is dedicated to working around all things open source—an entity called the open source program office (OSPO).

An OSPO creates an ecosystem for employees to use open source software in a way that compliance risks are kept at bay. The role of an OSPO is not limited to supervising open source usage; it is also responsible for contributing back to the community and managing the company's growth in the market by actively engaging in events, as well as conducting webinars and campaigns.

In this article we will see why there is a need for building an OSPO, and how it has emerged as a prominent entity for any open source policy and governance programme.

Why should you have an OSPO?

With the wide use of open source software, regulating its usage and keeping the compliance strategy in check can be often overwhelming for the teams involved in the product development cycle.

Developers often overlook licence obligations, and sometimes the management or stakeholders are also not fully aware of the implications of

non-compliance with these open source licences. OSPO handles open source software right from its on-boarding till the time it is delivered to the end user and everything inbetween, irrespective of whether it is being used for internal or external purposes.

An OSPO builds a solid foundation by starting compliance and regulatory checks in the early software development life cycle. This usually begins by guiding and aligning the involved team members towards a common path that benefits the organisation's values. The OSPO puts in place policies and processes around open source usage and governs the roles and responsibilities across the company.

To conclude, it aligns the efforts of all relevant teams involved in building the product and helps increase the organisation's capacity for better and effective use of open source.

The function of an OSPO

The function of an OSPO may vary from organisation to organisation depending on the number of its employees and the number of people that are part of the OSPO team. Another factor is the purpose of using open source. An organisation may only want to use open source software for building the product or may also look at contributing back to the community.

Evaluating factors such as which open source licences are appropriate or whether full-time employees should be contributing to an open source project may be part of the OSPO's role. Putting a contributor licence agreement (CLA) in place for developers that are willing to contribute and determining what open source components will help in accelerating a product's growth and quality are some other roles of an OSPO.

Some of the key functions of an OSPO involve:

- Putting an open source compliance and governance policy in place to mitigate intellectual property risks to the organisation
- Educating developers towards better decision-making
- Defining policies that lay out the requirements and rules for working with open source across the company
- Monitoring the usage of open source software inside as well as outside the organisation
- Conducting meetings after every software release to discuss what went well and what could be done better with the OSS compliance process
- Accelerating the software development life cycle (SDLC)
- Transparency and coordination amongst different departments
- Streamlining processes to help mitigate risks at an early stage
- Encouraging members to contribute upstream to gain the collaborative and innovative benefits of open source projects
- Producing a report with suitable remediation and recommendations for the product team
- Preparing compliance artifacts and ensuring licence obligations are fulfilled

Building an OSPO

The OSPO is typically staffed with personnel from multiple departments within the company. The process involves training and educating the relevant departments regarding open source compliance basics and the risks involved in its usage. It may provide legal and technical support services so that the open source requirement goals are met.

An OSPO may be formed by the following people within the organisation (this is a non-exhaustive list of people who can be a part of it):

- **Principal/Chief:** This role can be taken by the flag bearer, the one who runs the OSPO. The chief knows the various aspects of using open source like the effect of using different components, licence implications, development and contributing to the community. These requirements are entirely dependent on an organisation's needs.
- **Program manager:** The program manager sets the requirements and objectives for the target solution. He/she works alongside the product and engineering teams to

The rise of the OSPO

Companies like Microsoft, Google and Netflix have well established OSPOs within their organisations. Many others, like Porsche and Spotify, are building their own OSPOs to leverage the usage of open source in an efficient way.

Here is what leaders from renowned companies have to say about OSPO practices.

- "As a business, it's a culture change," explains Jeff McAffer, who ran Microsoft's Open Source Program Office for years and now is a director of products at GitHub focused on promoting open source in enterprises. "Many companies, they're not used to collaboration. They're not used to engaging with teams outside of their company."
- "Engineering, business, and legal stakeholders each have their own goals and roles, oftentimes making trade-offs between speed, quality, and risk," explains Remy DeCausemaker, head of open source at Spotify. "An OSPO works to balance and connect these individual goals into a holistic strategy that reduces friction."
- Gil Yahuda, Verizon Media's OSPO leader, states, "We seek to create a working environment that talent wants to be part of. Our engineers know that they work in an open source friendly environment where they are supported and encouraged to work with the open source communities that are relevant to their work."

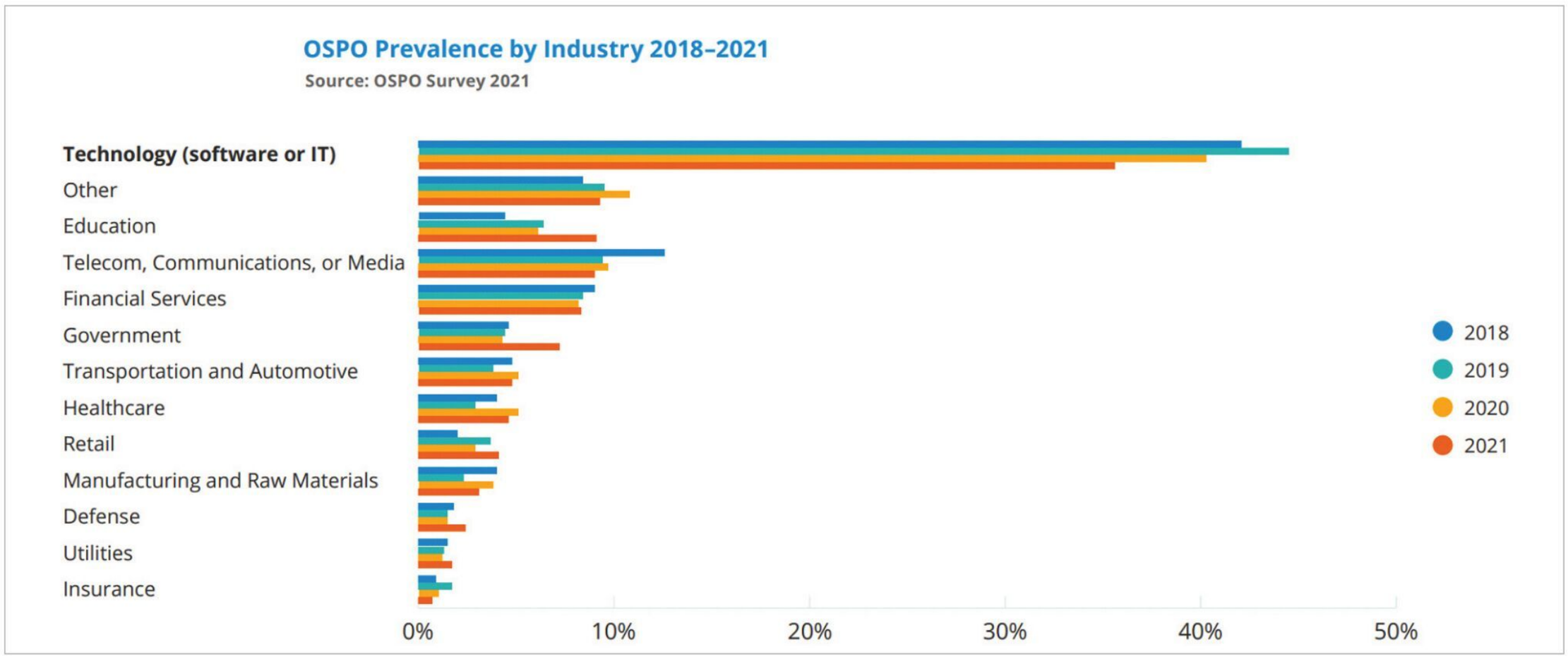


Figure 1: OSPO prevalence by industry 2018-2021 (Source: *https://github.com/todogroup/osposurvey/tree/master/2021*)

connect workflows. This includes ensuring that policies and tools are implemented in a developer-friendly manner.

- **Legal support:** Legal support can come from outside the firm or in-house, but is an important part of an OSPO. The legal role works closely with the program manager to define policies that govern OSS use, including which open source licences are allowed for each product, how to (or whether to) contribute to existing open source projects, and so on.
- **Product and engineering teams/ developers:** The engineering team should be well-versed with open source licence(s) and their associated risks. The team must seek approval from the OSPO before consuming any open source component. The team may have to be trained with respect to open source compliance basics and its usage at regular intervals
- **CTOs/CIOs/stakeholders:** A company's leadership has a huge impact on the OSPO strategies. The stakeholders have a great say in the decision making process for any product/solution's delivery. Due to the nature of the OSPO's function within a company, the VP of engineering, CTO/CIO, or chief compliance/risk officer must get involved in the OSPO.
- **IT teams:** Having support from the IT department is very important. An OSPO may be tasked with implementing internal tools to improve developer efficiency, monitor open source compliance, or dictate open source security measures. IT teams are key in helping to connect workflows, and ensure policies are implemented in a developer-friendly manner.

In the 2021 State of OSPO Survey conducted by the TODO Group, the key findings were:

- There are many opportunities to educate companies about how OSPOs can benefit them.
- OSPOs had a positive impact on their sponsor's software practices, but their benefits differed depending on the size of an organisation.
- Companies that intended to start an OSPO hoped it would increase innovation, but setting a strategy and a budget remained top challenges to their goals.
- Almost half of the survey participants without an OSPO believed it would help their company, but of those that didn't think it would help, 35 per cent said they haven't even considered it.
- 27 per cent of survey participants said a company's open source participation is very influential in their organisation's buying decisions.

The use of open source software when building any software solution is almost inevitable today. However, the open source licence risks cannot be overseen. What is needed is a strategic streamlining process that helps combat the compliance issues that come in the way of using open source components effectively.

An OSPO helps set a regulatory culture by building a centralised dedicated team that educates employees and brings awareness regarding everything related to open source usage in an organisation. An OSPO can also work as a guide to fetch top talent from the industry, which will eventually be a boon for business goals. END

By: Sakshi Sharma

The author is an open source consultant at Source Code Control Limited.

EFYGROUP
YOURS SINCE 1969

Wanna Support Open Source For You?

Subscribe to the Magazine— so that we can keep promoting Open Source.



AMAZING OFFER

Pay for 6 Issues Get 12!

(Buy One Get One Free!)

WORLD'S LEADING PUBLICATION ON OPEN SOURCE

TEAR OUT — **ORDER FORM** — TEAR OUT

Please Tick (✓) Your Choice

☐ **Pay for 12 Issues Get 24 Issues**
and save 50%
(Buy 12 Issues Get 12 Issues Free!)
₹1440

☐ **Pay for 24 Issues Get 48 Issues**
and save 50%
(Buy 24 Issues Get 24 Issue Free!)
₹2880

☐ **Pay for 36 Issues Get 72 Issues**
and save 50%
(Buy 36 Issues Get 36 Issue Free!)
₹4320

To subscribe online, visit
https://tinyurl.com/y5kuv4la

OR **SCAN THIS CODE**

Name_____ Organisation_____ Mailing Address_____

_____ City_____

Pin Code_____ State_____ Phone No._____ Email_____

Subscription No. (for existing subscribers only)_____. **I would like to subscribe to the above (✓)marked Open Source For You magazine starting with the next issue.** Please find enclosed a sum of Rs_____by DD/MO/crossed cheque bearing the No._____ dt._____in favour of **EFY Enterprises Pvt Ltd,** payable at Delhi. *(Please add Rs 50 on non-metro cheque)*

Please mark one (nearest) relating to your subscription: ☐ **Indian Company** ☐ **MNC** ☐ **R&D organisation** ☐ **Engineering institute** ☐ **College/School** ☐ **Any other (specify):**_____

Send this filled-in form or its photocopy to : **EFY Enterprises Pvt Ltd,** D-87/1 Okhla Industrial Area, Phase 1, New Delhi 110 020 | Ph: **011-40596600** | e-mail: **support@efy.in**

Terms:- # These rates are applicable for new subscribers as well as renewal by existing subscribers. # Can access ezine till your subscription is active # The rates are valid for subscribers within India only. # Please allow 4-6 weeks for processing of your subscription. # The subscription copies will be dispatched through ordinary post only # Subscription Agents will not get agency commission against this scheme # Disputes, if any, are subject to exclusive jurisdiction of competent courts and forums in Delhi/New Delhi only. * Replacement will be made if intimation of damaged / non-receipt of copies is received within 30 days of its publication ** After three months, if you are not satisfied with the magazine, your balance amount will be returned (Not applicable for gift offer)

Microservices Deployment Architecture with Kubernetes Clusters

Scalability and resilience are two of the most important reasons to move from monoliths to microservices. The Kubernetes platform offers both while orchestrating containers. In this Part 9 of the series, reference architecture for a user management system is presented and demonstrated around Kubernetes. This architecture includes Spring Boot microservices, Apache Kafka and MySQL.



Took at any e-commerce business. It flourishes during the weekends and on special days. The business is normally low till noon and peaks in the evenings. So the systems that back such e-commerce, banking and government services, etc, experience different loads at different points of time, and need to be scaled up or down as automatically as possible. Such systems require appropriate deployment architecture and orchestration tooling.

In the previous part of this series of articles, we have seen how docker-compose is useful in deploying multiple containerised services all at once, on a single machine. Though docker-compose is good enough for container deployment, it falls short when it comes to container orchestration. It cannot track the containers and maintain the stability of the infrastructure. That's where Kubernetes comes to our rescue.

Kubernetes can deploy containerised services not just on one machine but also on a cluster of any number of machines. It can deploy multiple instances of the same service across the cluster. Kubernetes keeps track of each of the deployed containers. And in case of crashes, it manages the scalability levels by automatically bringing up replacement containers without any manual intervention.

Let us architect the UMS (user management system) deployment with the help of Kubernetes.

Reference architecture

We have already decomposed our UMS into four microservices, namely: *AddService*, *FindService*, *SearchService* and *JournalService*. These use H2 relational databases for storage and Apache Kafka for asynchronous inter-service collaboration. Now, let's refactor the architecture to achieve the following:

1. Replace H2 with MySQL so that the data is saved persistently and shared across all the service instances.
2. Deploy Kafka cluster.
3. Deploy three instances of *AddService*.
4. Deploy six instances of *FindService* and *SearchService*.
5. Deploy one instance of *JournalService*.

Since we have only developed *AddService* so far, we will cover the first three goals in this part. Figure 1 gives our reference architecture.

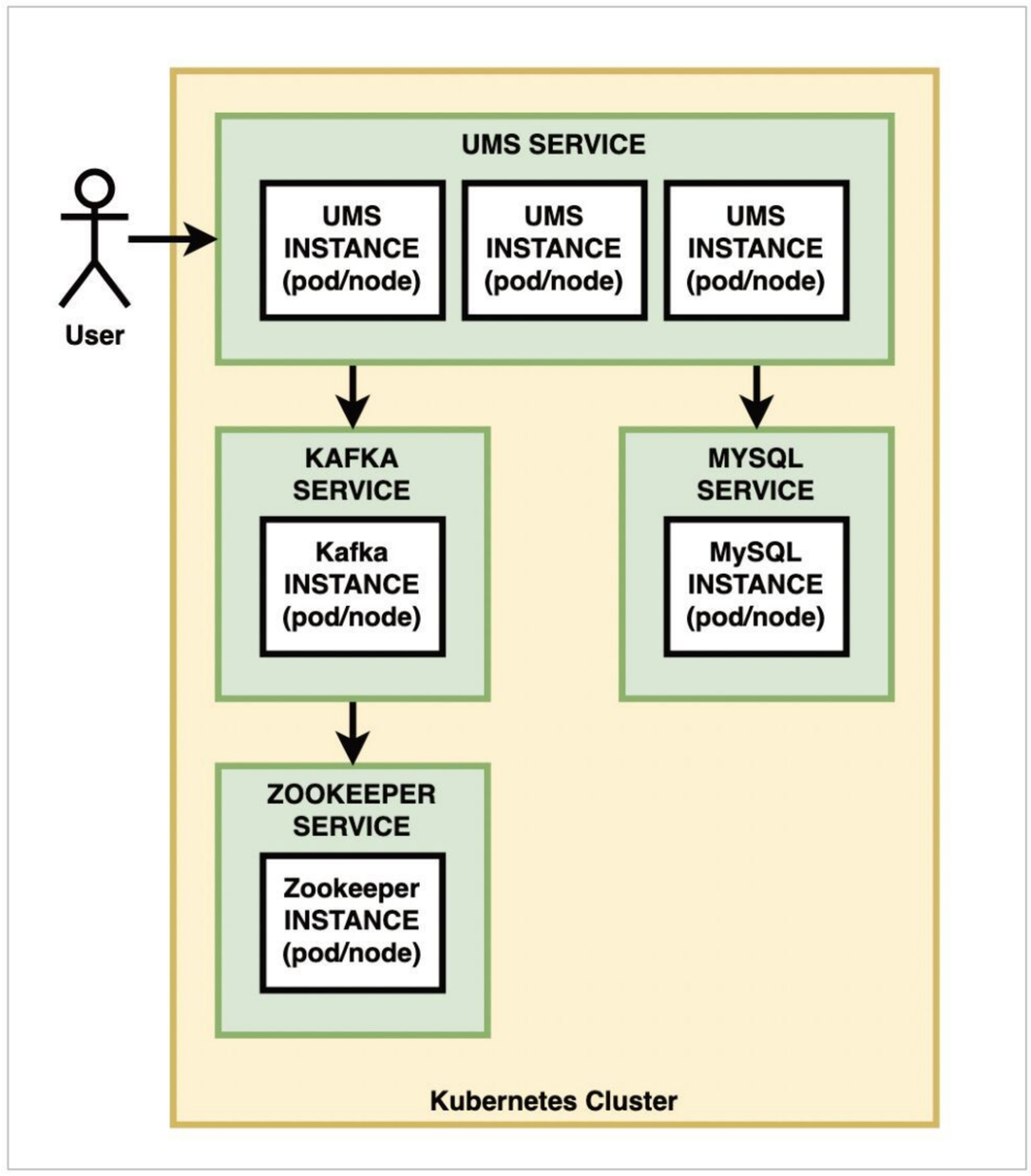


Figure 1: Reference architecture

Spring Boot and MySQL

The *AddService* is currently using the H2 database. As you know, H2 is an in-memory database engine. The data is lost once the engine is restarted. Such behaviour is not desired in the production. We need a database that is persistent. It can be an RDBMS or a NoSQL database like Mongo, etc. We chose MySQL for this illustration.

Since SpringBoot does not offer the MySQL connector out-of-the-box, we need to add it as a dependency in the *pom.xml* of the *AddService*.

```
<dependency>
  <groupId>mysql</groupId>
  <artifactId>mysql-connector-java</artifactId>
  <scope>runtime</scope>
</dependency>
```

We also need to update *application.properties* to specify the JDBC driver along with a connection string and access details for the MySQL database engine.

```
spring.datasource.driver-class-name=com.mysql.cj.jdbc.Driver
spring.datasource.url=jdbc:mysql://mysqldb:3306/glarimy?allow
PublicKeyRetrieval=true&useSSL=false
spring.datasource.username=root
spring.datasource.password=admin
```

Because of the above configuration, the repository of *AddService* attempts to connect to the *glarimy* database on a machine named *mysqldb* on port number 3306. We are recording the password in clear text in this configuration, only for simplicity. We will find a better way later!

A few other JPA-specific configurations may also be provided as needed. For example, the following will direct the Hibernate system to scan the code for JPA annotations and keep the schema on the database updated at the time of bootstrapping:

```
spring.jpa.hibernate.ddl-auto=update
spring.jpa.properties.hibernate.dialect=org.hibernate.
dialect.MySQL5Dialect
```

MySQL as a Docker container

Since the *AddService* depends on MySQL, we can update the existing *docker-compose.yml* for deploying and linking it:

```
  mysqldb:
    image: mysql:latest
    networks:
      - glarimy
    environment:
      - MYSQL_ROOT_PASSWORD=admin
      - MYSQL_DATABASE=glarimy
  volumes:
    - "mysql_data:/glarimy"
```

The above manifest pulls *mysql:latest* image from the Docker Hub and runs the container. The name of the container must be *mysqldb* as the *AddService* is looking for the database engine on a machine named *mysqldb*. Also, both must run on the same network to resolve the name. Since the *AddService* was configured to run on *glarimy* network (in the previous part), *mysqldb* is also configured to run on the same network.

The above configuration is also directing the container to create a database named *glarimy* since the *AddService* is configured to use that database.

However, this is still not sufficient. The MySQL container writes the data on to the file system that is mapped to the container. Once the container is restarted, the files are gone! That is not good for us. We want the data to be written on to the disk in such a way that it outlives the containers. In other words, we want to mount a volume so that the container uses only that mount point. The last line in the above configuration is meant for that.

The following is the resulting full manifest in *docker-compose.yml:*

```
version: "2"
networks:
  glarimy:
    driver: bridge
services:
  zookeeper:
    image: docker.io/bitnami/zookeeper:3.8
    ports:
      - "2181:2181"
    volumes:
      - "zookeeper_data:/glarimy"
    environment:
      - ALLOW_ANONYMOUS_LOGIN=yes
    networks:
      - glarimy
  kafka:
    image: docker.io/bitnami/kafka:3.1
    ports:
      - "9092:9092"
    volumes:
      - "kafka_data:/glarimy"
    environment:
      - KAFKA_CFG_ZOOKEEPER_CONNECT=zookeeper:2181
      - ALLOW_PLAINTEXT_LISTENER=yes
    networks:
      - glarimy
    depends_on:
      - zookeeper
  mysqldb:
    image: mysql:latest
    networks:
      - glarimy
    environment:
      - MYSQL_ROOT_PASSWORD=admin
      - MYSQL_DATABASE=glarimy
    volumes:
      - "mysql_data:/glarimy"
  ums:
    image: glarimy/ums-add-service
    networks:
      - glarimy
    depends_on:
      - zookeeper
      - mysqldb
volumes:
  zookeeper_data:
    driver: local
  kafka_data:
    driver: local
  mysql_data:
    driver: local
```

Run the following command to deploy the containers, like we did in the previous part:

```
$ docker-compose up
```

However, this is not our actual goal. The docker-compose can deploy multiple containers in one go, only on a single machine. It does not offer resilience and does not offer cluster deployment.

Kubernetes and Minikube

The production infrastructure of microservices consists of several machines forming a cluster. The containers are expected to be distributed fairly across the machines (aka nodes) in the cluster. New nodes may be added to the cluster and existing nodes may be removed from it at any time. Yet, the containers are expected to be rescheduled on the current set of nodes.

Kubernetes does take care of such an orchestration. It deploys the containers across the cluster and redistributes them whenever needed — all without any manual intervention.

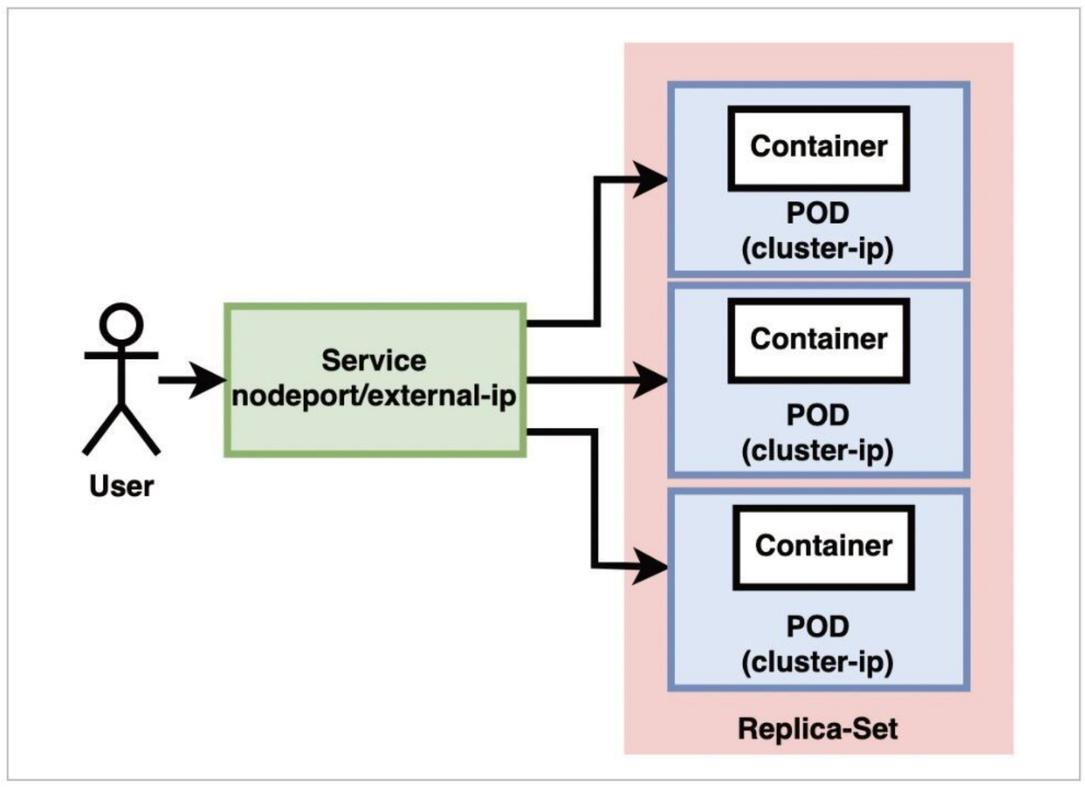


Figure 2: Kubernetes architecture

Figure 2 is a high-level presentation of Kubernetes architecture.

The Kubernetes cluster consists of one or more nodes, which may be physical or virtual machines. Each node can run several pods. A pod is a group of containers. Each pod gets an ephemeral IP address that is known as *cluster-ip* address. This address is local to the cluster and visible to all other pods across it. In other words, the pods within the cluster can reach out to each other using the *cluster-ip* address.

Normally, a pod consists of only one application container that runs a microservice. Besides this, a pod may also run several other infrastructure containers that take on tasks such as monitoring, logging, etc.

A deployment unit in Kubernetes consists of a set of such pods. This set is known as a replica-set. For example, you can create a deployment unit for *AddService* in such a way that three pods are scheduled in the cluster with each pod running an *AddService* container. If any of the pods crash for whatever reason, Kubernetes schedules another pod on the cluster in such a way that three pods of *AddService* are always running. Note that the pods of a replica-set do not necessarily run on a single node.

Though this is sufficient for the containers on different pods/nodes to collaborate with each other, it is very cumbersome for a pod to address another pod based on an ephemeral address. To solve this problem, we can create a front-end to each of the replica-sets. Such a front-end is called a service. Each service is exposed with an address that is not ephemeral. The address is called *node-port* if it is made visible only within the cluster or *external-ip* if exposed outside the cluster. A service can also be configured with a load balancer so that the incoming calls can be routed to the end-points (pods) in a fairly balanced manner.

This is Kubernetes in a nutshell. There are several tools available in the market to set up a Kubernetes cluster. Minikube is one such tool that helps in setting up single-node clusters. The instructions given below can be followed to set up the Minikube cluster on an Ubuntu machine that has Docker engine running.

Download Minikube distribution.

```
$ curl -LO https://storage.googleapis.com/minikube/releases/
latest/minikube-linux-amd64
```

Give the following command to install Minikube:

```
$ sudo install minikube-linux-amd64 /usr/local/bin/minikube
```

Create a group named *docker* and add the user:

```
$ sudo usermod -aG docker <user> && newgrp docker
```

Start the cluster:

```
$ minikube start
```

Create this handy alias:

```
$ alias kubectl="minikube kubectl --"
```

A single-node Kubernetes cluster is now up and running on the local machine.

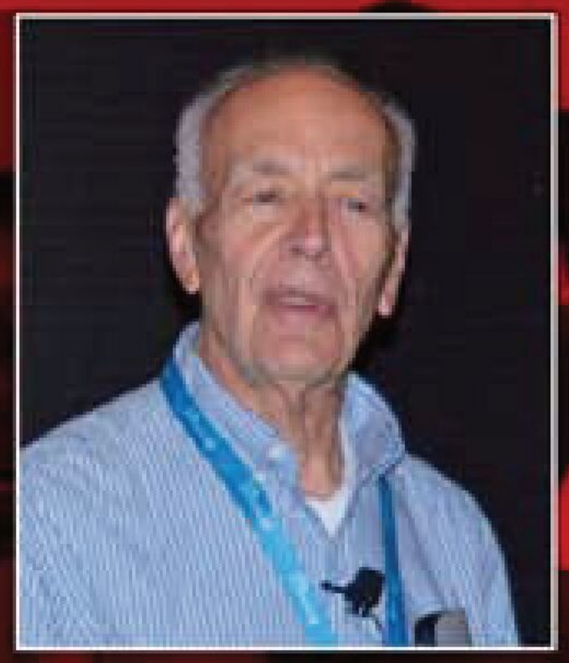
Deploying MySQL on Kubernetes

Let us deploy a replica-set consisting of just one pod of MySQL. The service is exposed by the name *mysqldb*. Other pods must use this name in order to access the database service. The port 3306 is exposed only within the cluster. We don't want any one from outside the cluster to log in to our database server. The deployment also mandates to create a schema by the name *glarimy* and to use a mounted volume.

```
apiVersion: v1
kind: Service
metadata:
  name: mysqldb
spec:
  ports:
  - port: 3306
  selector:
    app: mysqldb
  clusterIP: None
---
apiVersion: apps/v1
kind: Deployment
metadata:
  name: mysqldb
spec:
  selector:
    matchLabels:
      app: mysqldb
  strategy:
    type: Recreate
  template:
    metadata:
      labels:
        app: mysqldb
    spec:
      containers:
      - image: mysql:5.6
        name: mysqldb
        env:
        - name: MYSQL_ROOT_PASSWORD
          value: admin
```

ACCELERATING DEVELOPMENT AND DEPLOYMENT OF OPEN SOURCE















An EFYGROUP EVENT

tracks

- FOSS For Everyone (India contributes)
- CXO Summit (Business of Open Source)
- IT Infra (Security, Storage & Cloud)
- Developers' Meet
- DevOps
- Data & Bases: Open Source Rocks!

Asia's #1 Conference on Open Source

19th Edition

OPEN SOURCE INDIA

29-30 September 2022

NIMHANS CONVENTION CENTER, BENGALURU

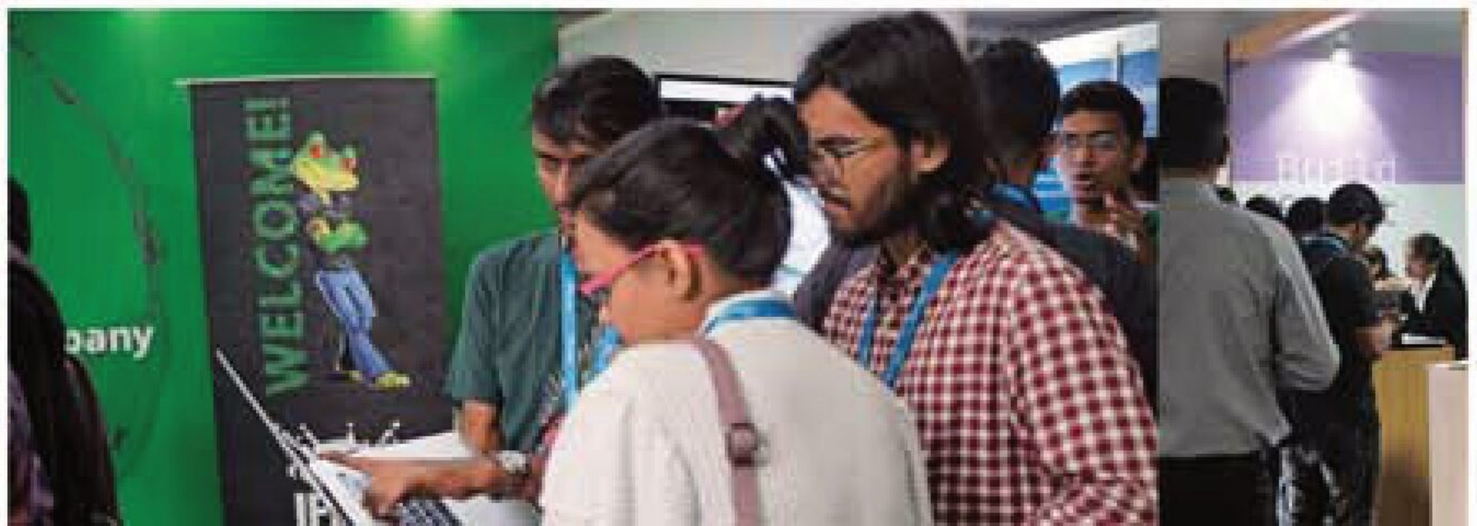
Register Now!

https://opensourceindia.in

www.OpenSourceIndia.in

For more details, call on +91 96677-43666 or write to info@opensourceindia.in







```
    - name: “MYSQL_DATABASE”
      value: “glarimy”
    ports:
    - containerPort: 3306
      name: mysqldb
    volumeMounts:
    - name: mysql-persistent-storage
      mountPath: /var/lib/mysql
  volumes:
      - name: mysql-persistent-storage
        persistentVolumeClaim:
          claimName: mysqldb
---
apiVersion: v1
kind: PersistentVolume
metadata:
  name: mysqldb
  labels:
    type: local
spec:
  storageClassName: manual
  capacity:
    storage: 20Gi
  accessModes:
    - ReadWriteOnce
  hostPath:
    path: “/mnt/data”
---
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: mysqldb
spec:
  storageClassName: manual
  accessModes:
    - ReadWriteOnce
  resources:
    requests:
      storage: 2Gi
```

Deploying Kafka on Kubernetes

Apache Kafka cluster requires Zookeeper for internal management. So we need to deploy both. Since Kafka and Zookeeper have their own discovery protocol, we expose them on NodePort and connect them.

```
apiVersion: v1
kind: Service
metadata:
  labels:
    app: zookeeper
  name: zookeeper
spec:
  type: NodePort
  ports:
  - name: zookeeper-port
    port: 2181
    nodePort: 30181
    targetPort: 2181
  selector:
    app: zookeeper
---
apiVersion: apps/v1
kind: Deployment
metadata:
  labels:
    app: zookeeper
  name: zookeeper
spec:
  selector:
    matchLabels:
      app: zookeeper
  replicas: 1
  template:
    metadata:
      labels:
        app: zookeeper
    spec:
      containers:
        - image: bitnami/zookeeper
          name: zookeeper
          ports:
          - containerPort: 2181
          env:
          - name: ZOOKEEPER_ID
            value: “1”
          - name: ZOOKEEPER_SERVER_1
            value: zookeeper
          - name: ALLOW_ANONYMOUS_LOGIN
            value: “yes”
---
apiVersion: v1
kind: Service
metadata:
  labels:
    app: kafka
  name: kafka
spec:
  type: NodePort
  ports:
  - name: kafka-port
    port: 9092
    nodePort: 30092
```

```
    targetPort: 9092
  selector:
    app: kafka
---
apiVersion: apps/v1
kind: Deployment
metadata:
  labels:
    app: kafka
  name: kafka
spec:
  selector:
    matchLabels:
      app: kafka
  replicas: 1
  template:
    metadata:
      labels:
        app: kafka
    spec:
      containers:
        - name: kafka
          image: bitnami/kafka
          ports:
          - containerPort: 9092
env:
- name: KAFKA_BROKER_ID
  value: “1”
- name: MY_MINIKUBE_IP
  valueFrom:
    fieldRef:
      fieldPath: status.hostIP
- name: KAFKA_ZOOKEEPER_CONNECT
  value: “$(MY_MINIKUBE_IP):30181”
- name: KAFKA_LISTENERS
  value: “PLAINTEXT://:9092”
- name: MY_POD_IP
  valueFrom:
    fieldRef:
      fieldPath: status.podIP
- name: KAFKA_ADVERTISED_LISTENERS
  value: “PLAINTEXT://$(MY_POD_IP):9092”
- name: ALLOW_PLAINTEXT_LISTENER
  value: “yes”
```

Deploying *AddService* on Kubernetes

And, finally, we want to deploy three instances of *AddService* and expose them to the outside world through a load balancer with an *external-ip*.

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: ums-add-service
  labels:
    app: ums-add-service
spec:
  replicas: 3
  selector:
    matchLabels:
      app: ums-add-service
  template:
    metadata:
      labels:
        app: ums-add-service
    spec:
      containers:
        - name: ums-add-service
          image: glarimy/ums-add-service
          ports:
            - containerPort: 8080
---
apiVersion: v1
kind: Service
metadata:
  name: ums-add-service
  labels:
    name: ums-add-service
spec:
  type: LoadBalancer
  ports:
  - port: 8080
  selector:
    app: ums-add-service
```

This whole configuration can be written in one single manifest file and deployed with one single command:

```
$ kubectl create -f <manifest-file-name>.yml
```

In order to run the load balancer, the following command also needs to run on a separate terminal:

```
$ minikube tunnel
```

You can check the deployed service using the following command:

```
$ kubectl get services
```

It gives an output that looks like Figure 3. It lists the services, their addresses, etc.

The following command lists the deployments, which

```
krishnamohan_koyya@instance-1:~$ kubectl get services
NAME                TYPE           CLUSTER-IP       EXTERNAL-IP    PORT(S)          AGE
kafka               NodePort       10.108.141.251   <none>         9092:30092/TCP   38m
kubernetes          ClusterIP      10.96.0.1        <none>         443/TCP          11h
mysqldb             ClusterIP      None             <none>         3306/TCP         38m
ums-add-service     LoadBalancer   10.96.38.181     10.96.38.181   8080:31248/TCP   38m
zookeeper           NodePort       10.107.137.176   <none>         2181:30181/TCP   38m
```

Figure 3: Kubernetes services

```
krishnamohan_koyya@instance-1:~$ kubectl get deployments
NAME                READY   UP-TO-DATE   AVAILABLE   AGE
kafka               1/1     1            1           39m
mysqldb             1/1     1            1           39m
ums-add-service     3/3     3            3           39m
zookeeper           1/1     1            1           39m
```

Figure 4: Kubernetes deployment

```
krishnamohan_koyya@instance-1:~$ kubectl get pods
NAME                                READY   STATUS    RESTARTS   AGE
kafka-588777fff6-86s24              1/1     Running   0          39m
mysqldb-8454bb5bdd-ph2xs            1/1     Running   0          39m
ums-add-service-7cb4dc9f9b-72xtq    1/1     Running   0          39m
ums-add-service-7cb4dc9f9b-cjx22    1/1     Running   0          39m
ums-add-service-7cb4dc9f9b-thgkn    1/1     Running   0          39m
zookeeper-6bb6f5f5-sl5mq            1/1     Running   0          39m
```

Figure 5: Pods

show the number of pods of each deployment that is running:

```
$ kubectl get deployments
```

The output looks like Figure 4.

And, finally, to see the real pods that run the containers, use the following command:

```
$ kubectl get pods
```

Figure 5 shows that there are three pods running for *AddService*, and one pod running for Zookeeper, Kafka and MySQL each.

Since the *AddService* is exposed with an *external-ip*, it can be accessed using the following command:

```
$ curl -X POST -H 'Content-Type: application/json'
-i http://<service-external-ip>:8080/user --data
'{"name":"Krishna Mohan", "phone":9731423166}'
```

Why is this reference architecture?

Irrespective of the nature of the application, number of microservices, platforms on which they are developed, services like databases, brokers, etc, the overall architecture remains the same like what has been described here.

The development architecture focuses on service decomposition, platform selection, framework selection, design of API, repositories, etc. This part was addressed using our understanding of domain driven design, object-oriented patterns, frameworks like SpringBoot, Flask, Express, etc.

The deployment architecture focuses on number of machines, nodes, replica-sets, pods, address-mechanisms, volumes, etc. This part is addressed using our understanding of container technology and Kubernetes. We will dwell into design patterns associated exclusively with microservices like gateways, circuit breakers, registries, etc, in future. The good thing is that Kubernetes and other such tools implement many of these patterns out-of-the-box.

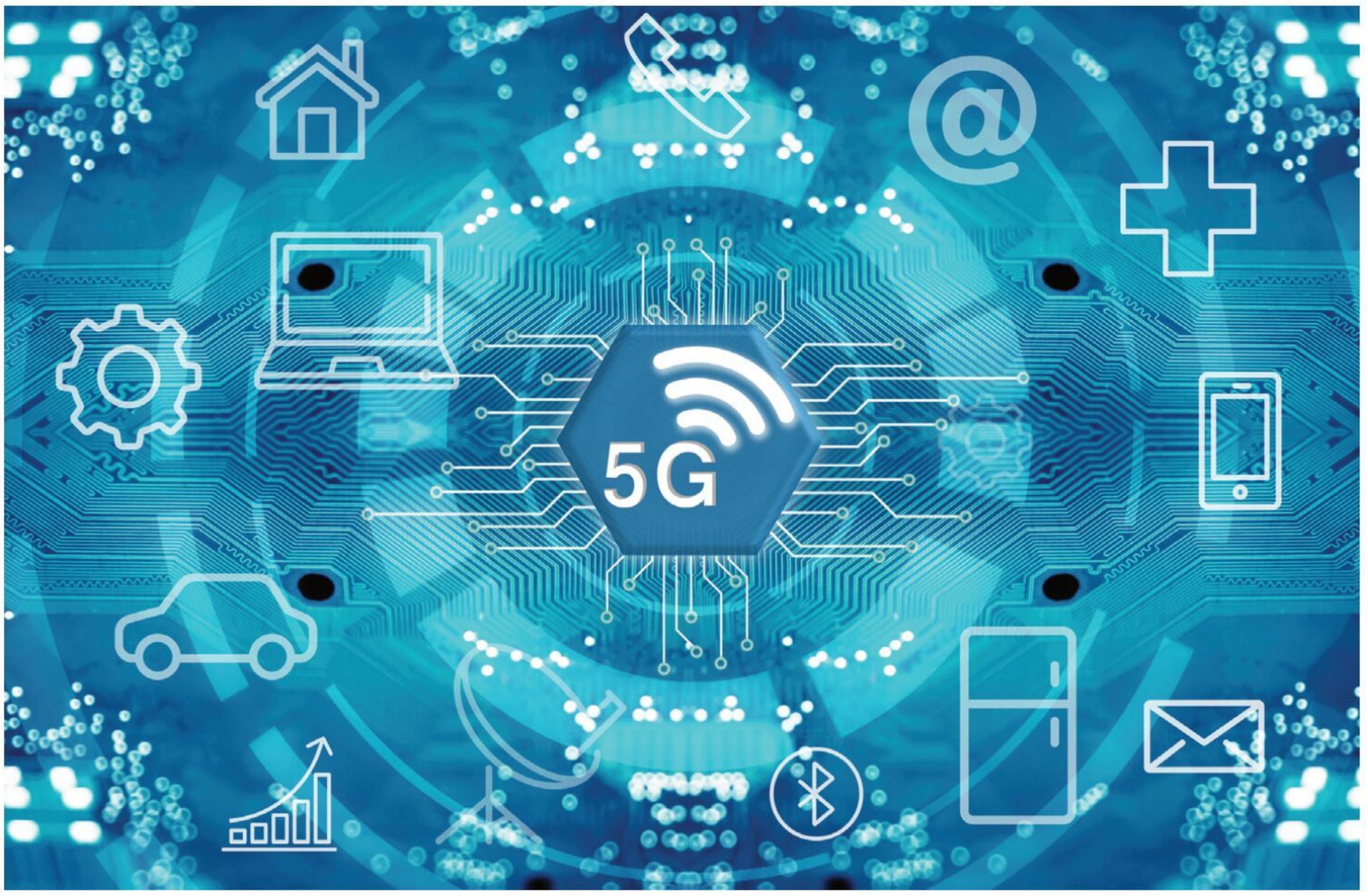
Before going that far, we will develop the *FindService*, *SearchService* and *JournalService* on Python and Node platforms in the next parts of this series of articles so that we take UMS to a conclusion.. END

By: Krishna Mohan Koyya

The author is the founder and the principal technology consultant at Glarimy Technology Services, Bengaluru. He has delivered more than 500 programs to corporate clients on design and architecture in the last 12 years. Prior to that, he held various positions in the industry and academia for a decade.

Why Telecom Service Providers Should Opt for Open RAN

Technology is evolving at a much higher pace than anyone could have envisaged a few decades ago. The telecommunications industry is no different in this respect. With 5G technology all set to be available in India, read on to understand what Open RAN brings to the table.



Many, many years ago we used analogue phones at home and the office. Then came mobile networks with 2G technology, which introduced digital voice calls along with text messaging and other data services, although speed was a major issue. This was followed by 3G technology, which ushered in smartphones and a significant improvement in the speed of the network. 4G brought along with it a very high speed network — anywhere between 2Mbps and 10Mbps — and allowed multiple devices to connect to a single network. Now, as we roll out 5G technology, we need to get ready for really high speeds – as much as 1100Mbps to 200Mbps — for downloading data. 5G also has the lowest latency rate of about one millisecond, which is far lower than what 4G has. This will improve real-time communications tremendously.

What does 5G offer?

- ***Enhanced mobile broadband:*** With 5G, augmented reality applications that require a good Internet speed with a low latency will be a lot more feasible. 5G will help to maintain the uniformity of the connection, ensuring that your experience is seamless.
- ***Improved mission-critical services:*** Self-driving cars require a real-time connection with servers, and 5G can help with that (a real-time connection helps in making a better decision). It can be of great use in the robotics sector too. In the healthcare sector, 5G can make it easier to conduct medical procedures remotely due to its ultra-low latency and strong security.
- ***Better connectivity of Internet of Things (IoT) devices:*** Many IoT devices demand good connectivity in order to function properly and provide users with a hassle-free experience. 5G will improve their overall coverage at a very affordable cost.

How does the mobile network work?

Mobile devices are a type of IoT device. These devices are connected to various towers known as base stations. A group of such cell towers forms a radio access network, which is connected to the core network. This core network is responsible for client profiling. It provides you the access for various services such as

subscriber trunk dialling (STD) calls and data services. Signals from the towers are sent to the core network, and the latter is responsible for the execution of the final service. This can be any kind of service, even a simple Internet connection. So the core network acts as a gateway between the radio access network and the final services (Figure 1).

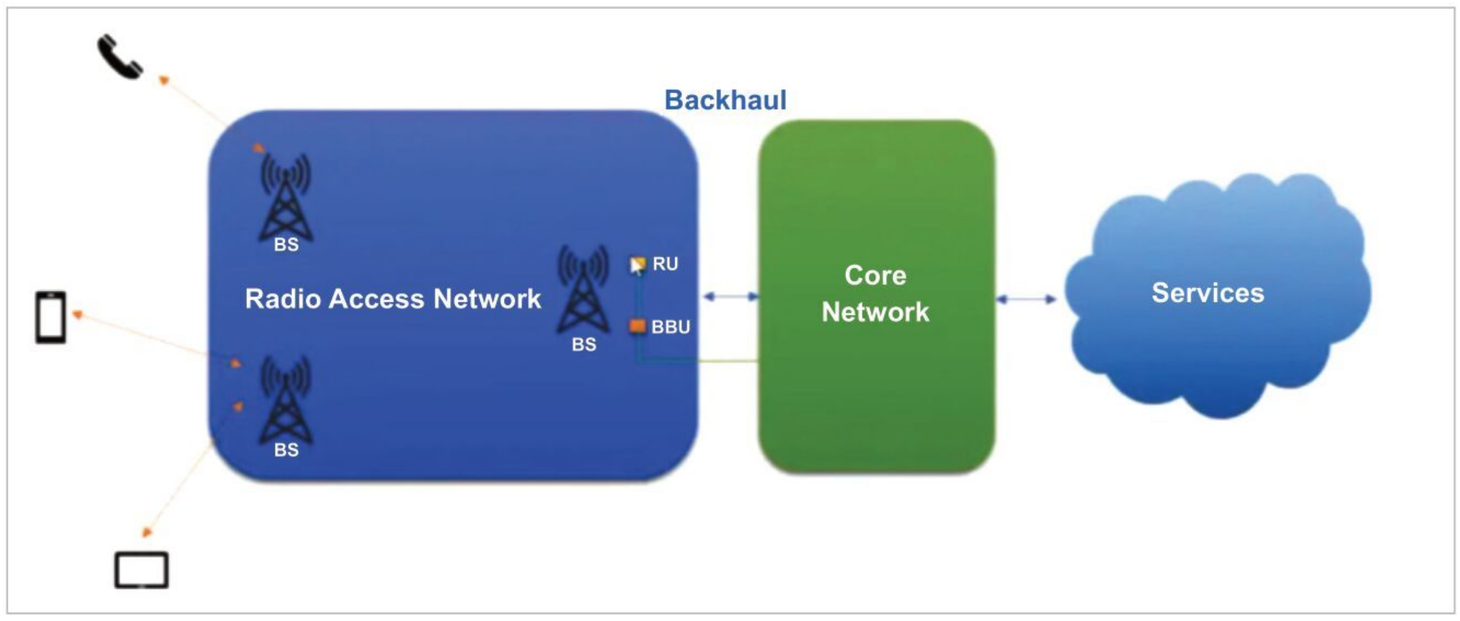


Figure 1: Mobile network

Traditional RAN (radio access network)

In a traditional RAN, the base station consists of mainly two units. The first is the radio unit (RU) or the signal transmitting unit and the second is the baseband unit (BBU), which is responsible for calculating the signals. The RU receives the signals and sends them to the BBU. These are then sent to the core network. A single vendor provides all the units and the connection between them is deployed only by that vendor. This is a very tightly coupled system, which makes it hard to change even a single component. The software is very tightly integrated inside the hardware, which presents a challenge to change it, making it difficult to upgrade the infrastructure. This increases the dependency on a single vendor, and one has to shift to a different vendor to change the entire hardware and the software, which is never convenient.

So, what is Open RAN?

Open RAN is a set of specifications for desegregating software and hardware. It allows hardware and software from different vendors to work together. Servers can now independently implement virtualisation due to this desegregation. Open RAN even makes cloud solutions possible. It lets one virtualised baseband unit support multiple radios instead of needing a proprietary or a physical unit at every cell site. This makes the infrastructure flexible and cost effective.

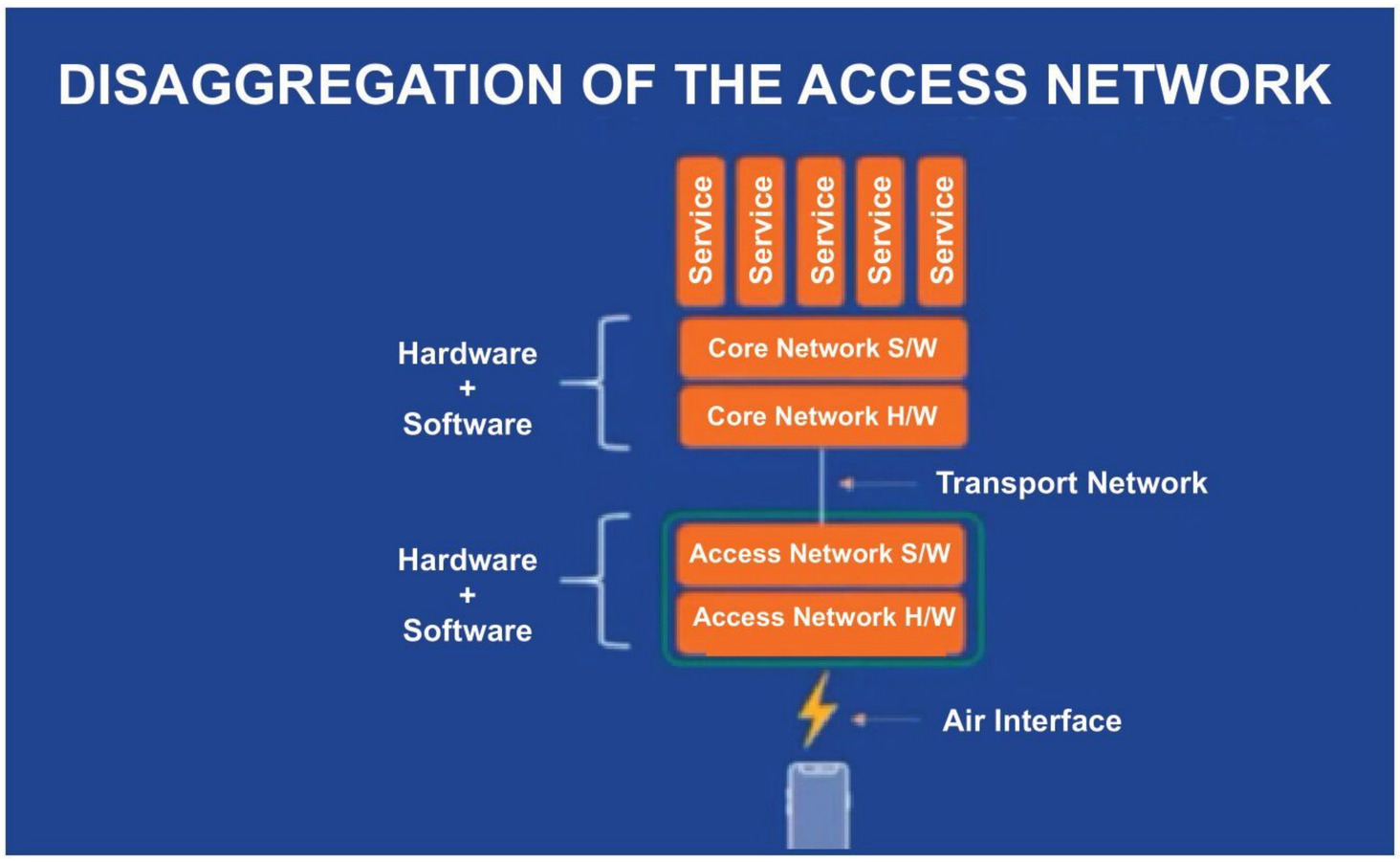


Figure 2: Typical mobile architecture

But why should one go for Open RAN (O-RAN)?

- ***Hardware and software disaggregation:*** There are no direct restrictions on the hardware and the software.
- ***Intelligence and automation:*** It makes management and orchestration more feasible since everything is quite open and there are standard rules to do things. It uses artificial intelligence and machine learning capabilities.
- ***Open internal RAN interfaces:*** The O-RAN lower layer basically splits a single unit into multiple units, mobilises them and makes it a multi-vendor solution.
- ***Rapid innovation:*** O-RAN has improved the innovation pace significantly due to the increased competition among vendors themselves. New companies are now coming up with their own components which tend to work a lot better.

Architecture of a typical mobile network

As can be seen in Figure 2, there is a core network and an access network, which is connected by the transport network (also known as the backhaul network). Then there is an air interface that connects a user with the service provider's network. There are various other services that run on top of the core network. The core network and the access network contain both the hardware and software, which can be seen on the left in Figure 2.

EFY GROUP
YOURS SINCE 1969

Stay **Connected.**
Stay **Informed.**
Stay **Ahead.**



SUBSCRIBE AND **SAVE**

ORDER FORM

PRINT MAGAZINE	1 YEAR (12 copies each)	3 YEARS (36 copies each)	5 YEARS (60 copies each)
		WITHIN INDIA (IN RUPEES)	
Electronics For You (Rs 100/copy)	840 ☐	2150 ☐	3000 ☐
		SAARC COUNTRIES (IN US$ BY AIR)	
	50 ☐	135 ☐	NA
		OTHER COUNTRIES (IN US$ BY AIR)	
	100 ☐	270 ☐	NA

e-magazine subscriptions within India are available at half the rates mentioned here.
Overseas rates for each e-magazine in US$: 1 year: $12; 3 years: $33; 5 years: $50 only

PRINT MAGAZINE SUBSCRIBERS GET:
- Free e-magazine every month
- Free delivery of print magazine by post
- And much more (check: subscribe@efy.in)
- For delivery by courier, please add Rs 50 for each copy

To subscribe online, visit https://payment.efyindia.com | OR | **SCAN THIS CODE**

Name_____Organisation_____Mailing Address_____

_____City_____

Pin Code_____State_____Phone No._____Email_____

Subscription No. (for existing subscribers only)_____. **I would like to subscribe to the above (✓)marked magazine(s) starting with the next issue.** Please find enclosed a sum of Rs_____by DD/MO/crossed cheque bearing the No._____dt._____in favour of EFY Enterprises Pvt Ltd, payable at Delhi.

Please mark one (nearest) relating to your subscription: ☐ **Indian Company** ☐ **MNC** ☐ **R&D organisation** ☐ **Engineering institute** ☐ **College/School** ☐ **Any other (specify):**_____

Send this filled-in form or copy to : EFY Enterprises Pvt Ltd, D-87/1 Okhla Industrial Area, Phase 1, New Delhi 110 020 | Ph: 011-40596600 | e-mail: support@efy.in

Terms:- # These rates are applicable for new subscribers as well as renewal by existing subscribers # Please allow 4-6 weeks for processing of your subscription.
Please include your pincode for prompt delivery of your copy.

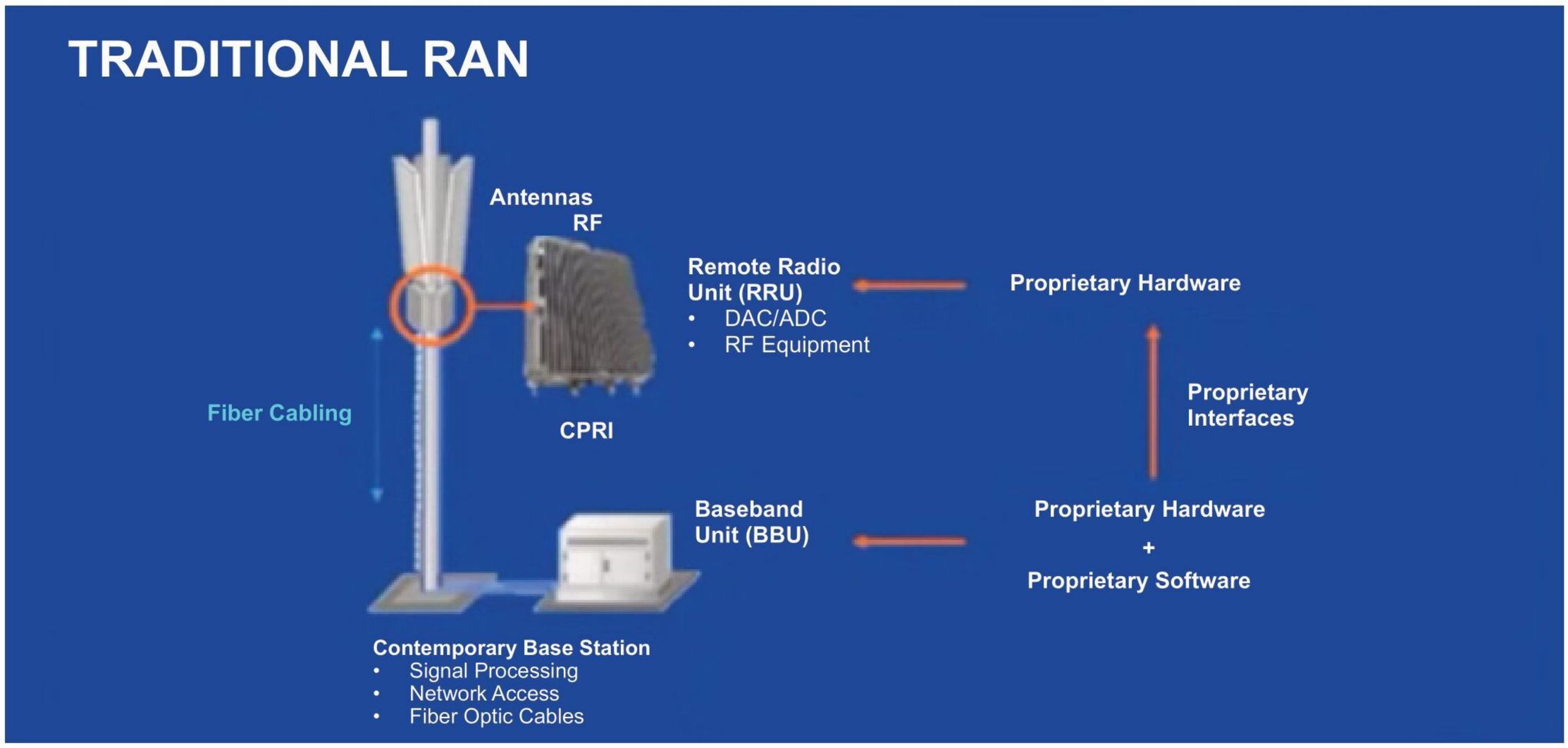


Figure 3: Traditional RAN

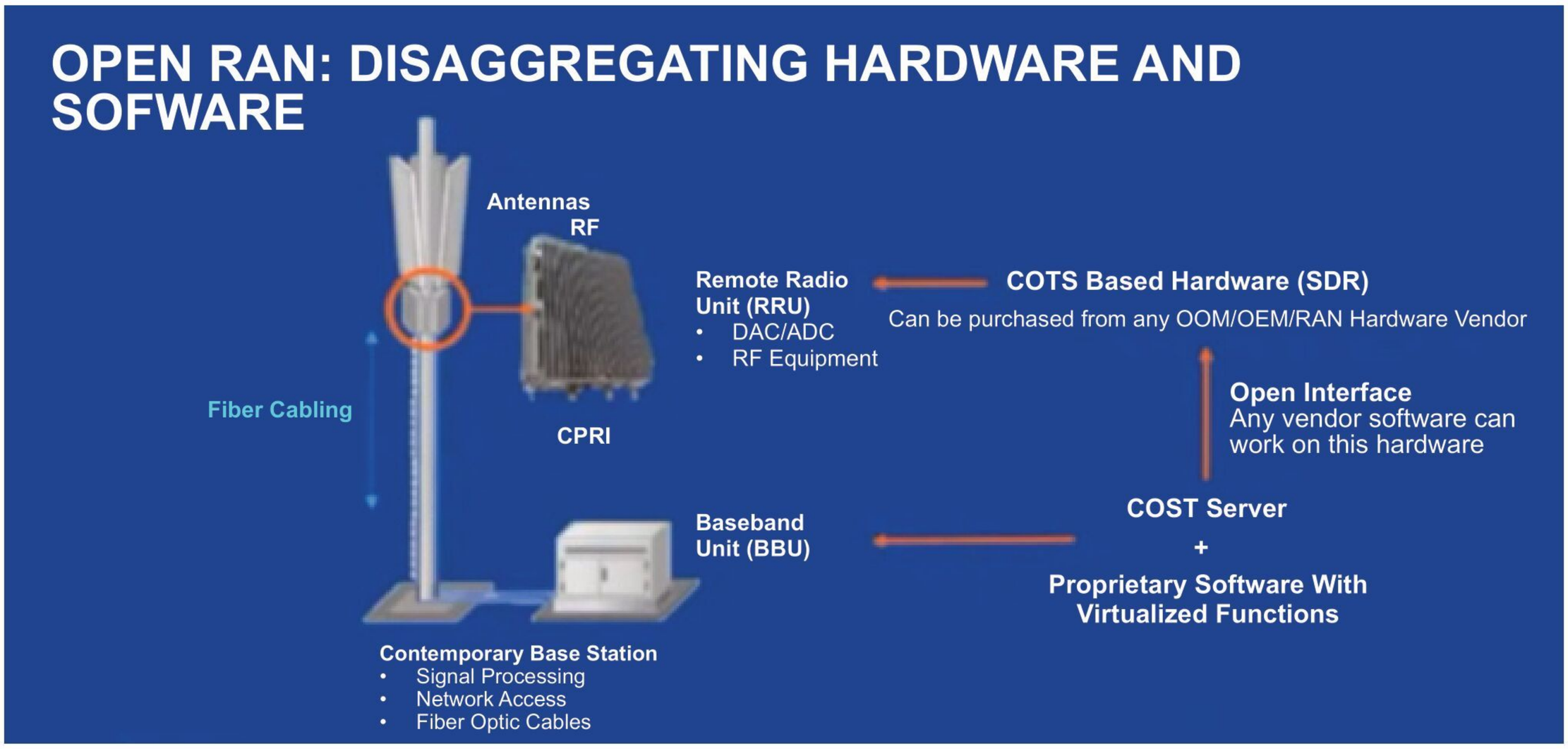


Figure 4: Open RAN architecture

To have a more detailed understanding of this architecture, let us understand a few things about previous generation network architectures first. All the electronics was in one or more cabinets at the bottom of a mast (tower) when 2G and 3G were in use. The RF cables would run to the antennas on top of the mast. The RF signal continuation in the RF cable was a big problem because of the congestion that used to occur. 3G and 4G allowed the industry to take a different approach that worked better. Instead of sending RF signals from the cabinet to the antennas, they were moved closer to the antennas (as can be seen in Figure 3). This ensured that there was a very minimal loss of signals.

This new piece of electronics is known as remote radio unit (RRU) or remote radio head (RRH) — at the bottom is the baseband unit. A cabinet consists of both the hardware and the software. RRU is always proprietary hardware, whereas BBU is proprietary software along with the interfaces between BBU and RRU.

Understanding the Open RAN architecture

In Open RAN, the RRU is COTS based hardware that can be

Open RAN components

- **Baseband unit:** This transports a baseband frequency, usually from a remote radio unit, to which it may be tied through optical fibre.
- **Remote radio unit:** RRU is the distributed and integrated frequency unit that connects to an operator's network with the user equipment (like cell phone and mobile devices).
- **Radio frequency:** RF band is a frequency band that is used for transmissions and broadcasting.
- **Common public radio interface (CPRI):** CPRI is a popular standard for transporting baseband signals to radio units. It is an interface between radio equipment control (REC) and radio equipment (RE).

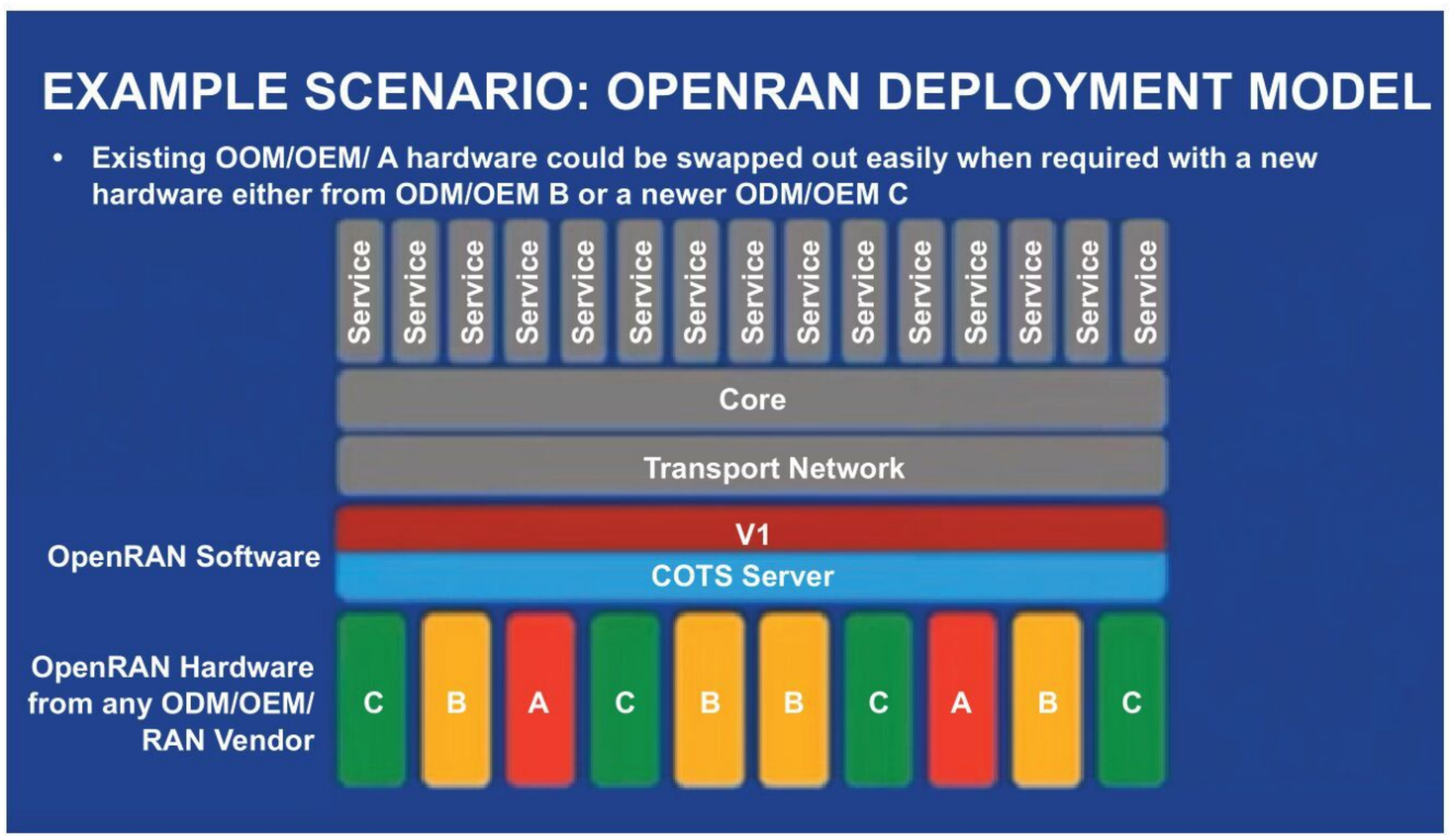


Figure 5: Open RAN deployment model

purchased from any original design manufacturer (ODM), original equipment manufacturer (OEM) or RAN hardware vendor. The BBU is proprietary software with virtualised functions running on a COTS server. It is important to note that the interface between BBU and RRU is an open interface, as can be seen in Figure 4. This will allow any vendor's software to work with a different vendor's hardware.

The example shown in Figure 5 will help to understand this better. Let us assume that a service provider has deployed O-RAN software from vendor V1 that runs on a COTS server, while the hardware is being used from an OEM/ ODM vendor A. If the service provider now decides to take components from a new hardware vendor, vendor B or vendor C (maybe because of better features), this hardware will also be compatible with the O-RAN software from vendor V1. Now, let us say that the service provider is unhappy with the software vendor V1 for some reason. In that case, he can just replace vendor V1 with a new vendor V2. Because of the open interface, the hardware from vendor A, B or C will work well with the software from vendor V2 too.

What Open RAN enables

- ***No vendor lock-in:*** With the fully interoperable and desegregated RAN, the service provider is able to choose from the RAN component manufacturers. This results in a mix of vendors as opposed to just one or two.
- ***Best-of-breed components***: Open RAN brings more flexibility in choosing components from the best vendors. It promotes supply chain diversification as well as price competition among vendors.
- ***Reduced costs:*** Open RAN helps to reduce costs by enabling rapid innovation, vendor diversification and higher automation.

Challenges with Open RAN

- ***Security:*** Network security is one of the major challenges as different vendors follow different security protocols.
- ***Openness:*** Since it is open source, it is susceptible to various kinds of threats.

Mobile network operators are benefiting tremendously with Open RAN standards. New features on the anvil will make O-RAN even more economical, while improving hardware and software integration. END

References

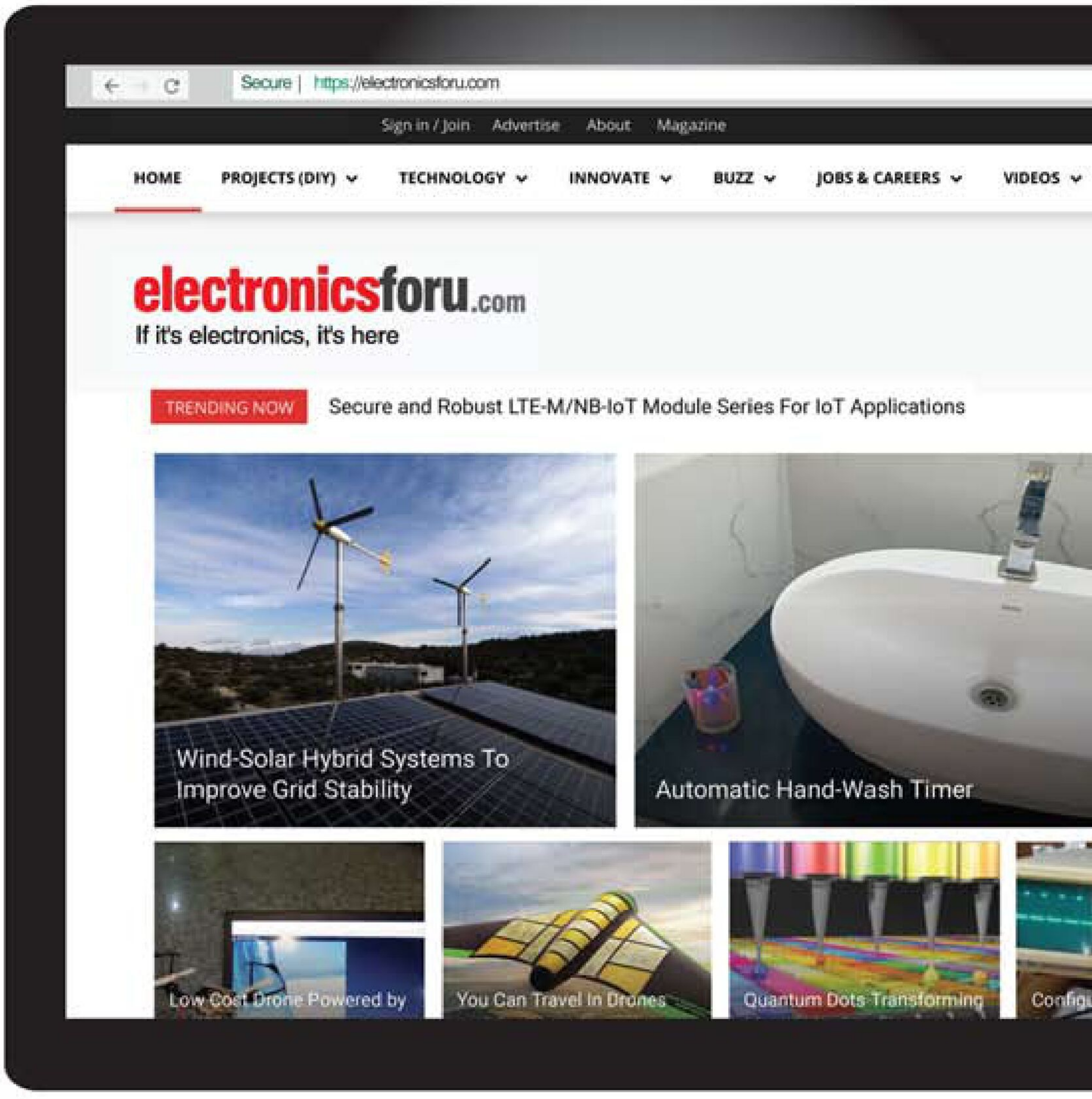
[1] *https://www.youtube.com/user/ParallelWireless*
[2] *https://www.parallelwireless.com/blog/dispelling-the-fear-uncertainty-and-doubt-about-open-ran-integration/*

By: Binitha MT, Subhankar Adak, and Krutika Dhanakshirur

Binitha MT is a software senior engineer at Dell Technologies.
Subhankar Adak is a senior software engineer at Dell EMC.
Krutika Dhanakshirur is test engineer 2 at Dell EMC. This article is based on a tech talk they gave at OSI 2021.

Transcribed and curated by Laveesh Kocher

Your favourite website has



electronicsforu.com

THANKS TO YOU—OUR ONLINE NETWORK IS

FACTS & FIGURES

- 4 websites (two more coming soon)
- Five major Facebook communities
- Seven major LinkedIN groups & pages
- Million-plus active users (monthly)
- Million-plus reach through Facebook
- Fifty-thousand-plus industry connections through LinkedIN

READERS

- You can access all content for FREE
- You can subscribe to newsletters for FREE --on most websites
- Register on our websites to get free invites to technical webinars and seminars

EXPERTS

- Experts who want to share their knowledge through articles, DIY Projects, etc are welcome
- We also welcome experts who want to share their knowledge through webinars or seminars
- You can contact us at editop@efy.in

fast growing peers now...





The Latest in IOT.
A platform for enablers, creators and providers of IOT solutions.



India. Electronics. Directory.
Enabling commerce between buyers & sellers of electronics in India



Business. Electronics. India.
Everything you want to know about India's electronics industry.

Amazing DIY Projects. Latest Tech trends.

The hang-out for electronics enthusiasts.

AMONGST THE WORLD'S TOP 5 AND GROWING!

INDUSTRY

- You can advertise for as little as US$ 100 per month
- Special combo offers for advertisers in our print publications
- We've now enabled flexible CPM-based advertising
- You can advertise on the platform of your choice (based on your target audience)
- We invite press releases at efy-edit-team@efy.in
- Press releases are published free of cost, subject to discretion of the editorial team

RESPONSE GUARANTEED SOLUTIONS

We now also act as marketing partners for our clients and drive entire marketing for them, where we charge them on basis of results and not efforts!

CONTACT US: Shrikant Rao • growmybiz@efy.in • +91-98111 55335

NFTs – Not Frivolous Tech

A web search for 'non-fungible token' or NFT throws back endless news about limited-edition virtual sneakers, digital art, memes, tweets, and trading cards being sold for unrealistic prices. But NFT means much more. It is serious tech with extremely great potential in the digital and physical worlds. Here is what you need to know about NFTs and their applications.

A non-fungible token (NFT) of Chris Torres' *Nyan Cat*, a video of a flying cat with a pop tart body leaving behind a rainbow trail, sold for $580,000, while *Everyday: The First 5000 Days*, a collage of 5000 digital images by Mike Winkelmann, sold for $69 million. A 21-year-old Dutch student recently put up his soul for sale as an NFT, and at the time of writing this article its price had crossed $3.4 million. Closer home, four Thar based NFTs auctioned by Mahindra & Mahindra in March 2022 sold for a sum of ₹26,00,000!

Seeing these examples, one may be fooled into thinking non-fungible tokens or NFTs are all about people burning cash in collectibles and memorabilia. However, NFT as a technology means much more, and can potentially be used for many more real applications that can affect our daily lives, including medical record maintenance, improving supply chain efficiency, industrial design prototyping, and protecting intellectual property (IP). It can be a very effective way of digitising real-world objects and processes.

Let us take a quick look at what NFTs are, a deeper look at their role in the art, collectibles and memorabilia space, and about other uses they can be applied to.



Figure 1: Mahindra's Thar based NFT - Standing Tall (Source: auto.mahindra.com)

What is an NFT?

A non-fungible token or NFT is a unit of data stored on a digital ledger, such as the blockchain. It is also a crypto-token like digital currency, but here the unit of data is associated with a particular digital or physical asset—say, a file, a work of art, a degree certificate. The data is used to give important information about the asset, such as the creator's signature, current ownership, history, and terms of use or license.

Let us take a look at its characteristics.

It is non-fungible: Cryptocurrencies like Bitcoin or Ether are fungible tokens. That is, if you take two Bitcoins, they represent the exact same value, which makes them interchangeable or fungible. On the other hand, NFTs represent assets that are rare and unique, and not replaceable by one another, which makes them non-fungible.

It is tradeable: You can sell or trade NFTs in digital marketplaces, such as OpenSea, Foundation, Sorare, or Decentraland, using supported digital currencies stored in digital wallets.

It is immutable: Once an NFT has been encoded using blockchain technology, you cannot alter the

metadata, ownership information, or history. A blockchain is a decentralised mechanism, open to scrutiny. This makes an NFT a trustable certificate of uniqueness and ownership.

It is verifiable: Not only is an NFT immutable, it is also verifiable. That is what makes it so special. At any point of time, you can quickly and easily verify the authenticity and ownership history of an asset using a block explorer for that specific blockchain, such as Etherscan for Ethereum and PolyScan for Polygon.

It is interoperable: Technically, NFTs can be traded across online platforms, wallets, and marketplaces because of the open nature of blockchain standards.

It can use any blockchain: Today, most NFTs are programmed using Ethereum standards, though technically they can be based on any blockchain like Algorand, Tezos, or Polkadot.

This ability to record and validate digital creations, and to digitise real-world objects, makes NFTs a force to reckon with in the emerging metaverse that aims to integrate real and online experiences, using extended reality tools and the Web3 paradigms of decentralisation and token based economics.

2014 to 2021—a quick recap

The first-known NFT was a small video clip called Quantum, which was registered on the Namecoin blockchain in 2014 and sold for $4. It was a small experiment by techie Anil Dash and artist Kevin McCoy, who paired at an event for sparking new ideas. The duo termed it as monetised graphics.

"McCoy used a blockchain called Namecoin to register a video clip that his wife had previously made, and I bought it with the four bucks in my wallet. We didn't patent the basic idea, but for a few years McCoy tried to popularise it, with limited success. Our first demo might just have been ahead of its time. The system verifiably demonstrated that day in 2014 is now making headlines in the form of non-fungible tokens, or NFTs, and it's the basis of a billion-dollar market," wrote Dash in *The Atlantic*.

In October 2015, the first NFT project Etheria was launched at Ethereum's first developer conference. They released 457 purchasable, tradeable hexagonal tiles. Most remained unsold.

In 2017, the online game CryptoKitties started selling tradeable cat NFTs. It became so popular that it temporarily slowed down the Ethereum network! "CryptoKitties has become so popular that it's taking up a significant amount of available space for transactions on the Ethereum platform," says Garrick Hileman of the University of Cambridge in a 2017 BBC News report. "Some people are concerned that a frivolous game is now going to be crowding out more serious, significant-seeming business uses." However, you will not find the term NFT in any of the news reports of that time!

The term NFT became popular only after the advent of the Ethereum based ERC 721 (Ethereum Request for Comments 721) standard in 2018. ERC 721 implements an application-programming interface (API) for tokens within smart contracts. It ensures the uniqueness or non-fungibility of tokens and adds more functionalities to know the owner of a specific token, transfer tokens from one account to another, and so on.

NFTs slowly picked up steam, and several NFTs and in-game assets were sold in platforms like Decentraland and Gods Unchained. In the first three months of 2021, more than $200 million was spent on NFTs. NFTs caught the attention of the art world, when a digital art by Mike Winkelmann or Beeple sold at a Christie's auction for more than $69 million in March 2021. In the same month, Ethereum relaunched the unsold tiles of 2015, and they sold within 24 hours for a total of $1.4 million. In August 2021, Jack Dorsey, co-founder of Twitter, sold an NFT of his first

Process of selling or buying NFTs

- Set up a digital wallet, compatible with the marketplace you have chosen. Metamask, Trust Wallet, Coinbase Wallet, and AlphaWallet are some of the popular ones.
- Buy cryptocurrency using your credit card on platforms like PayPal or Coinbase. Expect the exchange to charge a percentage of your transaction as fee.
- Load your wallet with the cryptocurrency.
- Connect the wallet to an NFT marketplace, like OpenSea, Foundation, Sorare, SuperRare, Nifty Gateway, Rarible, or Decentraland.
- Mint/list your NFT, or start bidding for one. Expect to be charged a 'gas fee'!

What is an NFT drop?

- It refers to the release of an NFT or a collection of NFTs on a digital marketplace
- It specifies the exact date and time
- Will often have White Lists to get in early
- To get into the list, you have to be active on social media and garner the attention of the creator or organiser of the drop
- It is worth getting in because you will get NFTs cheaper at the drop
- There will be purchase limits
- And, there might also be a limit on how many NFTs can be minted



"As the crypto and blockchain space evolves further we should be seeing decentralised management of copyrights, which then could be linked or encoded with the NFTs instead of just residing as a metadata."

—Jothi Krishnan Menon, co-founder & CEO, RtistiQ

tweet—'just setting up twttr'—for $2.9 million. In December 2021, digital artist Pak's *The Merge* fetched $91.8 million on Nifty Gateway!

Auction houses started tying up with digital artists, celebrities started making NFTs of their milestones and memories, sports leagues started launching NFT based trading cards embedded with top shots and historic moments, and companies ranging from fashion to food started launching themed NFT collectibles.

NFTs became so popular in 2021 that many experts found it a bit crazy! Some connected it to the boom in cryptocurrencies at that time. The price of cryptocurrencies skyrocketed in the first quarter of 2021. The price of Ether skyrocketed more than 1000% year on year, and that of Bitcoin rose more than 600%. This meant major returns for early crypto-buyers, which they started investing in digital assets. The Beeple, for example, was bought by Vignesh Sundaresan *aka* Metakovan, and Anand Venkateswaran *aka* Twobadour, two South Indian crypto-billionaires based in Singapore.

A few analysts even attributed the NFT trend to the Boredom Markets Hypothesis. The theory, proposed by lawyer and Bloomberg columnist Matt Levine in 2020, says that people will buy stocks when buying stocks is more fun than other things they could be doing for fun. So, many experts thought that since many missed their usual fun activities during the 2021 lockdown, they resorted to trading intangibles. Most predicted that the bubble would burst once people got back to their usual activities. But it does not seem to be so; NFTs are still on a bullish run!

Shaking up the art world

Considering that it is art that threw NFTs into the limelight, we think this demands some special attention before we delve into other segments of NFTs.

Digital art: NFTs offer a great way to monetise digital art. Artists can mint NFTs of their digital art and sell them on digital marketplaces. They can engage directly with their fans on social media and create enough awareness (or hype) about the NFT to ensure it sells when launched. International art dealer, Stephen Howes of Thomas Crown Art, remarked in a recent press release on NFTs, "It allows artists to make some money and exercise control over their work, to sell it more easily, reach a global audience, to more strongly protect against others appropriating it without permission, and sidestep the traditional biases of the 'art industry mafia'."



Figure 2: 'The Coquette'—a privately-held painting of Raja Ravi Varma, recently digitised and launched with the exclusive digital signatures of the Raja Ravi Varma Heritage Foundation (Source: RtistiQ)

There have been stories of people who made their fortune through NFTs. Robness, popularly known for his trash can art, was living out of his car and doing odd jobs when he started dabbling in the crypto world in 2014. He is now an NFT artist, with projects hosted across many marketplaces. His NFTs depict many elements of modern life—including trash cans, and an application he made to McDonalds! Most notable is his *64 Gallon Toter,* an NFT made from a photo of a trash can, which sold for $252,000 on OpenSea, and is currently valued at around $80,848,305.

Not ones to miss the trend, leading auctioneers like Christie's and Sotheby's have also tied up with famous digital artists like Sara Zucker and Refik Anadol to sell NFTs associated with their digital artwork, showcasing them in virtual galleries and real-life screens. Sotheby's even has a gallery in the virtual world Decentraland.

Physical art: NFT helps to build trust and transparency into the traditional art market too. Considering that art has usually been a high-value space and one shrouded in mystery (there have been gruesome crimes committed for the sake of art) a reliable method of authentication and the ability to track-and-trace can solve many headaches, including counterfeiting, theft, and fraud.

"NFTs (off-chain) represent a digitised proof of ownership of an asset (or even a liability) irrespective of its nature. In the case of physical art, it effectively replaces the physical Certificate of Authenticity issued by the creators, which has always been susceptible to tampering, in the absence of an issuing authority. As an extension to the tokenisation of physical art, we also offer a tamper-evident microchip, which acts as the physical-digital link for the artwork and can be verified for authenticity using any near-field-

communications (NFC) enabled mobile device. Further, the token transfer is linked to the shipment of the artwork to ensure a proper transfer of asset and acceptance before the transaction is reflected on the chain," explains Jothi Krishnan Menon, co-founder & CEO, RtistiQ, an online art marketplace based in Singapore. They use NFTs in the sale of traditional physical art as well as for modern digital art.

Phygital art: A use case that melds physical and digital art is where a traditional piece of artwork is digitised to extend an artist's legacy beyond geographical boundaries. RtistiQ ventured to digitise some of Raja Ravi Varma's privately-held paintings, which are considered national treasures and cannot be taken outside India. A gifted artist, Varma was ahead of his times. He thought that his work should reach everyone and not just the elite, and for that he set up a lithographic press in Mumbai as early as 1894! Raja Ravi Varma's paintings of Indian gods and goddesses frequently featured on calendars, and it became a habit of the previous generation to carefully frame the pictures at the end of the year to deck up their puja rooms! Today, people across the world have a chance to own NFTs of Raja Ravi Varma's works.

"At RtistiQ we probably started the reverse order of using NFTs for traditional mediums first, and that has started to gain acceptance as a model to meet some of the existing challenges of the industry. More recently, we have been working on different use cases extended to digital NFTs, to bring forward works of a unique nature, which may not be transactable in the physical form. In the case of the Raja Ravi Varma collection, brought out in collaboration with the Raja Ravi Varma Heritage Foundation, we tried two different models. Two of the paintings, which are privately-held and have been greatly admired, were brought out by gallery g in Bengaluru, India. Popular lithographs, recreated and embellished for enhancing the works further, were auctioned as a dual-combination of digital and physical works," explains Menon. In March this year, tokenised versions of Raja Ravi Varma's works were launched online by RtistiQ in collaboration with the Raja Ravi Varma Heritage Foundation and gallery g.

Recently, they have been helping Ukrainian artists who lost access to their physical paintings due to the war with Russia. They are minting digital editions using available images of the paintings, so the artists can still make a sale of their painstakingly created works. "We are working towards bringing forward more use cases as a combination of traditional and digital mediums for the art community through NFTs, where the original is difficult to

Ownership versus copyright

So, when you buy an NFT, what do you get? An NFT is not to be confused with copyright – an NFT only gives you a proof of ownership. There is no copyright transfer, unless explicitly mentioned. You get ownership of the artwork, with licensing terms encoded as metadata along with it. Some creators will let you use your copy only for personal use, while others might permit commercial uses. You will be free to showcase the NFT in a digital or physical gallery and resell it in a secondary marketplace. Some creators may demand a royalty every time someone resells the piece. For example, RTFKT takes a 5% cut of every sale and resale of an avatar, and 10% for all other products including virtual shoes. This is an example of royalties being built into the NFTs.

Since the copyright still exists with the creators, they can create more NFTs based on their art. They can even make more copies of the exact same NFT, but those can only be sold as 'copies' as someone has already bought the 'original.' There is a tool called CloneMyNFT, which the developers describe as an NFT backup service. It lets the creator keep a copy of the NFT in their wallet even after selling the original. According to the London based software company, "The system works by creating an exact digital copy of the artwork but with another unique contract on the blockchain, effectively making it an almost exact clone of the original NFT. The digital art copy will lack the provenance of the original NFT but will look identical in appearance and could even be sold as a copy."

"Where the confusion starts is when we mix ownership with copyrights, which are totally independent and governed by what the creator is offering. NFTs in there current form only resolve the ownership aspect of digitised assets, and not essentially the copyrights. As the crypto and blockchain space evolves further we should be seeing decentralised management of copyrights, which then could be linked or encoded with the NFTs instead of just residing as a metadata," comments Menon.

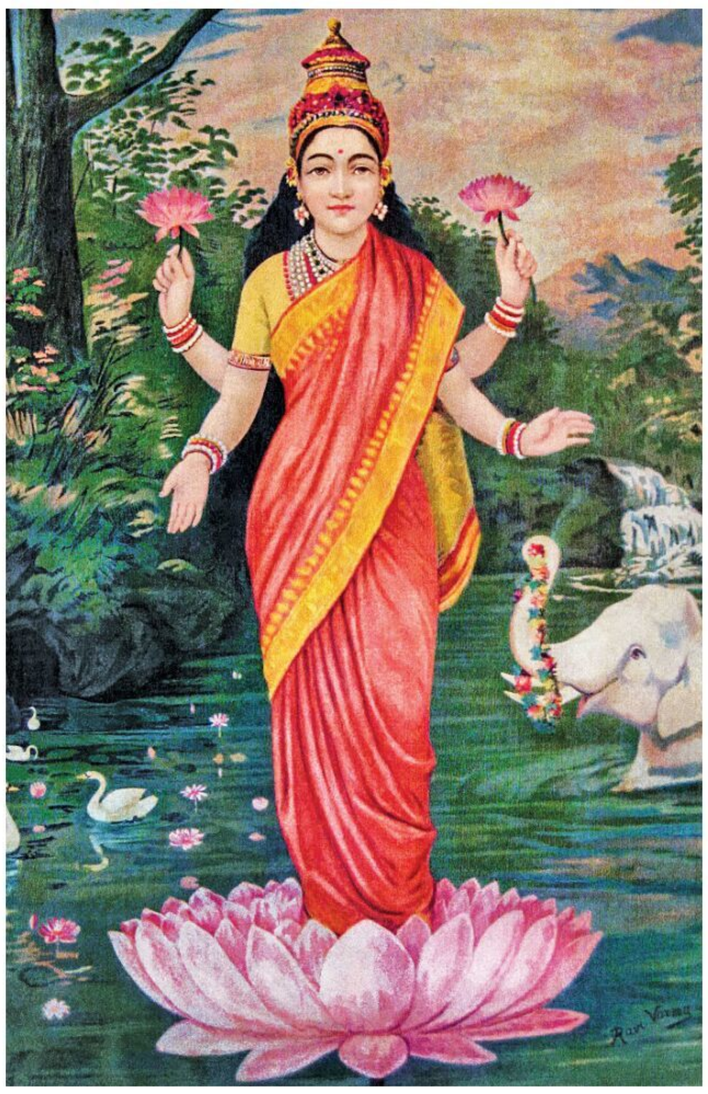


Figure 3: A famous lithograph of Raja Ravi Varma, recreated and auctioned as a combo of digital and physical works (Source: RtistiQ)

experience or transact in the current form," says Menon.

As artists struggle to salvage some of their works, multiple organisations, including Holy Water, Vandalz for Ukraine, Forze Ikonia, Ukraine DAO, and ArtWaRks Ukraine, are auctioning NFT collections for funding relief in the war-torn country.

Music and videos too: Art NFTs are not restricted to images but also extend to music and videos. In 2021, the background score of the film Triumph was launched as an NFT. American rapper Snoop Dogg is collaborating with Clay Nation to launch some of his unreleased music and limited-edition pitches as NFTs on the Cardano (ADA) blockchain. When you read this story, South Indian playback singer Karthik would have launched his first collection of music NFTs on India's first fully-curated NFT marketplace, Jupiter Meta, through his special metaverse solo concert on April 14, 2022.

Twitter, Instagram, and YouTube are all planning to integrate NFT features. In a blog post about the company's Web3 plans for 2022, YouTube mentioned introducing NFT and blockchain features to allow creators to build deeper relationships with fans and monetise their creations, while also providing fans an opportunity to own unique content. Twitter has been experimenting with hexagonal profile pictures to use NFT avatars. It will also bring in NFT authentication features along with Bitcoin and other crypto-payment systems.

Collectibles of every imaginable kind

Collectibles and memorabilia of every kind are floating around in NFT marketplaces—and pretty strange ones too. People buy NFTs of newspaper clippings, magazine covers, tweets, and memes. They buy GIFs featuring tacos, toilet paper, and cards. They buy NFT trading cards featuring key moments from the sporting world, and or milestones from the lives of their favourite actors or musicians. Note that much of this, like Jack Dorsey's first tweet or a football top shot, is already available on the Web for all to see. Why would people spend money on collecting such NFTs?

To answer this, ask yourself why people have historically collected seemingly small-value things like stamps, scale models, and sports trading cards. Well, the logic behind a collection is that its value increases over a period of time. When stamps go out of print, their value increases. Trading cards with MS Dhoni or Virat Kohli can be bought off shelves today for an insignificant price, but some years later, when cards with other faces fill the shelves, these will become scarce and valuable!

Then there is the hero worship aspect too. Fans want to own a piece of their hero's journey—like buying the ball that went for a winning sixer, or the sneakers that an athlete wore while winning a key race. They want to collect trivia and trinkets associated with their favourite brands.

So, people buy NFT collectibles hoping that scarcity and fan-following will increase the value of these collectibles in the secondary market.

Brand marvels: Taco Bell launched a collection of taco-themed GIFs and Pizza Hut sold pixelated pizzas, while Charmin sold non-fungible toilet paper, and TIME Magazine auctioned NFTs of some of its famous covers! Adidas debuted NFTs in December 2021 and sold $23 million worth within hours. Some of their 'Into the Metaverse' NFTs purchased at $765 at drop are now selling for over $2500 on OpenSea. Mahindra & Mahindra launched a collection of



Figure 4: An NFT from Adidas' Into the Metaverse collection (Source: Adidas)

The hidden charges

Reading about all this makes every other person eager to try their luck in minting NFTs! With mobile apps like Sketchar, you can even create NFTs on-the-go, right? While technically speaking anyone can mint an NFT, practically it is not so easy. First, you need a wallet, and it needs to be loaded with crypto, even to mint and sell an NFT. Most sites charge for buying and selling—very few let you list an NFT for free, and then charge a commission on the sale price. Along with this, many marketplaces also charge a 'gas fee' to compensate the energy consumed to complete the transaction, considering that it is an extremely energy-intensive process.

Add to this the conversion fees charged by crypto-exchanges to convert your money into cryptocurrencies, the rapid fluctuations that the crypto-market is known for, and the need for a reasonably advanced computer system to mint an NFT, and you might just end up spending more than you bargained for.

Buying NFTs is also not easy—at least not the ones with high potential—because many NFT drops are by invite. To get into one, you need to be active on the project's or creator's social media, and garner enough attention to be considered worth being in the White List.

four NFTs featuring the all-new Thar, on Tech Mahindra's NFT marketplace, Mahindra Gallery. The amount raised was donated to Project Nanhi Kali for the education of underprivileged girls in India. Anand Mahindra tweeted about the new experiment, "We believe it's not just a make-believe world, it's also a place where we can explore solutions for making the real world a better place. Only fitting that the Thar—our flagship off-roader—leads us into the uncharted landscape of NFTs."

Brands around the world are trying to engage with their customers in the emerging digital world. Some of the brand based NFTs are assets or accessories that can be used in the virtual world, like digital sneakers or sunglasses that your *avatar* can wear, while others are merely collectibles that are expected to increase in value.

Sometimes, the brand based NFT comes coupled with some tangible reward—after all, the customers have shown great loyalty! For example, some of the sneaker NFTs developed by RTFKT, which Nike recently acquired, include a tradeable ticket for a physical shoe that you can collect in the future. Similarly, the winners of Mahindra's NFTs also got a chance to experience the all-new Thar in one of Mahindra's 4x4 off-roading tracks. Adidas' debut NFTs offered the holder virtual wearables and the right to redeem them for physical freebies like a hoodie, a tracksuit, and a beanie! NFTs can be a great way to presell physical products in the virtual world!

Hero worship: When Under Armour launched a series of sneaker NFTs to commemorate Stephen Curry's NBA three-point record, it sold out in less than a minute! A single LeBron James highlight NFT sold for $200,000. Guardian Link and BeyondLife. club recently launched an exclusive collection of 250 NFTs dedicated to Indian-origin astronaut Kalpana Chawla, with never-before-seen images shared by her family. The proceeds were donated to an Indian charity for education. World over, celebrities like Snoop Dogg and Lindsay Lohan are launching unique memories and moments as securitised NFTs.

Sports collectibles: The US basketball league NBA is selling officially licensed digital collectibles under the banner NBA Top Shot. These are basically NFT trading cards embedded with notable basketball moments. After tasting success, they are now expanding into NFTs of virtual jewellery, accessories and clothing that can be used across social media.

Crictos is a series of official digital collectibles being launched by the International Cricket Council in association with FanCraze, which aims to build the ultimate cricket metaverse. Crictos gives fans a chance to own pieces of cricketing history. Different levels of Crictos are minted—ranging from common (extensively minted) to Genesis (the rarest ones). Each represents a highlight-worthy event in cricketing history. The current collections include clips from ICC World Cup 2015 and 2019. The packs are dropped on FanCraze.

"NFTs are a golden money-making opportunity for the sports industry and also a great way for sports fans to engage with their favourite teams, athletes, and historical moments. People have always loved to own sports collectibles," says James Green, group investment director at deVere, which has recently launched the dV Gems Platform to help clients identify and invest in winning NFTs.

Avatars

According to CoinMarketCap, the top seven NFT collections by market cap are avatar NFTs—headshots of cartoon characters or pixelated people. The most popular of these collections are CryptoPunks, Bored Ape Yacht Club (BAYC), and Pudgy Penguins. Quite a few studies show that avatar NFTs are overtaking metaverse game assets, fine art pieces, and sports NFTs.

Avatar NFTs can be your digital profile in the metaverse—across virtual worlds and social media platforms. Twitter, for example, lets you sport your avatar if you own one, so it becomes like a trendy and tech-savvy status symbol for many! Tonight Show host Jimmy Fallon and Grammy-winning producer Timbaland have Bored Ape NFTs as their Twitter profile pics, along with many other

Creating scarcity

The value of a collectible increases with scarcity. In a physical market, scarcity is inevitable at some point of time. Products get sold out, technologies get phased out, fashions change—it just happens with time. However, digital creations usually have a huge or infinite supply. For example, an audio track can be downloaded any number of times, and any number of years later from some web source. NFTs aim to create digital scarcity to increase the value of associated assets. NFTs that are rare and few have a greater value than those that are mass-minted. So, when the NFT is no longer available in the primary market, its value in the secondary market increases, especially if it is something that people want to own.

Here is the story of someone who took this concept too seriously! In March 2021, a group of private investors purchased Banksy's 'Morons' for nearly $100,000. They digitised and minted this as an NFT. They then burnt the original print in a live-streamed video and put up the NFT for sale. It sold for approximately $380,000!

Ironically, 'Morons' by graffiti artist Banksy depicts a Christie's auctioneer pointing at framed paintings in a crowded auction room. Next to him is a framed image accompanied by a phrase including the words, "I can't believe you morons actually buy this!"

Showcasing (showing off?) your treasures

Some collectors are secretive about their collections, while others love to flaunt it. One visit to the GD Naidu Car Museum in Coimbatore, and you will instantly understand what I mean. The flair with which the vintage cars are displayed resonates the pride of the collector. So, if you have collected a couple of good NFTs and want to display them, where would you and how would you?

- You can print, frame, and hang the NFT on a cosy wall in your house, if the contract permits.
- You can display the NFTs in digital marketplaces.
- You could download and install an app like token.art that lets you view NFTs easily on your phone. Most wallet apps also let you do that.
- If you own some space on the metaverse in virtual worlds like Decentraland or Cryptovoxels, you can display your NFTs there.
- You can use software platforms like Samsung Smartthings or Tokencast to download NFTs from your wallet and view them on state-of-the-art televisions.
- However, if you have a good collection, the best option might be to invest in a suave and smart NFT display with direct wallet integration, like Netgear's Meural, Lago's NFT Frame, Blockframe, or Tokenframe.

celebrities. Last year, Visa purchased a CryptoPunk avatar for $150,000, and Adidas purchased an Adidas-clad ape named Indigo Herz from BAYC, which is one of the key collaborators in Adidas' NFT projects.

You can get membership to the BAYC by purchasing one of 10,000 ape profile picture NFTs, which cost at least $200,000 each. The membership gives you access to exclusive features, product drops, and events too. It allows users to mint art for the group. It gives IP control to the community, thereby having an unending source of fresh ideas. Primary and secondary resell rights ensure that the artists get paid for their efforts.

In-game assets

NFTs enable you to own characters or items on gaming platforms and virtual worlds. In-game assets come in many forms—ranging from creatures in Axie Infinity and trading cards in Gods Unchained to virtual real estate in Decentraland. Gaming platforms often use tokens to reward players. In the play-to-earn model followed by platforms like Axie Infinity, you earn NFT rewards within the game, which you can monetise by selling in secondary markets.

With gaming platforms growing in scale to become virtual worlds with thriving marketplaces, real-world brands are also entering the space and selling their wares. You can, for example, buy virtual sneakers from Nike or Adidas for your avatar to wear in the virtual world.

The trend is advancing towards interoperable NFTs that enable cross-platform playability. You can take your NFTs along with you to other platforms to create personalised experiences or for trading. The Enjin generic axe is one such gaming NFT that you can take along to multiple platforms!

"NFTs are a standard protocol, so they can be traded across NFT marketplaces as long as the wallet holding the tokens is compatible and on the same blockchain network. So, anyone buying an NFT from one platform can resell it on another platform by connecting their wallet and having the smart contract listed in that platform. Cross-chain compatibility (cross-chain bridge) has recently been established across a few chains for NFTs but has also been a source of vulnerabilities and hacks," says Menon.

With the advent of currencies and assets, gaming platforms are coming under the influence of market forces much to the displeasure of traditional game developers. According to the Game Developer's Conference 2022 Annual Report, 70% of the developers surveyed said their studios were not interested in integrating NFTs or cryptocurrencies in their games.

Nascent real-world applications

NFTs have the potential to transform several real-world processes. Experiments are underway but it will probably take time to become mainstream, because of the high costs and energy-related issues surrounding NFT usage. Nevertheless, let us take a look at some of the most promising applications:

IDs, certificates, and other documents: NFTs being immutable can result in accurate record-keeping with no risk of tampering. An NFT birth certificate can serve as a lifelong ID on the blockchain. Other medical records can be appended to the same, with complete trust and confidentiality. Some graduate schools like Emlyon in France are planning to issue NFT based diploma certificates to avoid certificate related frauds. Going forward, other important documents like passports, driving licenses, and other documents can also be encoded in the blockchain. This would help iron out kinks in several administrative processes ranging from public distribution to voting.

Real estate: In the real world, NFTs could be an efficient way to check titles and verify ownership history. Timestamped NFTs are a good way to track changes in property value. Smart contracts can be used to simplify and speed up transactions. It becomes possible to create decentralised home rental services with auto payments

facilities, while taking care to protect sensitive information. In February 2022, the sale of a physical house in Tampa Bay, USA, was executed using NFTs, for a value of 210 Ether, approximately $653,000!

IP and patents: If IP is minted with a timestamped token, its current ownership and history can be easily verified and proved. Patent wars would be so easy to sort out! Sometimes, an inventor develops certain innovative IP along the way as a means to the whole. If the attempt is successful, he or she might file for a patent. If, unfortunately, the final result is not as expected, the creator can at least tokenise the successful components developed along the way and make them available to the community. These IP NFTs can also be monetised. Some researchers are considering self-publishing their work as NFTs so that their work can quickly reach the world, without the delay that is inherent in the patenting process.

Supply chain and logistics: NFTs can be used to authenticate products, verify their origin, and ensure quality. NFT based supply chain data cannot be tampered, and helps avoid counterfeiting in food, medicine, and high-value luxury products. In eco-sensitive industries, it can be used to track and trace recyclable and sustainable materials and ensure their proper recycling or reuse. Carrefour Supermarket uses NFTs to guarantee secure and unfalsifiable traceability of products. The Ownest platform also enables you to track ownership on supply chains using NFTs.

Industrial design and manufacturing: By tokenising the CAD data or files, manufacturers can ensure that their designs are not tampered or copied by contract manufacturers. File management and revisions can also be handled efficiently. Using NFTs, a manufacturer can track and trace components to ensure authenticity, and at each stage include information about the manufacturing process for the benefit of the next player in the workflow. NFT.SMARTMFG.IO from blockchain company Smart MFG Tech is a comprehensive tokenisation and marketplace solution for the creation, storage, sale, and transfer of industrial design assets as NFTs. It allows users to create and list unique NFTs, participate in auctions and commission custom-made industrial design and 3D models.

Problems aplenty

NFTs are not without their share of problems. Being a blockchain based technology, just like cryptocurrencies, it faces a lot of flak over its heavy energy usage. The proof-of-work system that keeps records secure on a blockchain is very energy-intensive. CrytoArt.wtf calculated that a simple GIF NFT can use around 200kWh for its creation, which is close to two weeks' power consumption of an EU resident.

Some players like Sega and ArtStation dropped their NFT plans after they faced a heavy backlash from the community over energy concerns. Beeple has promised to create carbon neutral work in the future. New blockchains like Palm and Flow claim to be low-energy and carbon-neutral, and plant trees to compensate for the environmental damage, if any. Polygon claims that you can mint an NFT on its blockchain using just as much energy as it takes to send three emails. Ethereum is also rolling out Ethereum 2.0 in phases. The 2022 and 2023 upgrades will enable the

To buy or not to buy

To buy or not to buy an NFT is a difficult decision because there is no thumb rule. At present it is highly speculative, and hence a very personal choice. The price of an NFT is entirely demand-driven and depends on how badly people want to buy it. A promising NFT might just fizzle out, or a seemingly dumb one might end up as a cash cow just because it caught someone's fancy! You buy an NFT hoping someone will want to buy it for a greater price in the secondary market—something that you have no control over. So, here are a few tips from experts on how to buy NFTs:

- Since it is a new trend with not much historic data, invest a small amount initially and see how it goes.
- Buy only what you can afford to lose.
- Make sure it offers some value to you—either emotional satisfaction or physical utility. You can buy it if you like the art or the artist, or believe in the cause behind it. You can buy it if it is something you really wish to own—a collectible close to your heart. That way, you do not feel the loss even if it does not sell in the secondary market!
- Do not buy NFTs just for the fear of losing out, or because you were invited to a drop.
- If you cannot get proper information about the art and the artist, avoid putting money in it.
- Look out for the good NFTs from social media, NFT marketplaces, and NFT drop calendars.

Do your due diligence each and every time, because no two NFTs are the same—remember they are non-fungible. "As not all stocks are same, so is the case for NFTs as well, and it is dangerous to bucket them all under one umbrella when thinking about investments. Just like we do research on which stocks or even property offers the potential of appreciation, we should be looking at individual NFTs in the same light. Each project has been created with a different set of utility surrounding it, being just for the artistic nature, access to some kind of associated utility, or even managing loyalty programs with multitude of variations," explains Menon.

platform to process thousands of transactions per second and scale globally, thereby reducing energy consumption and gas fees significantly.

Trading transactions can also be very expensive and slow because of the on-chain nature of NFTs. Experts suggest sidechains and rollups as possible solutions. Sidechains are separate blockchains that run parallel to Ethereum. A rollup takes multiple Ethereum transactions and rolls them up into a single piece of data before submitting them all to the blockchain.

Another risk is that of bit rot—the tendency of digital information to degrade or become unusable over long periods of time.

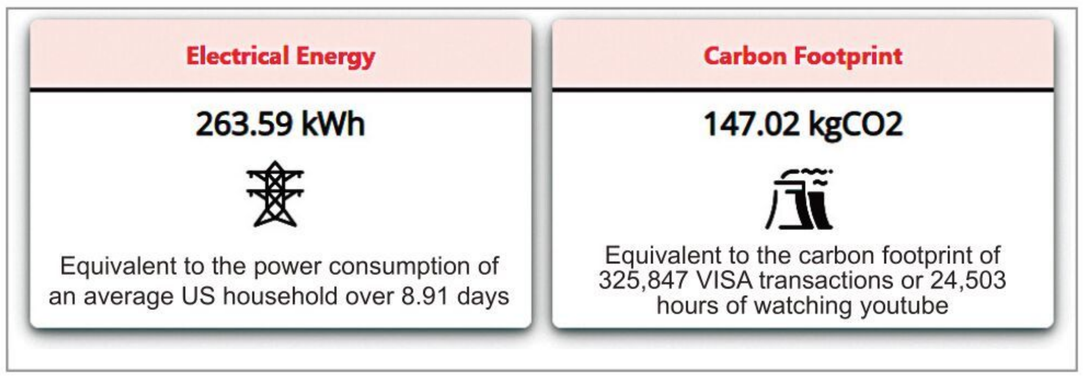


Figure 5: Footprint of a single Ethereum transaction (Source: Digiconomist)

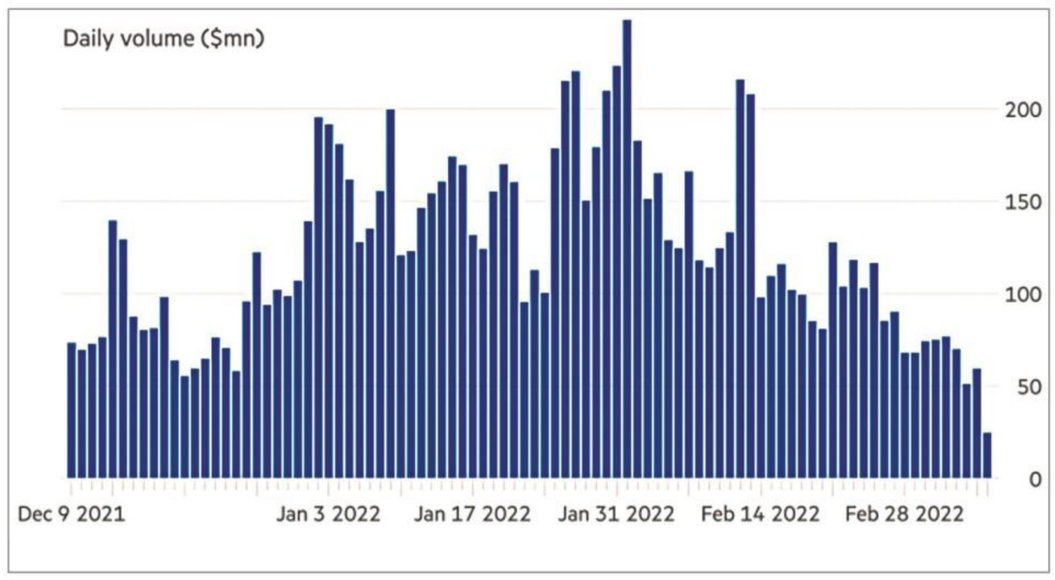


Figure 6: Trading on OpenSea in decline (Source: Richard Chen/Dune Analytics)

There is also the effect of frequent fluctuations inherent in the crypto-world. According to a Financial Times report, NFTs saw $41 billion in sales and huge VC investment in 2021. However, daily trading volumes in OpenSea went down by 80% in March 2022. The average selling price of NFTs and the number of accounts buying and selling NFTs also went down. This shows the fluctuating nature of this business and raises questions about its future. However, considering that the values in the crypto-world fluctuate rapidly, there is a high possibility of these figures being up when you read this article!

Despite its merits, the future of NFTs seems to be uncertain—but that is how most of today's mainstream technologies were in their nascent stages.

Should you invest in NFTs yet?

The blockchain and its spin-offs like cryptocurrencies and NFTs seem to be swerving from their intended path.

Ethereum co-founder Vitalik Buterin is not a big fan of NFTs as they are perceived today. Buterin, who hopes for the Ethereum to be a launchpad for all sorts of socio-political experimentation like fairer voting systems, urban planning, universal basic income, and public-works projects, is concerned about the dangers to overeager investors, the soaring transaction fees, and the shameless display of wealth that have come to dominate public perception of crypto. "The peril is you have these $3 million monkeys, and it becomes a different kind of gambling," he says, referring to Bored Apes, when speaking to TIME magazine in March this year.

"People may have lost their mind," says C.Z. Zhao, founder of Binance, a leading cryptocurrency exchange, who told Fortune recently, speaking of the NFT craze. He was quick to add that there is an undeniable plus, as NFTs allow artists to monetise their work globally and reach a much wider audience.

The Indian government is also discouraging crypto-investments. Amidst fear of it being banned in the country, Finance Minister Nirmala Sitharaman announced in this year's annual Budget presentation that income from cryptocurrencies and NFT trade will henceforth be taxed at 30%. This is seen by optimistic investors as a step towards regularising crypto-investments, although the government said that it has not decided yet on the status of crypto in the country.

Surprisingly, the man who struck gold, Mike Winkelmann, also seemed worried about NFTs when speaking to Fox News Sunday last year! "I absolutely think it's a bubble, to be quite honest. I go back to the analogy of the beginning of the internet. There was a bubble. And the bubble burst. But it didn't wipe out the internet. And so, the technology itself is strong enough where I think it's going to outlive that," he remarked.

Yes, NFT as a technology is a worthy one—but better put to use to induce efficiency and transparency in existing systems. While minting and trading of NFTs can be used for meaningful engagement with fans and customers, it must not become unethical and destabilising like gambling! Over time, quick-money fads will disappear, leaving more room for beneficial applications. END

By: Janani G. Vikram

The author is a freelance writer based in Chennai, who loves to write on emerging technologies and Indian culture. She believes in relishing every moment of life, as happy memories are the best savings for the future.

This article was originally published in the May 2022 issue of Electronics For You.

Exnovation: Another Side of Innovation



Once a product has been developed well and accepted by the consumers, do we need to keep tinkering with it and try improving it further? Well, the principle of exnovation advises against doing so. And there are, obviously, valid reasons for that.

Too much of anything is bad; for business and for anything else as such. Innovation is not an exception. Obviously, you cannot keep innovating the same product or process repeatedly. Not for the extended period. Therefore, we need to talk about the other side of the innovation coin—exnovation!

You may have barely heard of the word exnovation. The earliest usage of this term was observed in 1981 when John Kimberly referred to "removal of innovation from an organization." In 1996, A. Sandeep provided the modern definition of exnovation as the philosophy of not innovating. In other words, ensuring that best-in-class entities are not innovated further.

However, exnovation still has not become common parlance, at least in the intended way.

Why exnovation

The basic definition of exnovation is self-explanatory and quite simple *per se*. However, why do we need it is still a question. Especially if we are comparing it with the continuous innovation paradigm.

When should you call it a day and stop continuous innovation? And, of course, this itself demands a balanced strategy.

But, let me assure you that exnovation is not about refusing innovation in the company. It is about channeling it and narrowing the process down over a period. For example, you may be soliciting innovative ideas from almost everyone initially. It is the phase of exploration where quantity over quality matters. This is when you want everyone to think innovatively and contribute their ideas.

Exnovation is not about refusing innovation in the company. It is about channeling it and narrowing the process down over a period.

However, once you shortlist a few ideas to implement and the execution starts, there is no need to get more of the same ideas again. Moreover, once the changes are made, a new product or process is introduced, and the innovation has been implemented, it is time to reap its benefits. At this stage, colluding tried and tested outcomes with some more new ideas does not make any sense. It only delays the payback.

This is the stage when the focus must shift to exnovation. Earlier, you may have encouraged people for being more innovative. However, you must also encourage them to comply with the newly established product or service process. Then only the innovation that you implemented will result in profitable returns.

When and where to use it

Interestingly, in his book, 'The Myths of Innovation,' author Scott Berkun writes, "Competence trumps innovation. If you suck at what you do, being innovative will not help you. Business is driven by providing value to customers. Often, that can be done without innovation; make a good and

needed thing, sell it at a good price, and advertise confidently. If you need innovations to achieve those three things, great, have a go at it. If not, your lack of innovation isn't your biggest problem."

This thinking highlights an important principle of exnovation. For a multinational company with a presence in different geographies, you often face region-wise challenges. These challenges result from apparent differences in consumer behaviours compared to the parent market of the company. So, how do you manage such matters?

In my experience, a set process and structure of innovation and strategy are important. Additionally, specialised teams with vested authority make a good fit here. They can study the local market's needs and see how they could be fulfilled within the existing system. Then, if any changes are to be made, they can tailor the existing process, products, or system. I wrote about these special-purpose teams in a previous article in this series.

Special-purpose teams play a crucial role in innovation as well as exnovation.

During my tenure with LG Electronics in India, each LG factory had its own R&D and TDR (special-purpose tear down re-engineering) teams. So, we would work on joining and separating innovative projects all the time. But when it came to feedback from the local market and adapting the products, each team had a degree of freedom to work on it for making necessary changes that suited the local market and implemented them at scale. And for those improvements that were nationally or globally applied, the company made sure that these improvements got adopted by all.

The point is, if changes in established design and processes are to be made to suit geographic needs, teams need to be empowered to experiment with those changes first. Then, once validated, there must be a process to recommend them to other units and have them formally introduced or regularised for all, when applicable. Once done, it becomes a standard process again, which need not be tampered with any further.

"Exnovation is about hammering out a process and structure to implement it at scale. Once established, everybody need not innovate but implement it to the 'T.'"

Important piece in innovation puzzle

Exnovation is not about saying no to innovation. It is about hammering out a process and structure to implement it at scale. Once established, everybody need not innovate but implement it to the 'T.'

Counterintuitive but true. You can then put a team in charge that will check processes and structures throughout the organisation and improve them when the need arises. Exnovation means the opposite of innovation only to an extent. Mainly, it is about keeping a check on the innovation process in your company.

Once a process, product, or system is evaluated, validated, and almost mastered by the special-purpose team in the organisation, the next step is to put in place a critical system. A system that will ensure this new change is replicated across branches, departments, units, factories, or similar groups within your organisation.

Ensuring the process is implemented without any alteration is highly important. No one within the organisation should be allowed to tailor the process further for whatever reason.

Innovation can give an organisation the much-needed big leap, but only if favourable processes and systems are passed on and repeated seamlessly by all involved. A strict check on customisation by anyone at random must be kept. Let the special-purpose teams take that call. Give them the liberty and latitude to manage it.

The point is

In defence, not everyone is a commanding officer. In orchestra, not everyone is a conductor. In an organisation, not everyone can be an innovator.

If you want to be an innovative company, you cannot be one without the process and structure around it. Whether it is around innovation activity or the downstream delivery mechanism to deliver it, innovation and process within a company must coexist and feed off each other.

Innovation, in its true sense, includes creating a culture and systems that help in effectively sustaining innovation. Your people and innovation teams are central to promoting internal and external collaboration. However, for innovation to be really nurtured, you need to establish a solid process with clear communication and appropriate employee training.

It is not just the effort that we put into innovation that decides success. It is how we go about innovating, its structure and its process. END

By: Anand Tamboli

The author is a serial entrepreneur, speaker, award-winning author, and an emerging-technology thought leader.

The article was originally published in the February 2022 issue of Electronics For You.

AI Hardware that's Tailored for Computer Vision

Neural networks, AI, ML, and neuromorphic computing are technologies that are actually ways to make machines behave more like humans! Similarly, computer vision focuses on imitating the complex human vision system and leveraging it in applications.



Drone Skygrid, an airspace management system built on AI and blockchain, uses computer vision for near real-time object detection through a drone's live video stream (Credit: Skygrid)

Computer vision is a type of artificial intelligence (AI). The aim is to make computers visualise and understand images or videos as humans do. The field has grown a lot over the last decade, especially due to the new hardware and algorithms that came into the picture. Not just that, this type of computation has become faster and more accessible since the amount of data that is generated has also increased. Thanks to improved training models, deep learning, and better hardware, we are able to identify objects more accurately today.

Computer vision works by recognising patterns in images and videos. You feed the computer with an image of a particular object and then make the computer analyse it using algorithms—the borders, colours, shades, and shapes. We are essentially training a computer to understand and interpret the world.

Computer vision: Then and now

In his whitepaper titled 'Everything You Ever Wanted To Know About Computer Vision,' Ilija Mihajlovic, an iOS developer who is passionate about machine learning (ML), explains how ML and deep learning have simplified computer vision. Before deep learning came into the picture, computer vision did not have as much scope as it does today, mainly because a lot of manual coding was required, such as creating a database, annotating images, capturing new images, etc.

"With machine learning, developers no longer needed to manually code every single rule into their vision applications. Instead, they program features or smaller applications that could detect specific patterns in images," writes Mihajlovic. Today, we use deep learning for computer vision, and all of the work is done by neural networks. When we provide neural networks with lots and lots of labeled images of a particular object, they are able to identify patterns in them.

Hardware for computer vision

Capturing data is an important aspect of computer vision applications, and hardware platforms play a major role here. The main components required for computer vision applications are cameras and image sensors for capturing images, I/Os, a communication interface, and a processing unit. Imaging sensors present in cameras are characterised by their resolution (number of pixels), speed (frames per second), and number of colours.

Depending on the type of computer vision application, the processing unit might vary from CPUs to embedded boards. Here are a few examples:

- CPUs
- GPUs
- Heterogenous systems (CPUs +GPUs+others)
- FPGAs
- ASICs
- Microcontroller boards

Computer vision at work!

For the 2018 Masters Golf event, IBM employed computer vision to produce My Moments. IBM Watson was given hundreds of hours of film, after which it was able to recognise the sights (and sounds) of key scenes. These pivotal moments were chosen and distributed to viewers as individual highlight clips!

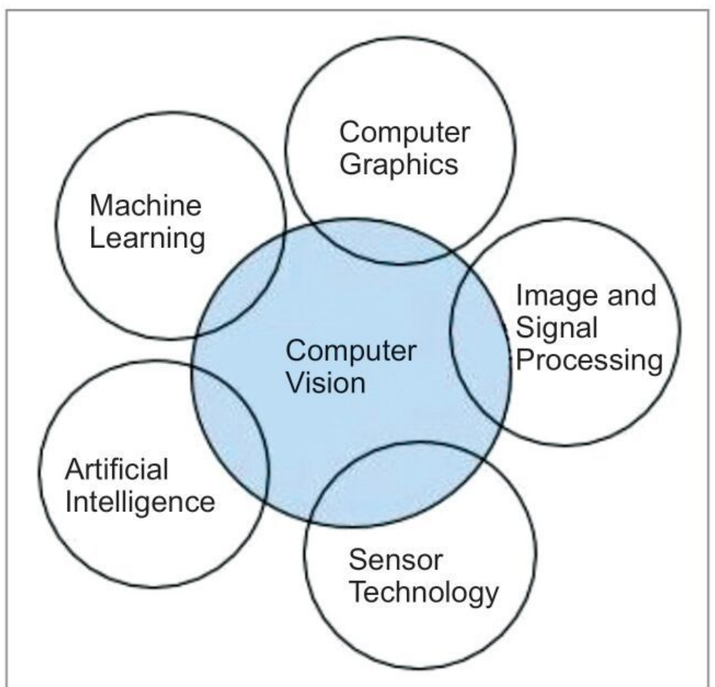
So where do development boards stand?

Traditional computer vision systems used a camera connected to a general-purpose computer, which used to run the application. In a smart camera though, the whole system can be integrated, right from capturing the image to processing to the final application. Today, the distinction between smart cameras and development boards has slowly started to fade, since many development boards have inbuilt cameras and are tailored for computer vision applications.

With more and more applications being deployed on edge, it makes more sense to deploy such devices on development boards. Here are some critical features that you need to consider while selecting a development board for your computer vision application:

- High processing speed
- Edge processing ability
- Faster access to memory
- Low power consumption
- Portable
- Inexpensive

While there are embedded platforms specifically for AI like NVIDIA Jetson and Raspberry Pi Compute, AI accelerators like Google TPU, SOMs/SBCs containing high-quality imaging sensors, we will focus on a few boards that were designed for computer vision and compare some of their features.



Fields that drive computer vision (Credit: IFA)

Current trends in computer vision

Computer vision on edge: To successfully deploy computer vision on edge, neural network design must be integrated into embedded platforms. Then, there are also difficult design decisions to be made around power consumption, cost, accuracy, and flexibility. Overall, for bandwidth, speed, and security reasons, edge computing is critical to computer vision processes. Still, hackers could take advantage of emerging security flaws in artificial intelligence. Designers must take this into consideration too.

However, there are some challenges when it comes to vision AI on edge, such as involving all edge endpoints can lead to complex device architectures. This makes

Components of computer vision

Hardware: Cameras/Imaging sensors
Software: Machine learning models for processing the images
Hardware: Processing units or conditional logic to automate application-specific use cases

Comparison of some computer vision boards

	Arduino Nicla Vision	OpenMV Cam H7	Huskylens	VC Picosmart	iVS-AWV3-AR0521
Processor	STM32H757AII6 Dual Arm Cortex M7/M4	ARM 32-bit Cortex-M7 CPU	Kendryte K21 (AI Chip with 64bit RISC-V dual core CPU)	FPGA module, High-end FPU processor	Allwinner V3 system on module
Camera/ image sensor	GC2145 (2MP colour camera)	OV7725 image sensor (can be interfaced with other sensors too)	HuskyLens – OV2640 (2MP camera), HuskyLens PRO – OV5640 (5MP camera)	1-Megapixel CMOS sensor	ON Semiconductor AR0521 sensor module
Notable characteristics and features	MP34DT06JTR digital MEMS microphone, LSM6DSOXTR 6-axis IMU, NXP SE050C2 Crypto chip	Double-precision floating point unit (FPU), Hardware JPEG encoding	It can be used as a standalone product or as a peripheral for a microcontroller or microcomputer	VC realtime-OS for real-time image processing	H.264, 1080p60 video codec, integrated ISP supports up to 8MP with auto image function
Connectivity and communication	Bluetooth, Wi-Fi, SPI, I2C, UART	Wi-Fi, BLE, SPI, I2C, UART, CAN	UART, I2C	FPC connector	Multiple wireless connectivity options available
Supported battery/ power source	Li-ion/LiPo single cell, 3.7V	3.7V LiPo battery	Two power pins on the 4-pin JST connector, micro USB connector	N/A	N/A
Suitable for	Asset tracking, object recognition, predictive maintenance	Object detection, line detection, video recording, QR code detection/ deco-ding, eye tracking, colour tracking, frame differencing	Face recognition, object tracking, object recognition, line tracking, colour recognition, tag recognition	Object recognition, position control, barcode reading, Web edge and level control	Autonomous vehicle and aftermarket home automation camera systems
Compatibility/ integrations	Arduino IDE, Arduino CLI, Arduino IoT Cloud, OpenMV IDE	Edge impulse CLI, OpenMV IDE.	Mind+Editor, Arduino IDE	N/A	N/A
Primary programming language	C/C++	Python	C/C++	N/A	N/A
Edge capability	Yes	Yes	Yes	Yes	Yes
Industry-ready?	Yes	Yes	More suited to prototyping	Yes	Yes
Dimensions in mm (l x w x h)	22.86 x 22.86	45 x 36 x 30	52 x 44.5	22 x 23.5	N/A
Weight	2gm	19gm	N/A	N/A	N/A
MSRP	$115.00	$65.00	$54.90	Quote based	Quote based
Datasheet	https://docs.arduino.cc/static/592480018567f53345d4f2b18199cd67/ABX00051-datasheet.pdf	https://cdn.shopify.com/s/files/1/0803/9211/files/OpenMV-H7.pdf?10726770933929369108	https://wiki.dfrobot.com/HUSKYLENS_V1.0_SKU_SEN0305_SEN0336	https://www.vision-components.com/en/products/oem/arm-linux/board-cameras/vc-picosmart/	https://www.ienso.com/product/ivs-awv3-ar0521/

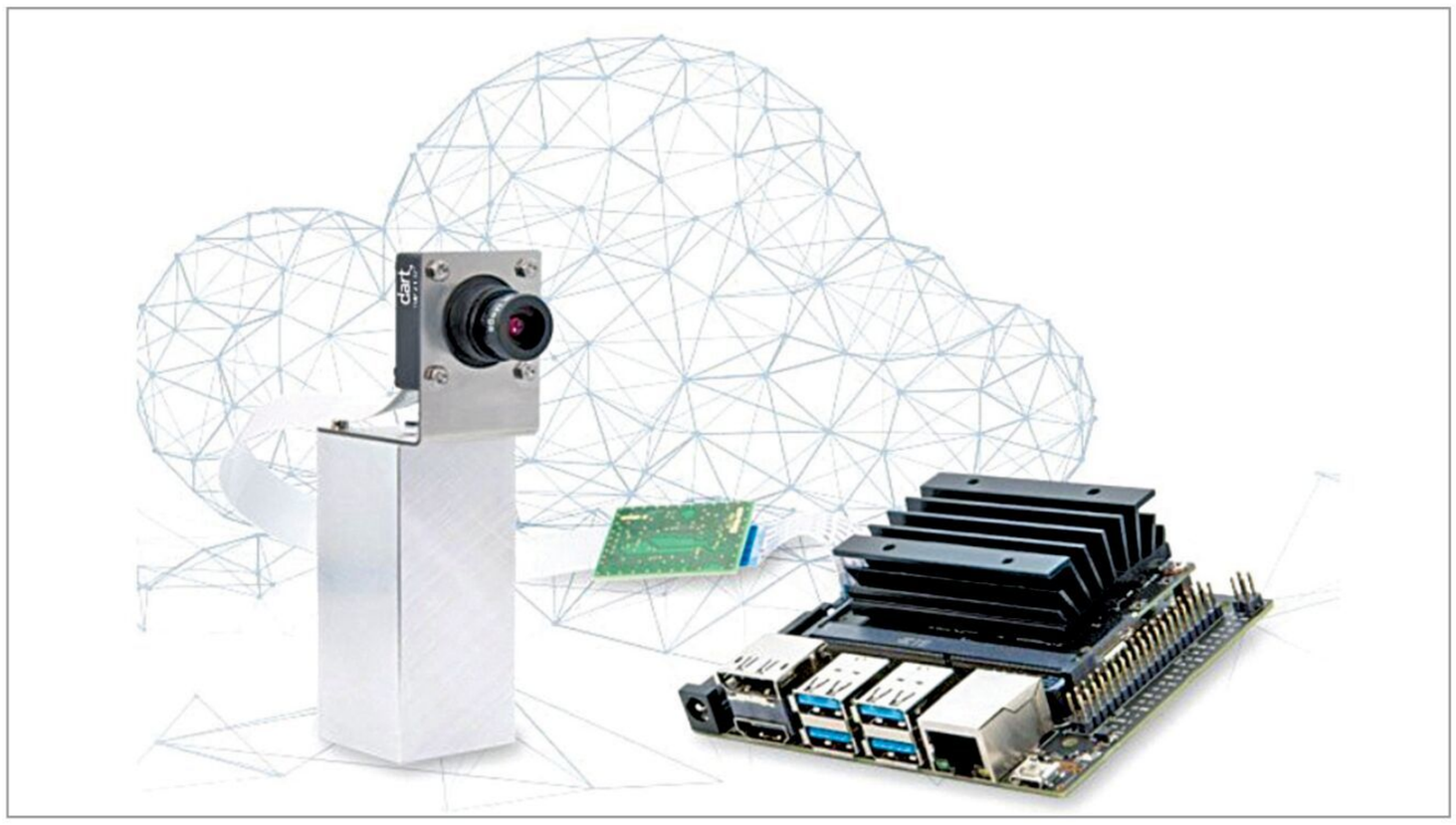
device management and edge computing implementation harder. Edge computing differs from cloud architecture as it is spread out rather than centered around a single cloud. For large enterprises, it may be assumed that such an infrastructure will have higher maintenance costs than one centered upon the cloud.

Embedded vision and IoT embedded vision: Embedded vision is when we combine cameras with the processing unit, like smart cameras mentioned before. Many companies, like Lattice Semiconductor and Blaser AG, provide embedded vision solutions. For example, Basler's AI Vision Solution Kit comes with cloud connectivity and contains a dart MIPI camera module, an NVIDIA Jetson Nano development board, lens, and cable. It allows developers to test AI based IoT applications on an optimised vision system and to access cloud services.

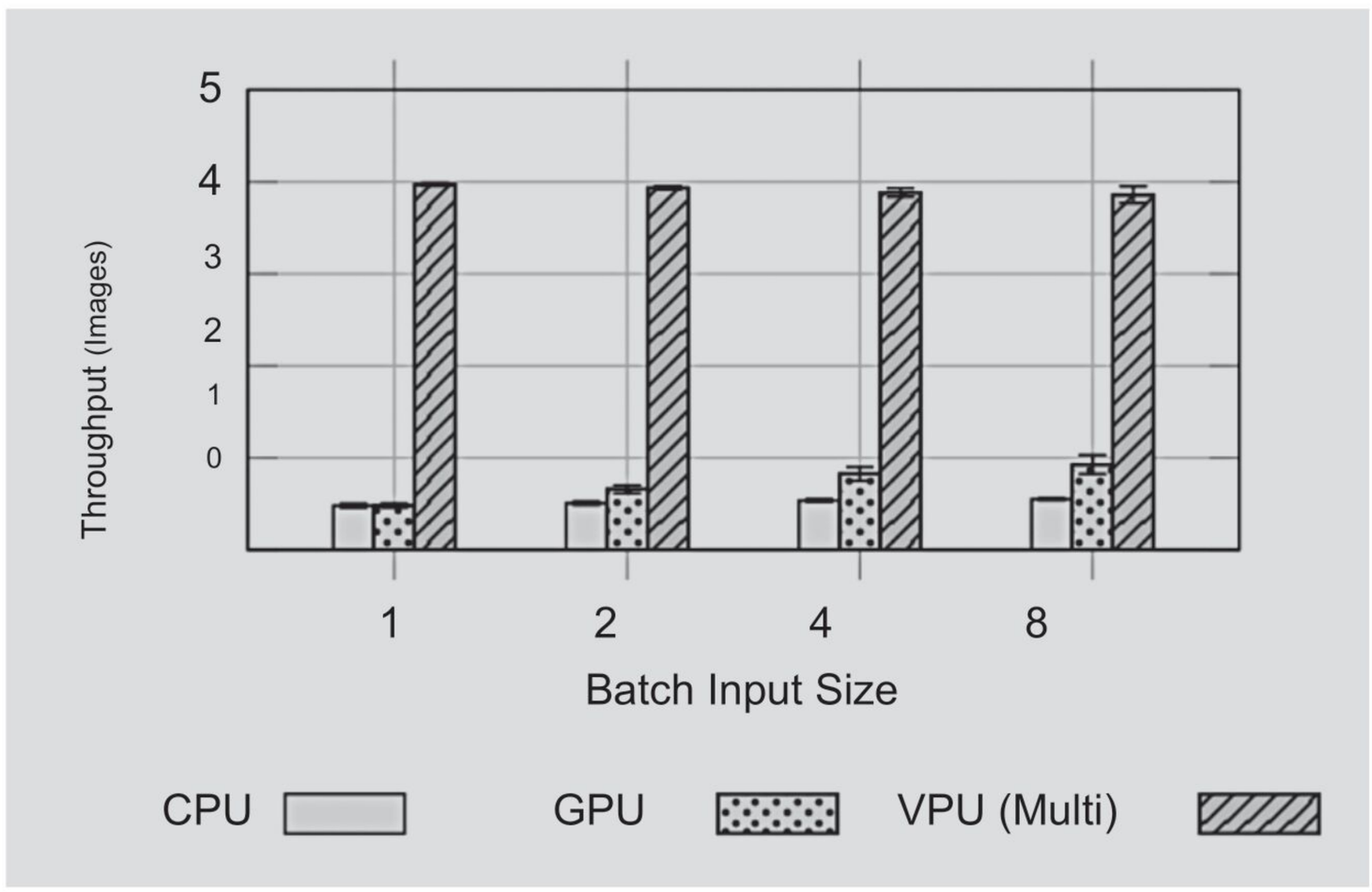
But we are about to enter the next stage of the Internet of Things. So far, the focus was on connecting devices, accumulating data, and creating Big Data platforms. Now, our focus will move to making the 'things' smarter using technologies like computer vision and deep learning on edge. This way, the things will be able to make informed decisions about their surroundings. That's IoT embedded vision.

Vision processing units (VPUs): A VPU is a specific type of AI accelerator designed for computer vision tasks. It is well suited for parallel processing. A very popular edge AI accelerator is the Intel Neural Compute Stick 2, which is built on the Myriad X VPU. It offers an easy-to-use USB interface.

Pre-trained convolutional network (CNN) models are used in the VPU chips to execute inference tasks. Studies have shown that VPUs are superior to CPUs and GPUs in terms of performance as well as power consumption. Moreover, a combination of multiple VPU chips can even reduce the thermal-design power (TDP) by up to eight times.



Blaser's AI Vision Kit (Credit: Blaser AG)



Throughput performance comparison per watt using the CPU, GPU, and multi-VPU configurations (Credit: *https://arxiv.org*)

Software 2.0 and computer vision: As we move towards Industry 4.0, software development is also moving towards Software 2.0. Software 1.0—the technology stack that includes languages like C/C++, Python, etc—is the one that we are familiar with. It needs a coder. On the other hand, Software 2.0 literally writes itself when we feed enough data to a neural network that knows how to code.

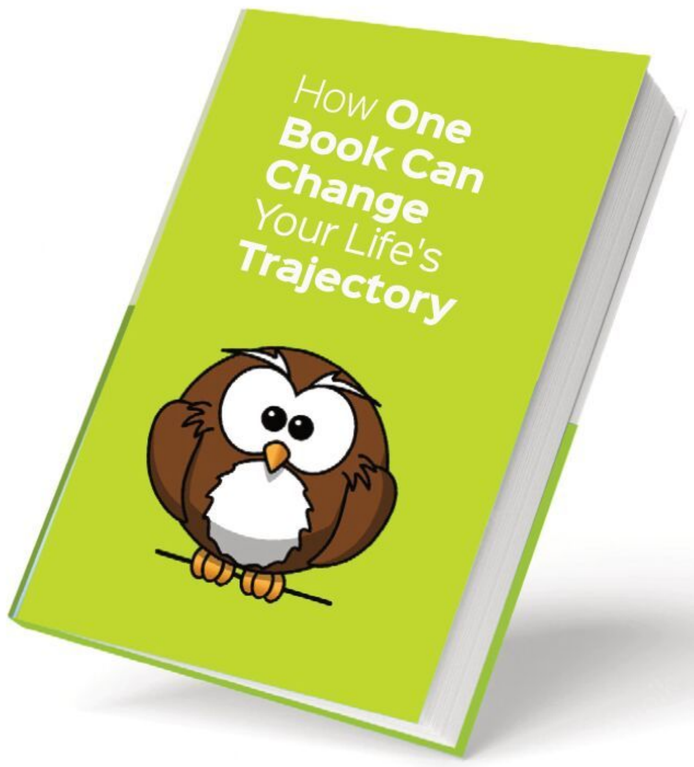
This is yet another reason why the computer vision market will continue to grow exponentially. It will drive Software 2.0. Andrej Karpathy, director of AI at Tesla, is a big proponent of Software 2.0. He notes that with Software 2.0, the computer vision library (for example, OpenCV) could even be auto-tuned on our specific data! END

By: Aaryaa Padhyegurjar

The author is an Industry 4.0 enthusiast at EFY with a keen interest in innovation and research.

This article was originally published in the May 2022 issue of Electronics For You.

How One Book Can Change Your Life's Trajectory



An unexpected e-mail popped up in my mailbox the other day. It read: "You may or may not remember me. We interacted briefly several years ago. I am now trying to contact 'Thaths' Chandrasekhar of Netscape, and a Linux *guru*. The email ID that I have seems defunct. How can I contact him? Please help me." That email was from an intellectual who has great love for the open source world. Read on to know more about this veteran who says he is still learning.

For those who came in late, as the good ol' Phantom comics would say, Netscape (earlier Mozilla) was the American company that created an early, widely-used Internet browser.

In the early 1990s, a Goa University friend, Prof Gurunandan Bhat used this browser to demonstrate to us the first-ever pages we saw on the Internet. It seems like yesterday, but that was almost three decades ago.

To encounter the potential of this "online library on your desktop" was quite something then. But Netscape subsequently lost out to Internet Explorer. In turn, IE succumbed to ... and so the story goes on. But, at one point of time, Netscape indeed had a 90 per cent share of the browser market.

And that was where 'Thaths' worked.

Minor cult figures

During the early days of the Linux/ILUG campaigns across India (this was the late 1990s), the diaspora working in tech firms worldwide were treated as minor cult figures back home. Everyone looked up to them. Thaths was one such figure.

The mail mentioned above came from Prof Parthasarathy, better known in the *desi* FOSS world of the 1990s as 'Doc Partha'. He was quick to say, then and now, "Call me Partha."

In a few days, Thaths was indeed found! But this search helped me to run across Dr Partha himself — after years. No, actually after two full decades. Doc Partha explained the context, and why the search came about: "I got my first book on Linux by begging for it. Thaths sent it to me." This mathematician with a PhD in the subject has focused for over two decades on promoting free and open source software (FOSS) computing.



Doc Partha, professor and Linux evangelist

In his time, he got known for his unique idea of 'Linux in a teaspoon'. "I never stopped; but became addicted to Linux. I'm what you would call a Linux fanatic," he said in an interview with *OSFY* last month. He still sees proprietorial code as "untouchable". No biases, but just a policy for which he explains his rationale.

Says Doc Partha: "I wish to help my students discover mathematics and computer science." He lives in the city of Secunderabad, and adds with some pride: "I share my birthday with George Boole (mathematician, the father of Boolean algebra)."

Doc Partha has a background in discrete mathematics, graphics, graph theory, logic, crypotgraphy, number theory and "everything related to the computer." In all of these fields, he says he finds "plenty of tools on Linux."

Some strong reasons...

There are strong reasons for this, of course.

The sharing system [in the FOSS world], the brotherhood, the fact that nobody's cheating nobody ... or that I'm not going to push something just to make money out of you — these are some of the "absolutely honest and clean" values of GNU/Linux which he finds amazing. "Honest and clean are the right words for Linux," says he.

As he puts it: "If you really want to learn computer science, you will not learn anywhere else other than via Linux. It teaches you a whole lot. Every day."

Rip Van Winkle

When we reminisced recently, both of us jokingly said we felt like Rip Van Winkle. Coming back to civilisation twenty years later, and being surprised to note how the "world has changed." Today, says Partha, everyone swears by Linux. People, particularly those in the academic field, "have taken well to FOSS. They want something reliable."

He argues: "People once thought that if it was free, there must be something wrong with it. Today almost everybody I meet swears by it and says it's either Linux or nothing. Particularly those in the academic field have a lot of respect and admiration for Linux. They want something reliable and honest, which they can copy and freely give to their students."

"Sharing is a very human emotion. Everybody wants to share. Even when sitting with people you do not know in a train, sharing is a human instinct. So why should it not be so with computer science, and with software?" he asks.

How does he see the strengths and weaknesses of the two-decade-old campaign in India? He quips: "I would not say we've failed. But we have not learnt how to sell ourselves. Basically, Linux people are poor sales people."

He remembers Naidu, who worked for AT&T, and brought him a set of magazines and Linux distros. "The first distro I installed with my own hands was Slackware, given to me by Naidu. It was Kernel 1, maybe some 25 years ago. Slackware was a complete no-graphics distro."

We spoke about those who had shaped the FOSS world in India then. Among others, Atul Chitnis, Raj Mathur, Prof G. Nagarjuna, Prakash Advani, expat Indians in Singapore and Australia, and even a Hindu priest who was also a Debian hacker in the US!

He recalls Prof Rajeev Sanyal of IIIT Hyderabad. "I had a CD but could not share it with him, as I didn't have a CD burner (which were rare to come by then). So Prof Rajeev (and others wanting to borrow it) would send their men, and the CD would crisscross Hyderabad because someone or the other needed to share the software. That's how we started it all. It's an unbelievable story. If you want something badly enough, there are people who will help you. I'm happy I went through all this."

> "If you really want to learn computer science, you will not learn anywhere else other than via Linux. It teaches you a whole lot.

Adult education

Was he then one of the first Linux users in the country? "I don't think so," he says with modesty. But he believes he was one of the first Linux learners. "Talk about adult education, and I'm a good example," says he, with a laugh.

He recalls the youngsters who taught him and are now in New Zealand. Doc Partha says his 'teacher' in Linux was Srinivasan, a young man two years junior to his own son.

Punched cards

Dr Partha did his PhD thesis during the punched-card days, from the Institut National Polytechnique de Grenoble, France, located on the France-Italy-Switzerland border. "I have punched cards right here, at home. To write a 200-line FORTRAN project took me one box of punched cards," he recalls.

That was in 1980, after which he worked in China. He's been in the software industry since the 1980s — in corporations, as an independent entrepreneur, and a university teacher. Partha has also taught in France, Macau, Thailand, Indonesia, Germany, Spain, Poland, Nepal, China, and India. Wow! His last assignment was in 2015, in Kathmandu, the scenic city he left after the major earthquake that very year.

In Kathmandu, he taught cryptography for four years. Cryptography's best tools are in Linux, he says.

"In fact, I started life (in computing) with a Multics machine," he notes. Linux follows a POSIX (Portable Operating System of UNIX) standard. This is a family of standards specified by the IEEE Computer Society for maintaining compatibility between operating systems. On its first page, the book outlining this standard acknowledges some 200 contributors. His is the only India-based Indian name there, he says proudly.

'Linux in a Teaspoon'

One of Doc Partha's much-remembered works is called 'Linux in a Teaspoon'. It is an introduction to all aspects of Linux. "It starts by saying, 'Don't use Linux if you think it's not user-friendly, or if you want to play games.' I began that work by giving people ten reasons why they should not use Linux," he says.

Back then, tonnes of bytes of GNU/Linux resources were packed into a single CD. It contained all the stories, links and downloaded material related to Linux. "In those days, the Internet was not popular [or easy to access in countries like India]. So I had to make a CD, and distribute it free of cost to everyone around. Whichever course I conducted, I would distribute the CD free of cost."

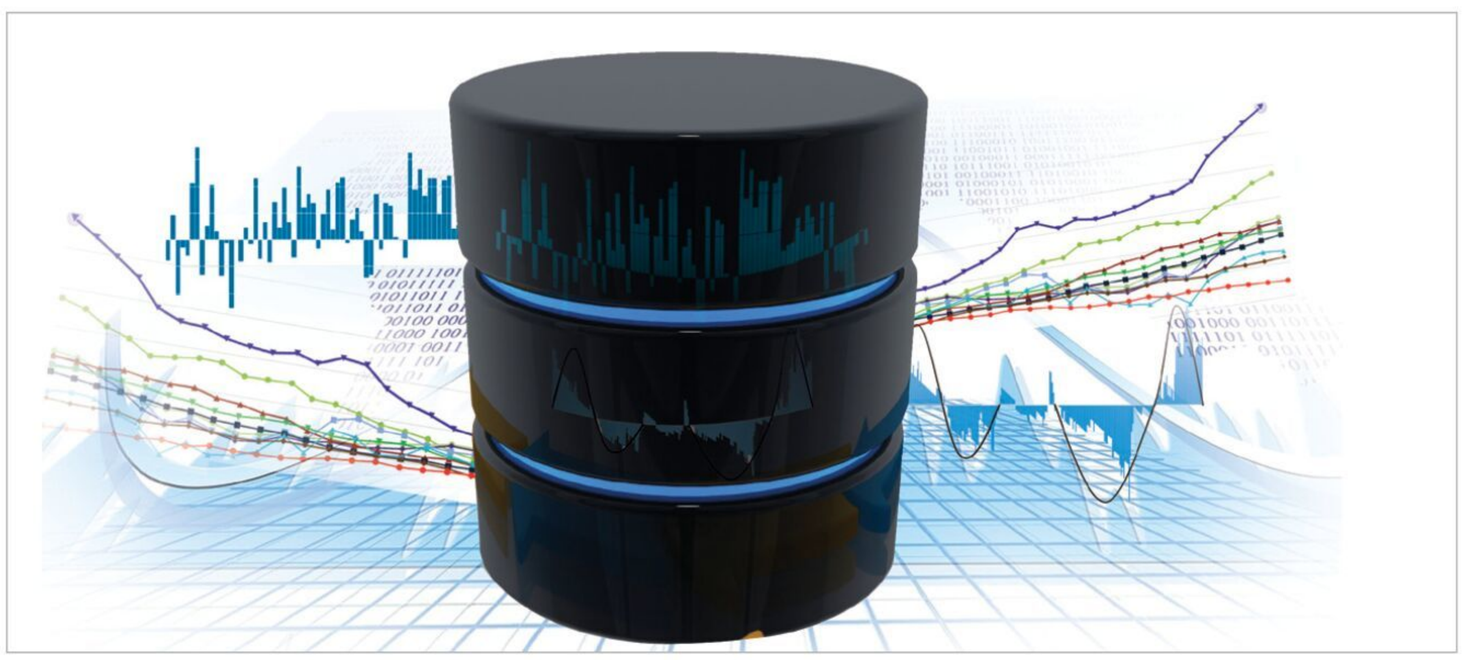
This was a simple-yet-radical idea in its time — collating and freely sharing knowledge, in a way that made it even easier to spread and proliferate it.

Continued on Page....60

FOCUS

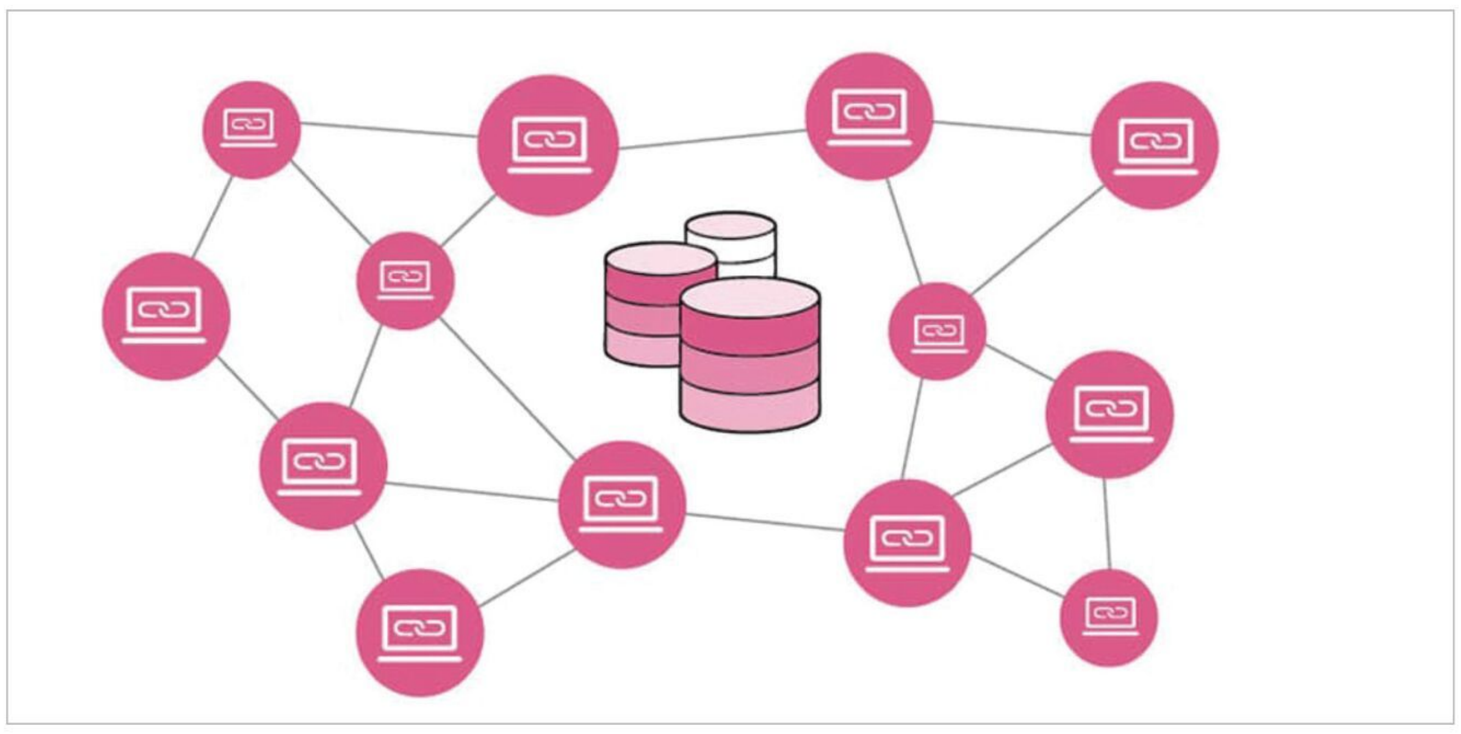
Manage and Optimise your Database

Why We Need Time Series Databases for Site Reliability Engineering



The recent explosion of IoT devices, an increased focus on site reliability engineering for hybrid clouds, real-time trading analysis, etc, have led to a greater spotlight on time series analysis and time series databases. This is because much larger time series data sets are being analysed today. *Read more on page.....56*

Blockchain Based Databases: Decentralised and Secure



In traditional databases, the centralised server maintains the complete set of tables and records. Such servers are subject to malicious attacks including SQL injections, cross-site scripting, and so on. Blockchain based databases are decentralised and are therefore very secure. This article introduces you to some of the best blockchain based databases. *Read more on page.....66*

HeatWave: Making MySQL Smarter

MySQL is a popular and powerful open source relational database management system. The inclusion of HeatWave in MySQL has enhanced its capabilities multifold. Read on to know more. *Read more on page.....61*

YugabyteDB: The Reasons Why It's Popular

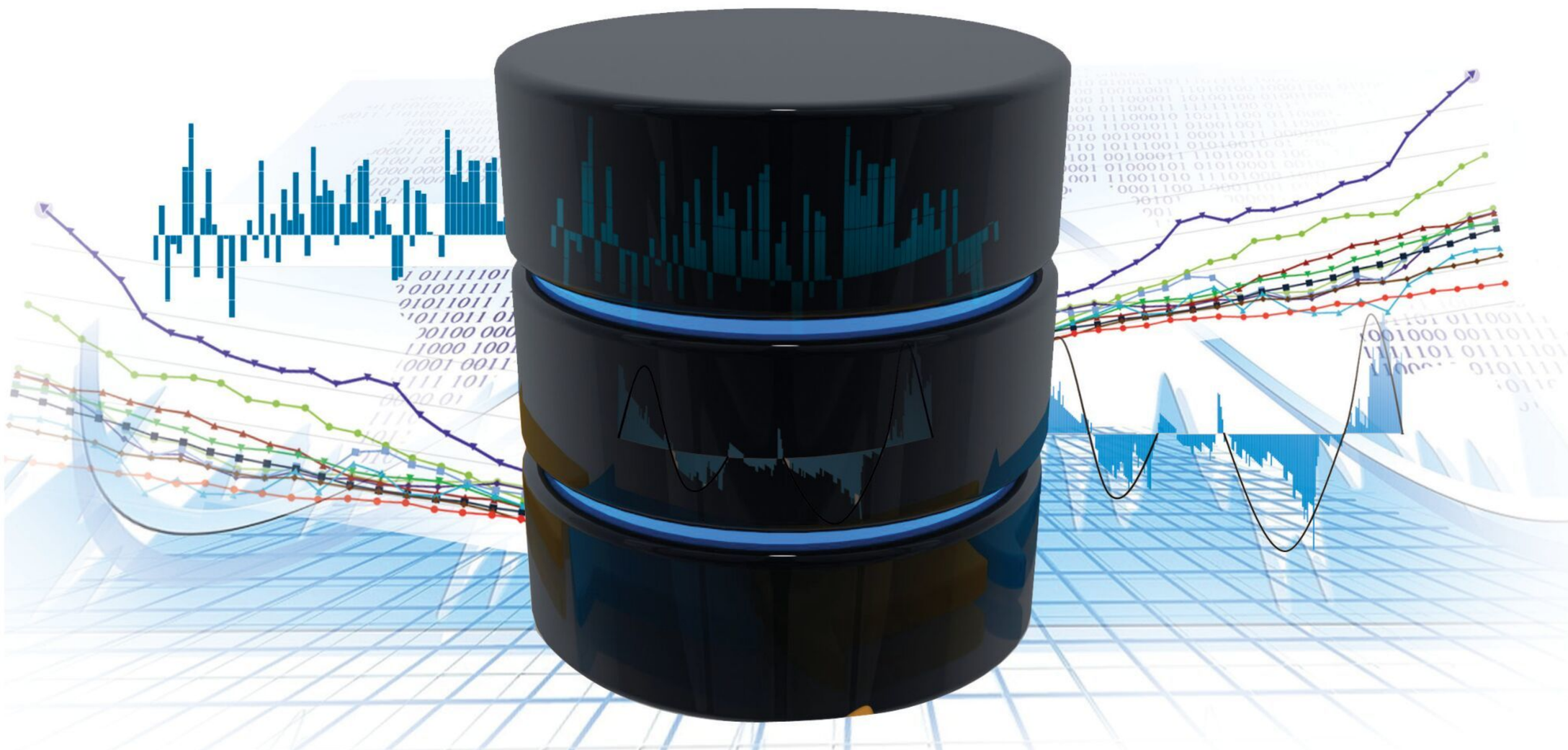
Yugabyte was first developed as closed source commercial software. However, by 2019, its founders realised the power of the open source community and made it available under the Apache 2.0 license. The timing was right as developers were on the lookout for a cloud native database. The adoption and popularity of this database has increased significantly in recent years for a number of reasons. *Read more on page.....69*

Choose the Right Database for Your Project

Selecting the right database is essential for the success of any project. A good data model underpins the overall architecture, provides tremendous insights into the application itself, and builds a solid foundation for a project. There are a number of SQL (RDBMS) and NoSQL based databases available in the market. Which one should you go for? *Read more on page.....71*

Why We Need Time Series Databases for Site Reliability Engineering

The recent explosion of IoT devices, an increased focus on site reliability engineering for hybrid clouds, real-time trading analysis, etc, have led to a greater spotlight on time series analysis and time series databases. This is because much larger time series data sets are being analysed today.



It's not uncommon to deal with petabytes of data today, even when carrying out traditional types of analysis and reporting. Traditional databases, however, do not offer optimal mechanisms to store and retrieve large scale time series data. To meet the demand of time series analysis, new types of databases are emerging.

Real-world use cases

Time series data streams are gathered in many real-world scenarios. The volume of data and the velocity of data generation differ from case to case. Typical scenarios from different fields are described below.

- *Monitoring data gathered as part of site reliability engineering*: Health, performance, and capacity parameters are collected periodically over time from various layers of infrastructure and applications. These time series data streams are analysed for anomalies, fault detection, forecasting, etc. Huge amounts of time series data need to be analysed in real-time to avoid service breakdowns and for quick recovery. This data is also stored and retrieved for processing later, such as capacity forecasting.
- IoT devices and sensors generate continuous streams of data.
- Autonomous trading algorithms continuously collect data on how the markets are changing in order to optimise returns.
- The retail industry collects and monitors supply chain and inventory data to optimise costs.
- Weather forecasting teams continuously collect climate parameters such as temperature, humidity, etc, for predictions.

- Autonomous vehicles continuously collect data about how their environment is changing, adjusting the drive based on weather conditions, engine status, and countless other variables.
- In the medical field, sensors generate time data streams for blood pressure tracking, weight tracking, cholesterol measurements, heart rate monitoring, etc.

There are numerous real-world scenarios where we collect time series data streams. These demand an efficient database for storing and retrieving time series data.

Time series data analysis

Time series data is a sequence of data points collected over time intervals, giving us the ability to track changes over time. Because data points in time series are collected at adjacent time periods, the observations can be correlated. This feature distinguishes time series data from traditional data. Time series data can be useful to help recognise patterns or a trend. Knowing the value of a specific parameter at the current time is quite different than the ability to observe its behaviour over a long time interval. Time series data allows us to measure and analyse change — what has changed in the past, what is changing in the present, and what changes may take place in the future.

Time series data can track changes over milliseconds, days, or even years. Table 1 outlines the typical questions that time series analysis can help to answer.

Typical steps in time series data analysis are:

- Collecting the data and cleaning it
- Visualising with respect to time vs key feature
- Observing the stationarity of the series
- Developing charts to understand the nature of the data
- Model building such as AR, MA, ARMA and ARIMA
- Extracting insights from the predictions

There are three components of time series analysis — trend, seasonality and residual analysis.

Trend: This indicates the direction in which the data is moving over a period of time.

Seasonality: Seasonality is about periodic behaviour — spikes or drops caused by different factors like:

- Naturally occurring events like weather fluctuations
- Business or administrative procedures like the start or end of a fiscal year
- Social and cultural behaviour like holidays or festivals
- Calendar events, like the number of Mondays per month or holidays that change every year

Residual analysis: These are the irregular fluctuations that cannot be predicted using trend or seasonality analysis.

An observed data stream could be additive (trend + seasonality + residual) or multiplicative (trend * seasonality * residual).

Once these components are identified, models are built to understand time series and check for anomalies, forecasting and correlations. For time series data modelling, AR, MA, ARMA and ARIMA algorithms are widely adopted. Many other advanced AI/ML algorithms are being proposed for better evaluation.

Time series databases

A time series database (TSDB) is a database optimised for time-stamped or time series data. Time series data is simply measurements or events that are tracked, monitored, down sampled, and aggregated over time. These could be server metrics, application performance monitoring, network data, sensor data, events, clicks, trades in a market, and many other types of analytics data.

Looking back 10 years, the amount of data that was once collected in 10 minutes for some very active systems is now generated every second. To process these high volumes, we need different tools and approaches.

Table 1

Category	Typical questions to be addressed
Prognostication	What are the short- and long-term trends for a measurement or group of measurements?
Introspection	How do several measurements correlate over a period of time?
Prediction	How do I build a machine learning model based on the temporal behaviour of many measurements correlated to externally known facts?
Introspection	Have similar patterns of measurements preceded similar events?
Diagnosis	What measurements might indicate the cause of some event, such as a system failure?
Forecasting	How many more servers will be needed for handling next quarter's workload?
Segmentation	How to divide a data stream into a sequence of discrete segments in order to reveal the underlying properties of its source?

To design an optimal TSDB, we must analyse the properties of time series data, and the demands of time series analysis applications. The typical characteristics of time series data and its use cases are:

- Time series is a sequence of values, each with a time value indicating when the value was recorded.
- Time series data entries are rarely amended.
- Time series data is often retrieved by reading a contiguous sequence of samples.
- Most of the time, we collect and store multiple time series. Queries to retrieve data from one or a few time series for a particular time range are very common.
- The volume and velocity of time series data is very high.
- Both long-term and short-term trends in the time series are very important for analysis.
- Summarising or aggregating high volume time series data sets is a very basic requirement.
- Traditional DB operations such as searching, sorting, joining tables, etc, are not required.

Properties that make time series data very different from other data workloads are data life cycle management, summarisation, and large range scans of many records. TSDB is designed to simplify and strengthen the process for real-world time series applications.

Storing time series data in flat files limits its utility. Data will outgrow these and the access is inefficient. Traditional RDBMS databases are not designed from the ground up for time series data storage. They will not scale well to handle huge volumes of time series data. Also, the schema is not appropriate. Getting a good performance for time series from an SQL database requires significant customisation and configuration. Without that, unless you're working with a very small data set, an SQL-based database will simply not work properly. A NoSQL non-relational database is preferred because it scales well and efficiently to enable rapid queries based on time range.

A TSDB is optimised for best performance for queries based on a range of time. New NoSQL non-relational databases come with considerable advantages (like flexibility and performance) over traditional relational databases (RDBMS) for this purpose. NoSQL databases and relational databases share the same basic goals: to store and retrieve data and to coordinate changes. The difference is that NoSQL databases trade away some of the capabilities of relational databases in order to improve scalability. The benefits of making this trade include greater simplicity in the NoSQL database, the ability to handle semi-structured and denormalised data and, potentially, much higher scalability for the system.

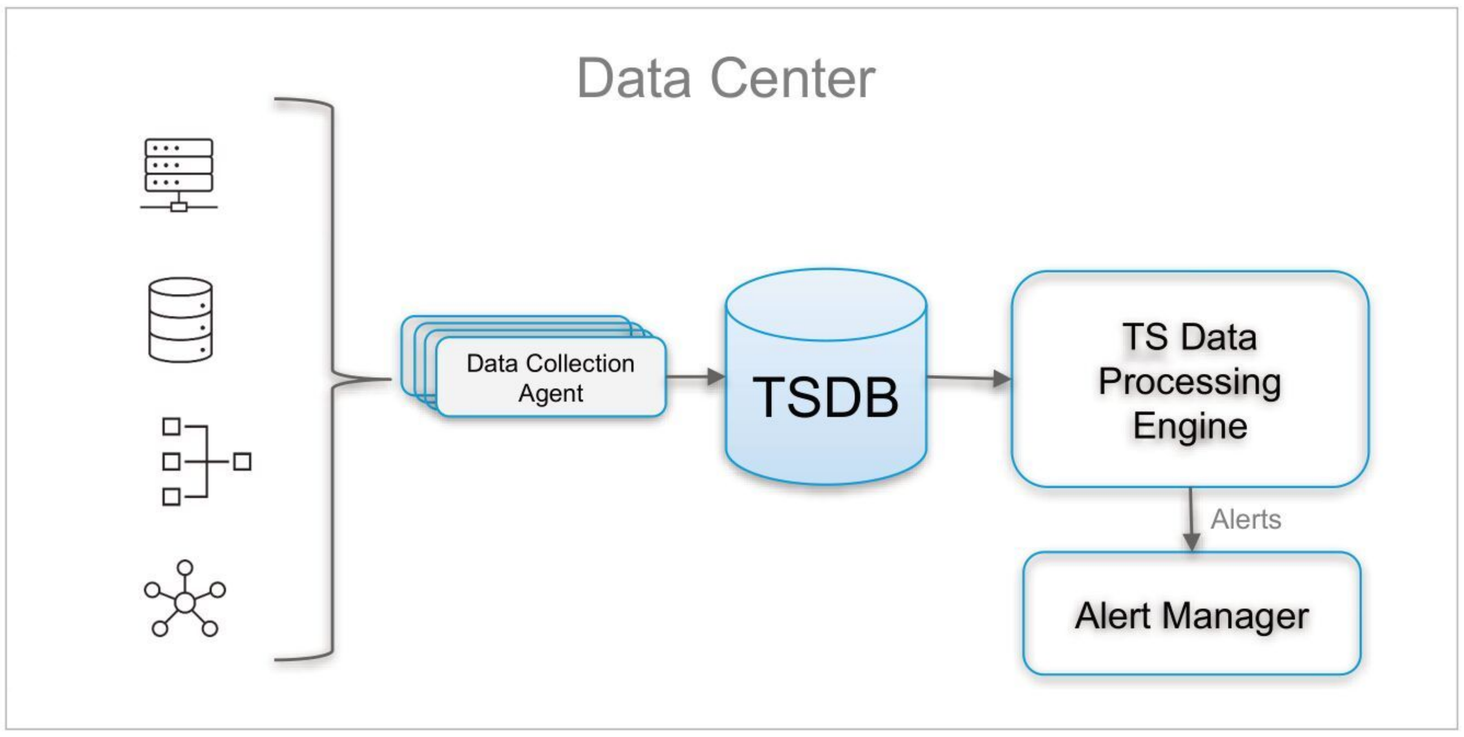


Figure 1: Alerting engine based on time series data analysis

At very large scales, time-based queries can be implemented as large, contiguous scans that are very efficient if the data is stored appropriately in a time series database. And if the amount of data is very large, a non-relational TSDB in a NoSQL system is typically needed to provide sufficient scalability.

Non-relational time series databases enable discovery of patterns in time series data, long-term trends, and correlations between data representing different types of events. The time ranges of interest extend in both directions. In addition to the very short time-range queries, long-term histories for time series data are needed, especially to discover complex trends.

Time series databases have key architectural design properties that make them very different from other databases. These include time-stamp data storage and compression, data life cycle management, data summarisation, the ability to handle large time series-dependent scans of many records, and time series aware queries.

For example, with a time series database, it is common to request a summary of data over a large time period. This requires going over a range of data points to perform computations like a percentile increase this month of a metric over the same period in the last six months, summarised by month. This kind of workload is very difficult to optimise for with a distributed key value store. TSDBs are optimised for exactly this use case and can give millisecond-level responses over months of data. Here is another example. With time series databases, it's common to keep high precision data around for a short period of time. This data is aggregated and down sampled into long-term trend data. This means that for every data point that goes into the database, it will have to be deleted after its period of time is up. This kind of data life cycle management is difficult for application developers to implement in regular databases. They must devise schemes

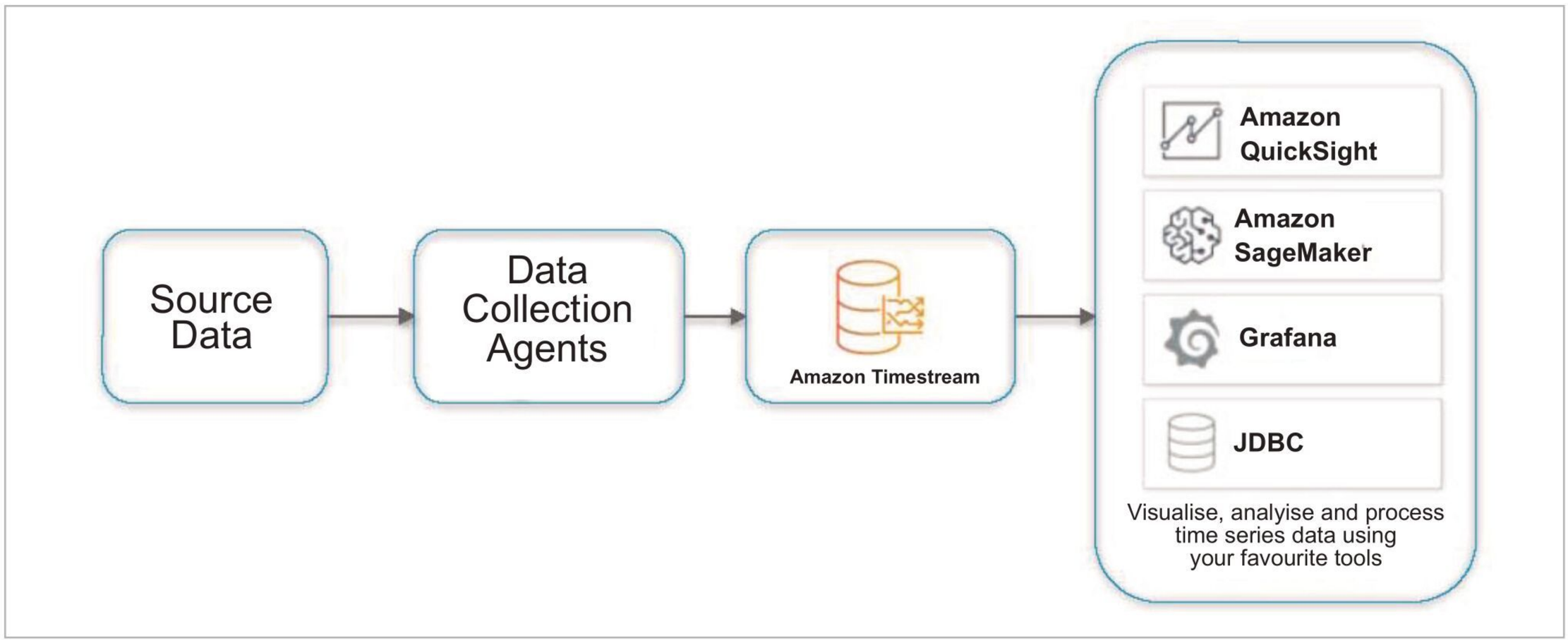


Figure 2: Time series analytics engine on AWS Cloud

for cheaply evicting large sets of data and constantly summarising that data at scale. With a time series database, this functionality is provided out-of-the-box.

Since time series data comes in time order and is typically collected in real-time, time series databases are immutable and append-only to accommodate extremely high volumes of data. This append-only property distinguishes time series databases from relational databases, which are optimised for transactions but only accommodate lower ingest volumes. In general, depending on their particular use case, NoSQL databases will trade off the ACID principles for a BASE model (whose principles are basic availability, soft state and eventual consistency). For example, one individual point in a time series is fairly useless in isolation, and the important thing is the trend in total.

Alerts based on time series data analysis for site reliability

Time series data models are very common in site reliability engineering. Time series analysis is used to monitor system health, performance, anomaly detection, security threat detection, inventory forecasting, etc. Figure 1 shows a typical alerting mechanism based on analysing time series data collected from different components.

Modern data centres are complex systems with a variety of operations and analytics taking place around the clock. Multiple teams need access at the same time, which requires coordination. In order to optimise resource use and manage workloads, systems administrators monitor a huge number of parameters with frequent measurements for a fine-grained view. For example, data on CPU usage, memory residency, IO activity, levels of disk storage, and many other parameters are all useful to collect as time series.

Once these data sets are recorded as time series, data centre operations teams can reconstruct the circumstances that lead to outages, plan upgrades by looking at trends, or even detect many kinds of security intrusions by noticing changes in the volume and patterns of data transfer between servers and the outside world.

Open source TSDBs

Time series databases are the fastest growing segment in the database industry. There are many commercial and open source time series databases available. A few well-known open source time series databases are listed below:

- **InfluxDB** is one of the most popular time series open source databases, and is written in Go. It has been designed to provide a highly scalable data ingestion and storage engine. It is very efficient at collecting, storing, querying, visualising, and taking action on streams of time series data, events, and metrics in real-time. It uses InfluxQL, which is very similar to a structured query language, for interacting with data.
- **Prometheus** is an open source monitoring solution used to understand insights from metrics data and send the necessary alerts. It has a local on-disk time-series database that stores data in a custom format on disk. It provides a functional query language called PromQL.
- **TimescaleDB** is an open source relational database that makes SQL scalable for time series data. This database is built on PostgreSQL.
- **Graphite** is an all-in-one solution for storing and efficiently visualising real-time time series data. Graphite can store time series data and render graphs on demand. To collect data, we can use tools such as collectd, Ganglia, Sensu, telegraf, etc.

- **QuestDB** is a relational column-oriented database that can perform real-time analytics on time series data. It works with SQL and some extensions to create a relational model for time series data. It supports relational and time-series joins, which helps in correlating the data.
- **OpenTSDB** is a scalable time series database that has been written on top of HBase. It is capable of storing trillions of data points at millions of writes per second. It has a time-series daemon (TSD) and command-line utilities. TSD is responsible for storing data in or retrieving it from HBase. You can talk to TSD using HTTP API, telnet, or a simple built-in GUI. You need tools like flume, collectd, vacuumetrix, etc, to collect data from various sources into OpenTSDB.

Cloud native TSDBs

Cloud hyperscalers like Azure, AWS and Google offer time series databases and analytics services as part of their cloud portfolio. AWS Timestream is a serverless time series database service that is fast and scalable. It is used majorly for IoT applications to store trillions of events in a day and is 1000 times faster with 1/10th the cost of relational databases. Using its purpose-built query engine, you can query recent and historical data simultaneously. It provides multiple built-in functions to analyse time series data to find useful insights.

Microsoft Azure Time Series Insights provides a time series analytics engine. For data ingestion there are Azure IoT Hub and Event Hub services. To analyse cloud infrastructure and time series streams, these cloud vendors offers a range of native tools such as AWS CloudWatch, Azure Monitor, Amazon Kinesis, etc. END

By: Narasimha Sekhar Kakaraparthi

The author has more than 27 years of software industry experience in product development in the hybrid cloud, telecommunications, and manufacturing domains. He is currently working as principal architect for virtual desktops and end user services at Wipro Technologies. He is a senior member of Wipro's eminent architects group called Distinguished Member of Technical Staff (DMTS).

Disclaimer: *The views expressed in this article are that of the author, and Wipro does not subscribe to the substance, veracity or truthfulness of the said opinion.*

Continued from Page....54

His idea got written about in newspapers in Bangalore (as the city was known then), Karachi, and elsewhere. He recalls that a news story in the *Deccan Herald* drew attention to his contributions, as did another in *The Dawn.*

He then started getting responses from a series of obviously foreign names. He learnt why this was happening only when he asked what had triggered their interest — it was the newspaper articles.

Hacking for learning

Doc Partha's ebook on LaTeX — the free and open source software system for document preparation — has been used in more than 65 countries, spanning six continents. His goal is to promote "hacking for learning" through his books and publications, which he shares under a Creative Commons (CC-BY-SA) license. The latest version of this LaTeX generated document is available via *https://drpartha.org.in/*.

He takes pride in being a "regular supporter, promoter and practitioner of FOSS," and has contributed to international efforts to build this volunteer-driven system. His contributions feature in major GNU/Linux 'distros' (or distributions) worldwide, and have been translated into nine languages — Japanese, Dutch, French, Turkish, Russian, Spanish, German, Czech and Bulgarian.

Among other feathers in his cap are being a senior member of IEEE (USA) and a life fellow of IETE (India). He was twice decorated by the IEEE (USA) with international awards, for outstanding contributions to the profession.

But that's not all. In his spare time, he provides free advice and technical assistance for making handicapped-accessible buildings, structures and facilities. He is a serious promoter of eco-friendly, vermiculture-based organic farming, and an active campaigner and volunteer for organ donation.

Doc Partha has authored several research publications, popular articles and, in their time, educational CD-ROMs. He was an editor of the international journal 'Engineering Applications of Artificial Intelligence' (Elsivier, UK), and a reviewer for the Practex Journal, published by the TeX Users Group.

He recalls his "begging" for a book on Linux, by writing to people whom he "didn't even know." The book that Thaths sent him is still on his table. "I'm still learning from this book," says the 72-year old.

"It has been a long and interesting journey. Even today, I keep learning something new every day," he notes. END

By: Frederick Noronha

The author is an independent journalist and publisher, and has written on FOSS for almost two decades.

HeatWave: Making MySQL Smarter

MySQL is a popular and powerful open source relational database management system. The inclusion of HeatWave in MySQL has enhanced its capabilities multifold. Read on to know more.



Oracle MySQL HeatWave (formerly called MySQL database service analytics) is an in-memory query accelerator developed at Oracle Labs, and provisions HeatWave clusters for accelerated processing of analytic queries. It is a very powerful service hosted on the cloud, eliminating the need for a separate analytics database and ETL processes.

Traditional analytics solutions: The challenges

Generally, customers have online transaction processing (OLTP) applications, which are connected to an OLTP database like MySQL. If users have to execute a query, they need to move their data to an online analytical processing (OLAP) database. This is done through extraction, transfer and load (ETL), and typically requires a database administrator's (DBA) involvement. A DBA assists in migrating data from OLTP to OLAP, performs the query execution and sends the results back to OLTP. However, there are a few challenges existing solutions face:

- Running analytic queries against an OLTP database is always slow.
- The ETL processes with separate databases for OLTP and OLAP are lengthy.
- The data that is being processed is not up-to-date.
- The solutions are complex and expensive, as you need two databases and have to pay for two different services along with an experienced DBA.

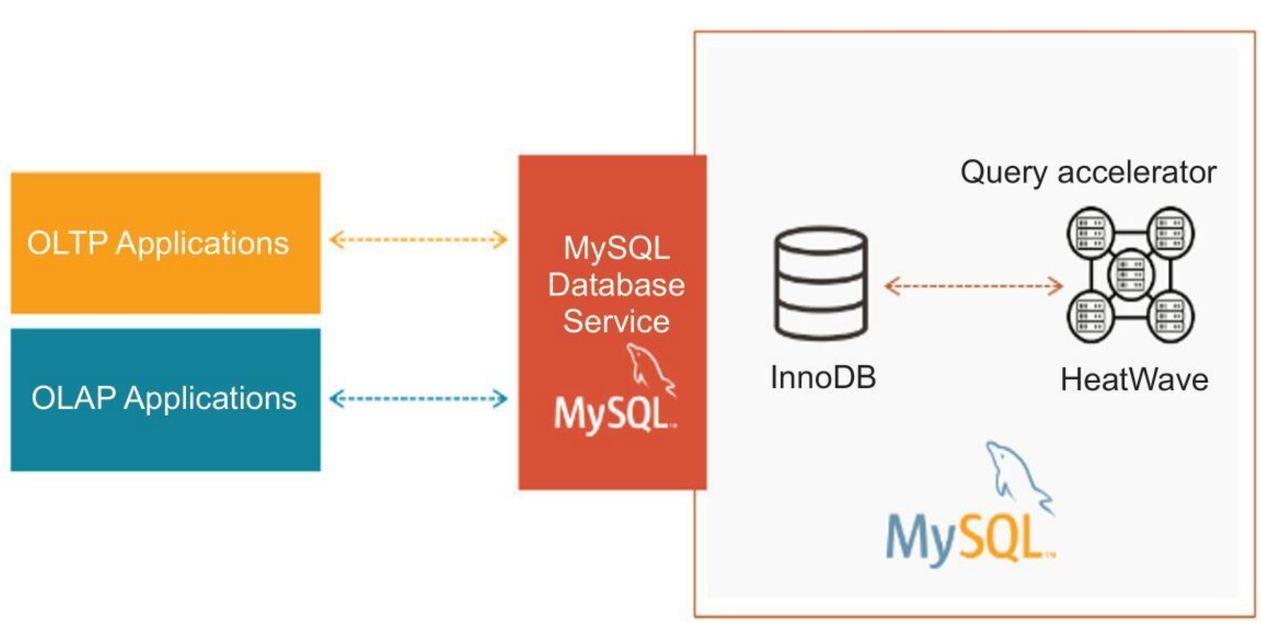


Figure 1: Architecture for HeatWave integration

Oracle MySQL HeatWave

MySQL HeatWave has emerged as a single solution to all the problems mentioned above. Both OLTP and OLAP are integrated adequately, and a built-in real-time query accelerator has been

specifically designed for this service. This accelerator is available only in Oracle cloud infrastructure (OCI) and optimised only for it. It always ensures that real-time data is being processed whenever you run the queries, and provides an overall better price to performance ratio. The query processing time and the performance is 5,400 times faster than that of its competition.

Figure 1 shows how HeatWave is integrated into InnoDB. As can be seen, MySQL has InnoDB and HeatWave built in. HeatWave has an Autopilot advisor that helps get recommendations, which makes queries run faster. Let us now move on to a more detailed architecture for better clarity.

As can be seen in Figure 2, the OLAP queries get optimised by the optimiser present in the MySQL database service itself. When the HeatWave cluster is enabled, queries that meet certain prerequisites are automatically offloaded from the MySQL DB system to the HeatWave cluster for accelerated execution without any user intervention. The HeatWave node gives recommendations and the query estimation time, and then executes these within a very short time compared to a regular OLAP processor. It then sends the results back to the MySQL database service.

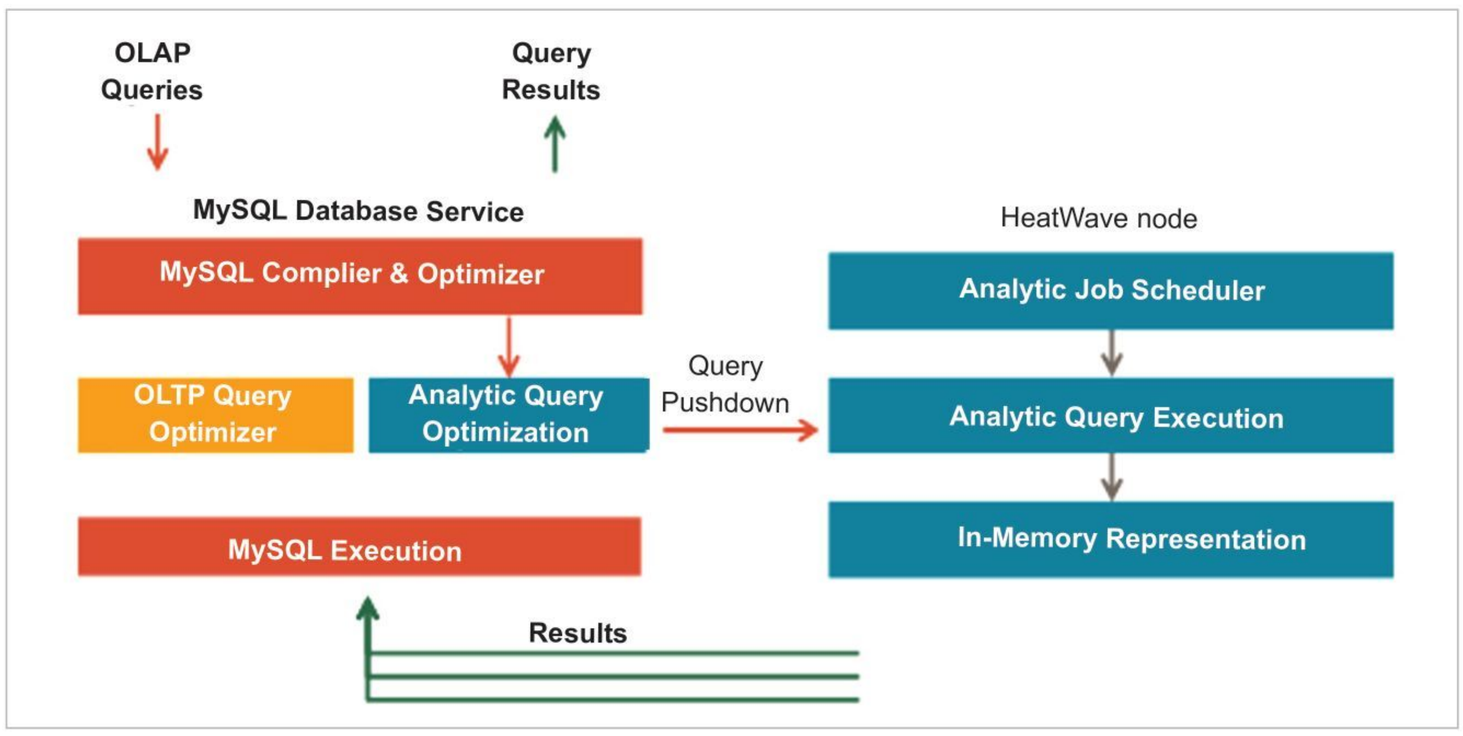


Figure 2: Detailed HeatWave architecture

How is MySQL Autopilot making a difference?

MySQL Labs has been working on Autopilot for at least a decade to enhance its machine learning (ML) capabilities. This optimiser has been increasingly made intelligent with the integration of ML algorithms leading to an excellent performance. It gives smart recommendations based on the data, and only uses 0.1 per cent of the data sample to come up with a query estimation. The query performance at scale is also high, improving the usability of the system. The system is used effectively, especially with respect to data placement or managing change propagation.

Oracle MySQL Autopilot: How it works

MySQL Autopilot helps in four major domains.

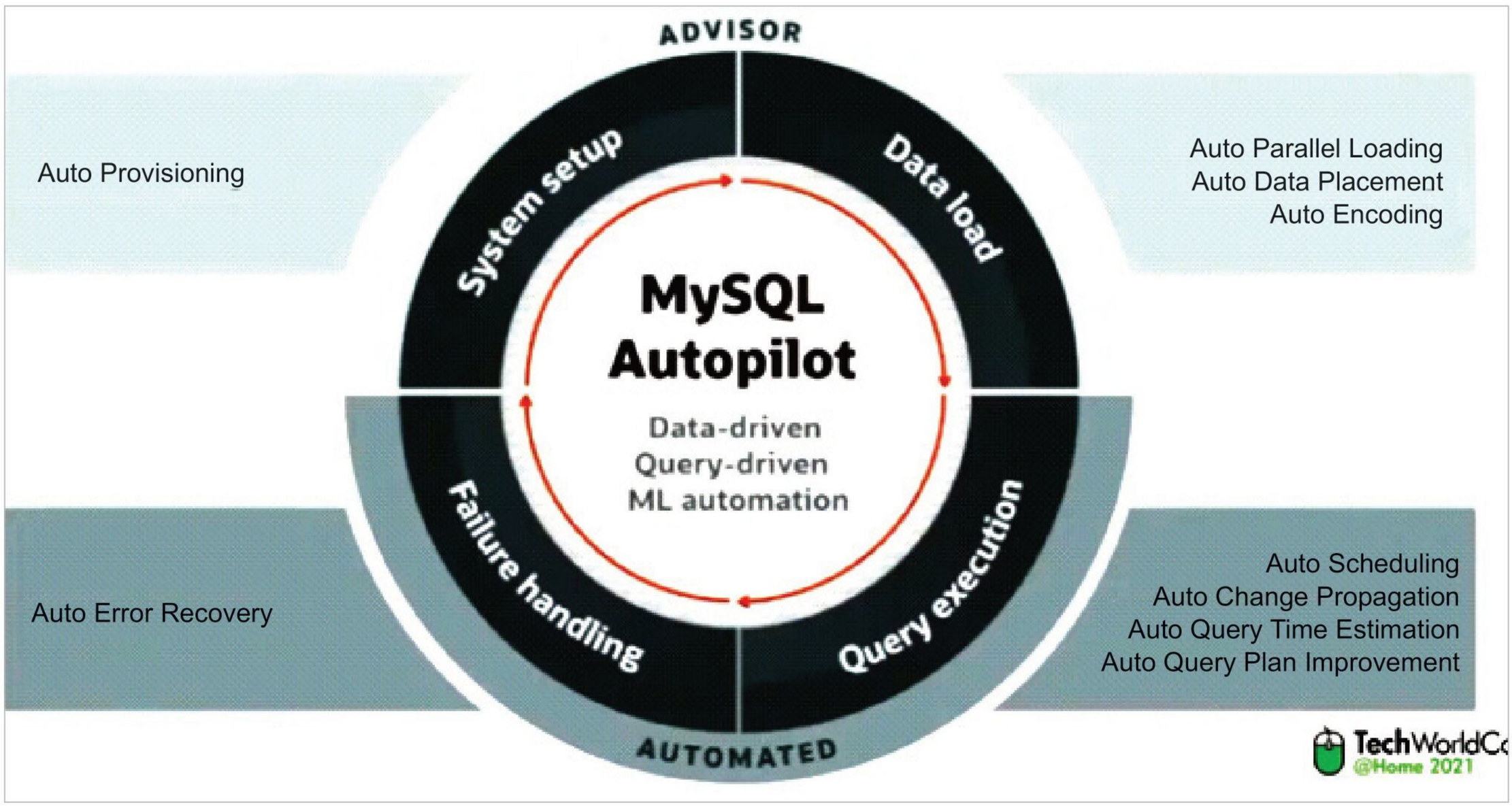


Figure 3: MySQL Autopilot

System setup: This is essentially the first step. You often need assistance to figure out how many nodes are needed to query across a certain amount of data. That is when auto provisioning comes into the picture. You have to load your data, and the auto provisioning advisor will tell how many nodes you require based on the number of tables and primary keys.

Data load: The auto data load advisor consists of auto parallel loading, auto data placement, and auto encoding. Auto parallel loading is designed intelligently to tell you the number of threads you require to load a certain piece of data in parallel. Auto data placement predicts the column on which the table should be partitioned in the memory in order to achieve the best performance for queries. It also predicts the expected gain in query performance with a new column recommendation.

Query execution: This consists of auto scheduling, auto change propagation, auto query time estimation, and auto query plan improvement. Here again the query plan advisor simply executes a query and improves the execution plan for future queries. It works in an efficient way to optimise query plans via the machine learning algorithms. This allows real-time optimisation of queries because the query optimiser learns and adapts during the database usage time. Auto query time estimation is based on an analytical model of Autopilot's query processing algorithms. Using a data driven approach improves performance over time as more queries are run. In auto scheduling, short queries are prioritised over long running queries. The system reduces the wait time for shorter queries without changing total execution time. Auto change propagation intelligently determines the optimal time for changes in the MySQL database to be propagated.

Failure handling: This consists of auto error recovery. Whenever there is a HeatWave error, the recovery is automatically taken care of as it is being continuously monitored by the Autopilot advisor. A system failure is handled by restarting the failed node and the HeatWave error is fixed by restarting the failed HeatWave process.

To wrap things up, it is essential to note that MySQL Autopilot supports machine learning based automation. It provides functionalities such as data loading, query execution and failure handling while further improving the performance of HeatWave itself. It is faster and more economical than its competitors. However, remember to make a clear assessment of the requirements before choosing a service as the system's productivity will rely heavily upon that. END

MySQL Enterprise Edition: A fully managed MySQL database service

MySQL Enterprise Edition comes with a whole lot of functionalities such as backup, replication, monitoring and many other tools associated with it. Any on-premise customer using this version of MySQL does not have to worry about the latest updates for their databases as all new updates to the Enterprise Edition are added automatically. This not only saves time but also lets customers focus more on their applications than on maintaining the database and its version.

	Automation	MySQL On-Premises	MySQL Database Service
Database	Scaling	✗	✓
	Backup	✗	✓
	Security Patch & Upgrade	✗	✓
	Provision & Configure	✗	✓
OS	OS Security Patch & Upgrade	✗	✓
	OS Installation	✗	✓
Server	Hardware Purchase & Maintenance	✗	✓
Storage	Storage Purchase & Maintenance	✗	✓
Data Center	Rack & Space	✗	✓
	Power, HVAC, Networking	✗	✓

Figure 4: A fully managed MySQL database service

By: Jothir Ganesan

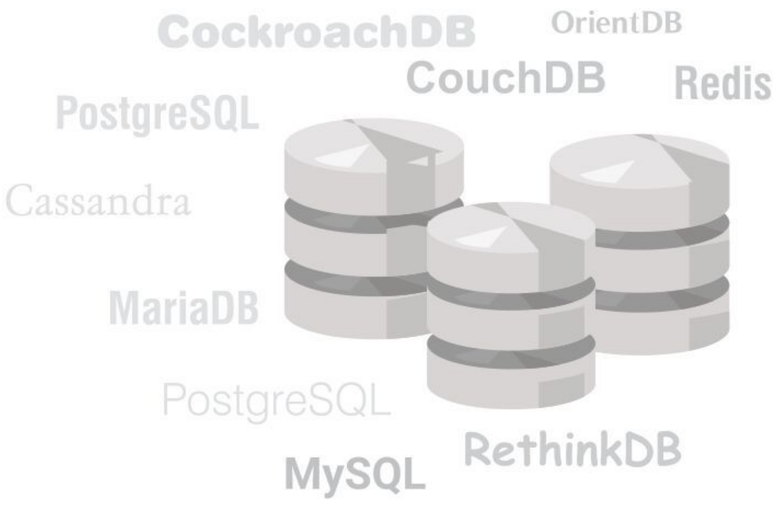
The author is the site reliability engineering development manager at Oracle. This article is based on the tech talk he gave at OSI 2021.

Transcribed and curated by Laveesh Kocher

THE COMPLETE MAGAZINE ON OPEN SOURCE
OpenSource ForYou
The latest from the Open Source world is here.
OpenSourceForU.com
Join the community at facebook.com/opensourceforu
Follow us on Twitter @OpenSourceForU

Are You Familiar with These Popular Open Source Databases?

As data grows by leaps and bounds, its organisation becomes all-important. This article briefly describes the most popular databases being used by software development teams today.



In software systems, data is stored in an organised format and can be accessed electronically using a database. As data has become a very important asset today, it is very important for us to have a basic knowledge about the different databases in use today.

The first database that we will be looking at is MySQL.

MySQL

MySQL is one of the most widely used open source database management systems. Owned by Oracle Corporation, it works on most major operating systems like Windows, MacOS, Linux, etc. One big benefit of MySQL is that it works really well for small and large applications.

Advantages

- It works well with a variety of operating systems and programming languages like PHP, C, C++, Perl, etc.
- It is open source and free.
- It supports a huge size of data of about 8 million terabytes.
- MySQL is customisable as it is open source.
- It is also much faster than other databases.

To install and get started with MySQL on your Ubuntu based computer, use the command given below:

```
$sudo apt update
$sudo apt install mysql-server
$sudo systemctl start mysql.service
```

MariaDB

MariaDB is popular primarily because of its good performance and compatibility with MySQL. It supports relational databases. The developers of MySQL have built MariaDB and guarantee that it is going to stay open sourced. This popular database server is part of most major cloud offerings today, and gives great importance to stability and performance. MariaDB has recently added clustering techniques using the Galera cluster, and is also compatible with Oracle databases.

Advantages

- Easy installation
- Supports operation on Big Data
- High scalability
- Easy import of data

To install and get started with MariaDB on your Ubuntu based computer, use the command given below:

```
$sudo apt update
$sudo apt-get install mariadb-server
$sudo systemctl start mariadb
```

RethinkDB

RethinkDB is an open source, free, distributed, and document based database server. This NoSQL database has been developed by Rethink, and can store JSON files with schemas that are dynamic. More importantly, real-time updates for query results can be pushed for applications to use. Since it is a distributed database, it is highly scalable. It has many automatic functions, making it a highly available database. As it is a popular database today, it is important that we learn how to use it.

Advantages

- This is basically an open source database that can be used for Web based applications.
- It is easy to scale because it's a distributed database.
- It has many automatic functions with a high availability.
- This NoSQL database is JSON dynamic document based.

The following commands can be helpful for using RethinkDB on your Ubuntu machine:

```
Get required packages source
 /etc/lsb-release && echo" deb https://
download.rethinkdb.com/repository/ubuntu-
$DISTRIB_CODENAME $DISTRIB_CODENAME main"
| sudo tee /etc/apt/sources.list.d/
rethinkdb.list

Get required repositories
$wget -qO- https://download.rethinkdb.
com/repository/raw/pubkey.gpg | sudo apt-
key add
$sudo apt update
$sudo apt-get install rethinkdb
$sudo systemctl start rethinkdb
```

OrientDB

OrientDB is a NoSQL and Java based open source database management system. This multi-model database service system supports all sorts of data

like documents, dictionaries, objects, and graphs. It stores the relationships in the form of a graph database.

The following commands can be helpful for using OrientDB on your Ubuntu machine:

```
$sudo apt-get update
$wget -O orientdb-community-2.2.20.tar.gz http://orientdb.com/download.php?file=orientdb-community-2.2.20.tar.gz&os=linux
$tar -zxvf orientdb-community-2.2.20.tar.gz
$sudo mv ~/orientdb-community-2.2.20 /opt/orientdb
```

CouchDB

CouchDB is an important NoSQL based open source database server developed in Erlang. It uses many protocols and formats to transfer, process and share data. JSON files are used to store the data, MapReduce is used as the base and JavaScript is used as the querying language.

Advantages

- It can store any kind of data.
- MapReduce helps in increasing its efficiency.
- The overall structure is very simple.
- Indexing and retrieval is fast.

The following commands can help you to use CouchDB on your Ubuntu machine:

```
$echo “deb https://apache.bintray.com/couchdb-deb focal main” >> /etc/apt/sources.list
$sudo apt-get update
$sudo apt install apache2 couchdb -y
```

Firebird

Firebird is an open source database management system that mainly works on relational databases. It is compatible with all operating systems like Linux, Windows and MacOS. It was originally forked from the Interbase repository, which was also an open source database.

Advantages

- The functionality of the database is not limited.
- It is a very stable and powerful database.
- The configuration and usage of the database is much simpler than other databases.

The following commands can help you use Firebird on your Ubuntu machine:

```
$sudo apt-get update
$sudo apt-get install firebird2.5-superclassic
```

Cassandra

Cassandra is owned and developed by Apache. This highly scalable, distributed, high performance database can handle large amounts of data and works really well. As it is distributed among many servers, it has no single point of failure. It is basically a NoSQL database, which implies it is not a relational database.

Advantages

- High performance
- High scalability
- Peer-to-peer architecture is used instead of master slave architecture

The following commands can be helpful in using Firebird on your Ubuntu machine:

```
$curl https://www.apache.org/dist/cassandra/KEYS | sudo apt-key add -
$sudo apt-get update
$sudo apt-get install cassandra
$sudo systemctl enable cassandra
$sudo systemctl start cassandra
```

PostgreSQL

Today PostgreSQL is one of the most popular open source relational database management systems. It is easily extensible and, at the same time, is compliant with SQL. More than 30 years of active development has gone into this DBMS.

Advantages

- In Postgres we can store a wider variety of data compared to MySQL.
- It is largely compliant with the SQL standard.
- It is also highly expandable.

The following commands can be helpful for using PostgreSQL on your Ubuntu machine:

```
$sudo apt-get update
$sudo apt apt install postgresql postgresql-contrib
```

CockroachDB

CockroachDB is a database built for reliability, i.e., it can survive any kind of adverse situation (just like cockroaches can survive any disastrous situation and multiply). This database can handle large amounts of data. Multicluster databases can also be built.

Advantages

- Deployment is easy
- High consistency
- Transactions are distributed
- Availability is high
- It is also compatible with SQL

Redis

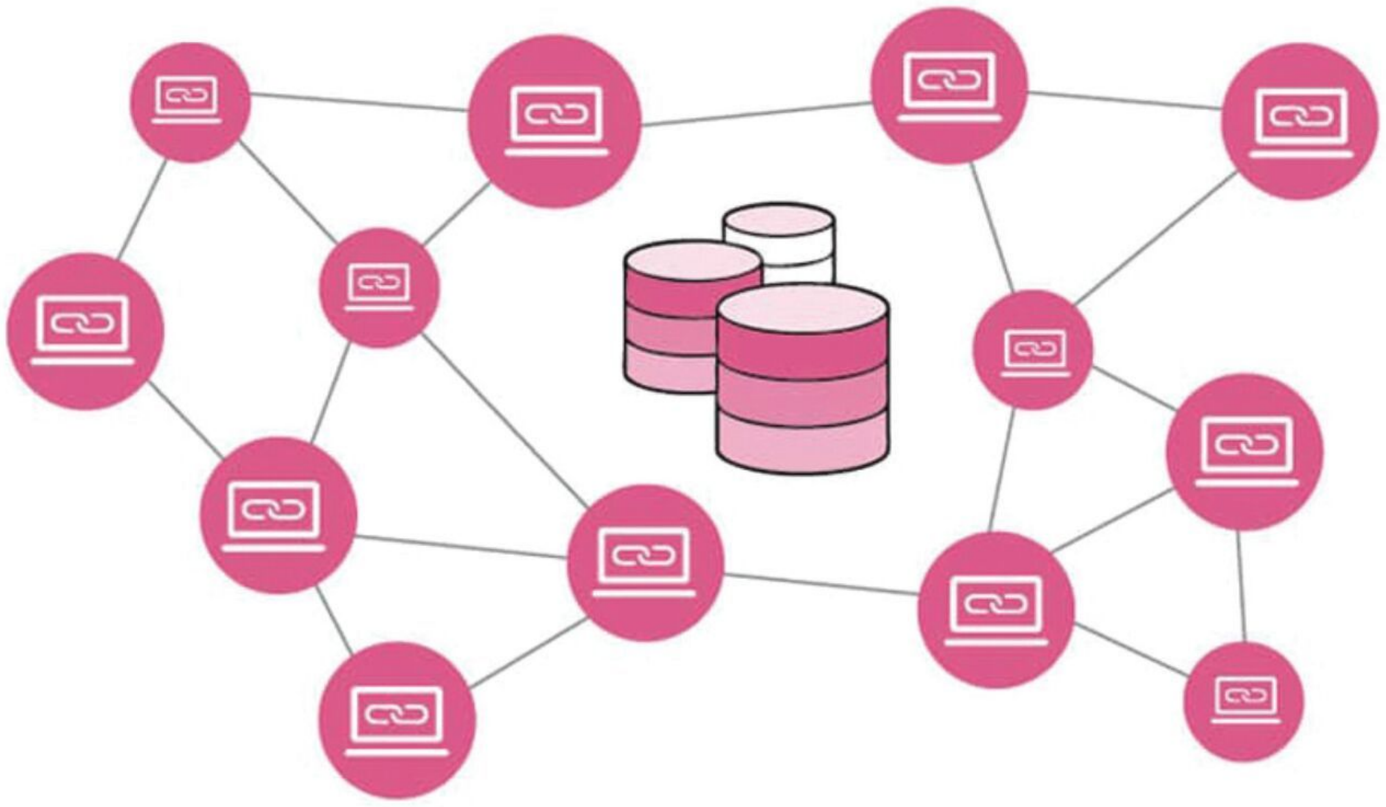
Redis is an open source, key value based data storage database. This NoSQL database is really handy because it uses different keys of various types.

We’ve gone through the most well-known and popular open source database management systems that are being used to store and manage data. Learning about these different databases can be a lot of fun. Try the different options, and use the one that fits your requirements the best. Also, do check out the official documentation of these databases. END

By: Jishnu Saurav Mittapalli

The author is a full stack Web developer, interested in modern technologies like artificial intelligence and machine learning. He is currently working in the field of natural language processing.

Blockchain Based Databases: Decentralised and Secure



In traditional databases, the centralised server maintains the complete set of tables and records. Such servers are subject to malicious attacks including SQL injections, cross-site scripting, and so on. Blockchain based databases are decentralised and are therefore very secure. This article introduces you to some of the best blockchain based databases.

Almost all government and corporate services are today available on digital platforms and smart gadgets. These online services and platforms are vulnerable to cyber-attacks and data leakage issues. This is because centralised database servers can be hacked by third parties and malicious traffic.

To meet this security challenge, blockchain based technologies are under development for multiple applications including e-governance, Internet of Things (IoT), online games, smart health services, supply chain management, and so on. Blockchains provide a decentralised digital database for storage of data with a higher degree of security and privacy. Cryptocurrencies like Bitcoin, Ethereum, Dogecoin, Stellar and many others rely on blockchains to keep track of transactions in a safe and decentralised manner. A blockchain ensures data accuracy and security, and can be trusted. No trusted third party is needed. Each node on a computer network has a copy of the blockchain's database.

Key applications and use cases of blockchain

- Cryptocurrencies
- Smart contracts
- Financial services
- Gaming and entertainment
- Supply chain management
- Medical and healthcare
- E-governance
- Media and advertising
- Telecommunications
- Manufacturing
- Travel and transportation
- Social media

Blockchain databases: What's different?

The blockchain is a decentralised database or public ledger that records all network transactions. Data in a database is typically organised into tables, but data in a blockchain is organised into blocks (as per the name) that are linked together without scope for manipulation or hacking. Using this data structure in a decentralised manner creates an irreversible chronology of data which provides a very high level of security, integrity and privacy. Once a block has been linked in the blockchain, it becomes permanent and nobody can alter it accidentally or intentionally.

A blockchain organises data into blocks, and each of these contains a specific sequence with encrypted data. The blockchain is a chain of data made up of blocks, and these store a set amount of data before being closed

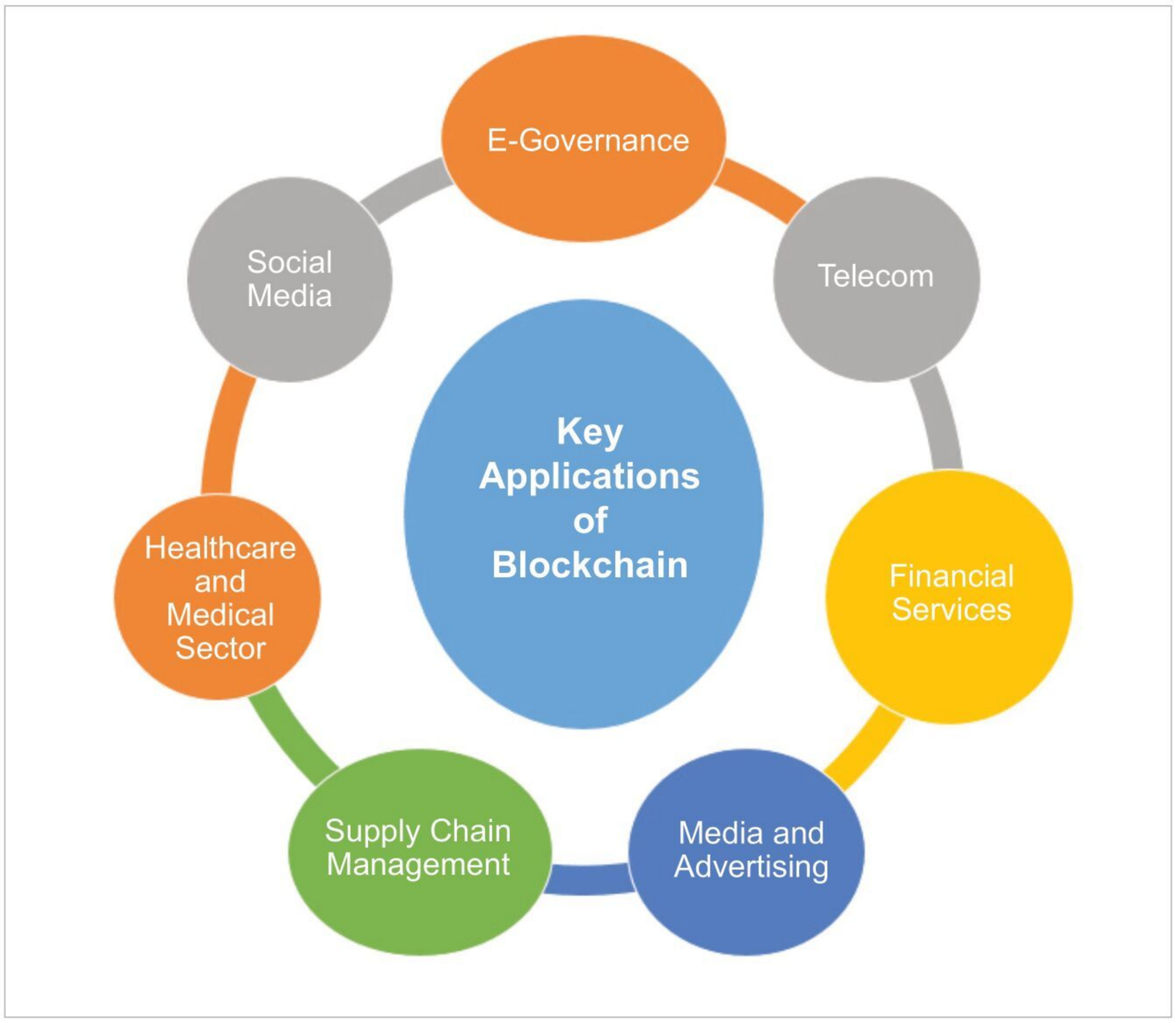


Figure 1: Applications and use cases of blockchain

Table 1: Blockchain vs traditional databases

Blockchain	Traditional databases
Distributed ledger	Centralised database server
Decentralised	Single point of control and scope of failure
Peer-to-peer	Multiple gateways and human control
Cryptography based verification and authentication	Human intervention and scope of attacks
Resilient and fault-tolerant	Need for backups

and connected to the next block in the chain. All subsequent data is assembled into a new block, which is added to the chain after it has been filled.

Blockchain based databases differ from traditional databases due to their decentralised nature and the underlying blockchain technology (Table 1). The data in a blockchain is stored and managed on a digital ledger, which is highly secure. There is no administrator who can exploit the records or tables in it. This is a peer-to-peer network.

The node stores the record in its database and broadcasts it if the vetting procedure by all nodes is successful. The record is then written to each node's database, ensuring that the network's status and records are always up-to-date. These features make it nearly impossible to alter or replicate the data recorded in blockchains. In order to ensure the integrity of the database, all nodes review each new entry before it is added.

The transactions in the blocks of the blockchain are hashed using cryptography, and each block establishes a connected chain with the previous one. Because no node or client can dispute a record, it provides a high degree of transparency.

Open source database engines for blockchain development

Let's take a look at the major open source databases that can be used for development of blockchain based applications.

BigchainDB

(https://www.bigchaindb.com/)

BigchainDB is a blockchain based storage system powered by MongoDB that lets you incorporate decentralised and blockchain technologies into your application. It is perhaps the most popular blockchain database because of its features including immutability and decentralisation.

The BigchainDB database is fault-proof, which implies that once a record has been confirmed and put in the database, it cannot be updated or altered in any way. With the usage of Byzantine Fault Tolerance (BFT), even if up to half of the network's nodes are defective, it can still analyse what the next block will be. For this reason, the network can erase one of the node's MongoDB databases if a hacker gets hold of it.

In addition, it can support multiple assets at once, and can hold numerous types of assets. All assets in the Bigchain network can be issued by the nodes' users. BigchainDB is a leader in supply chain management applications, where data organisation, immutability, and security are really needed.

Apache Cassandra

(https://cassandra.apache.org)

This NoSQL distributed database promises linear scalability and high consistency without sacrificing speed. As a Java based distributed database system, Cassandra provides high availability without a single point of failure by spreading data over a large number of commodity computers.

Each row in Cassandra comprises tables that need a primary key, and each row is partitioned. Cassandra's partitioning allows it to split the rows

among many networks and gadgets. When rows and partitions are deleted or added to the network, they are adjusted across it.

Cassandra is a unique blockchain database because of its numerous significant characteristics. Each node in the cluster has a row and a partition. Because every cluster is both a client and a server, there is no master cluster. Cassandra is a fault-tolerant system because it is distributed and does not have a single point of control.

If one of the nodes in the cluster is attacked, it doesn't bring down the entire system since each of the nodes has a copy of the database. Because the data from the node that is down is still stored on the cluster's other nodes, there is no risk to the information on it. The rows, tables, and columns distinguish Cassandra from other relational databases. Cassandra does not query data using the SQL language as it uses its own query language Cassandra Query Language (CQL).

ChainifyDB

(https://www.chainifydb.com/)

ChainifyDB is a blockchain based database management system. Databases may be integrated into the ChainifyDB network and their records synchronised across the network. Once a record is added to a database, ChainifyDB notifies all other database nodes of the new information. They reach an agreement and the record is written to the databases, making it decentralised, unchangeable, and open.

Unlike other blockchain based databases, ChainifyDB has its own database/storage space for each block it is part of. A blockchain layer is plugged into the databases offered by ChainfyDB, with security, privacy and integrity.

The ChainifyDB network uses strong encryption to transmit data between its connected databases. It is possible to execute ChainifyDB's basic components and administrative configuration via a Web frontend. As it doesn't require a large number of tools to set up, it is more user friendly than other databases. ChainifyDB may be inserted into any database without disrupting the apps that are running on it.

CovenantSQL

(https://covenantsql.io/)

CovenantSQL is a database used widely for blockchain based decentralised applications. As stated on the official website, CovenantSQL intends to make DApp development easier by connecting idle storage resources via a consensus mechanism. Like Ethereum, CovenantSQL provides a foundation over which decentralised applications may be built. This database has a wide range of applications, including asset management and IoT integration.

CovenantSQL is distributed through a P2P network, just like the blockchain technology it employs. Querying databases with SQL is the most common database operation. The immutability of the database is ensured by CovenantSQL's blockchain. Before they can be committed to the database, all records must be verified by every node in the network.

Modex Blockchain Database (BCDB)

(https://modex.tech/blockchain-database-product/)

Modex Blockchain Database (BCDB) offers a simple plug-and-play approach to blockchain development for enterprises. Client applications and databases are separated by Modex BCDB. It can be used with various blockchain frameworks since it is adaptable and easy to integrate. The Hyperledger Sawtooth framework and the Tendermint protocol are now being used in the network and consensus protocols of this platform.

Modex BCDB can support multiple databases. It can smoothly operate and sync data between a node that uses MongoDB and one that uses MySQL, without any configuration or porting to a supported database. Modex BCDB is capable of managing data flawlessly and without sacrificing security in anyway. Nodes in the Modex BCDB network can either be fully or partially accessible to all data, depending on their API requests.

Postchain

(https://postchain-docs.readthedocs.io/en/latest/)

Like Ethereum and Hyperledger, Postchain has a blockchain architecture with a distributed network of nodes. It uses a proof-of-authority consensus process to retain a collection of data. Postchain keeps this data in an SQL database, and every node on the Postchain network has a validator. Transactions are encrypted and signed before being sent. A group of validators works together to verify the message's provenance and source. This synchronisation is carried out to ensure that all network nodes have the same database state.

ProvenDB

(https://www.provendb.com/)

ProvenDB is a MongoDB database that incorporates blockchain technology. Records in ProvenDB can never be changed or removed once they are entered. All types of sensitive data can be stored in a highly secure blockchain database, including financial records that cannot be changed as well as intellectual property, legal papers and public records. ProvenDB delivers a secure database that is encrypted and resistant to hacking.

Blockchain based dedicated database engines can help to develop real-time and secure applications for smart cities, e-governance, supply chain management, education based ERP, etc. The performance of such blockchain based database systems with respect to energy, security, privacy, etc, can be analysed effectively by using different programming platforms. END

By: Dr Gaurav Kumar

The author is associated with various academic and research institutes for delivering expert lectures and conducting technical workshops on the latest technologies and tools.

YugabyteDB: The Reasons Why It's Popular

Yugabyte was first developed as closed source commercial software. However, by 2019, its founders realised the power of the open source community and made it available under the Apache 2.0 license. The timing was right as developers were on the lookout for a cloud native database. The adoption and popularity of this database has increased significantly in recent years for a number of reasons.



YugabyteDB was developed by former Facebook software engineers nearly six years ago. It is today widely adopted across enterprises and industries as it is a fully functional open source distributed SQL database, which supports cloud-native applications.

Distributed SQL acts as a single logical database, whereas behind the scenes it is a database cluster of one or more nodes. A distributed SQL database has a three-layer architecture:

- **SQL API** is the primary interface for applications to model relational data. Like RDBMS indexes, foreign key constraints, JOIN queries and multi-row ACID transactions are unique data models for distributed SQL.
- **Distributed query** execution takes the responsibility for distributed query processing by sharing query requests across other nodes in the cluster. The execution across multiple nodes minimises latency and keeps a check on the amount of data transferred amongst them.
- **Distributed data storage** ensures data is distributed and replicated, avoiding bottlenecks and ensuring high availability, leading to high performance.

Figure 1 gives a reference implementation of distributed SQL across three nodes. An application connecting to node 1 leverages node 2 and 3 for query processing. In case node 1 goes down, node 2 or node 3 picks up the request and continues to process the application, ensuring high availability at all times.

There are many distributed SQL implementations. YugabyteDB is one of the first and most successful ones

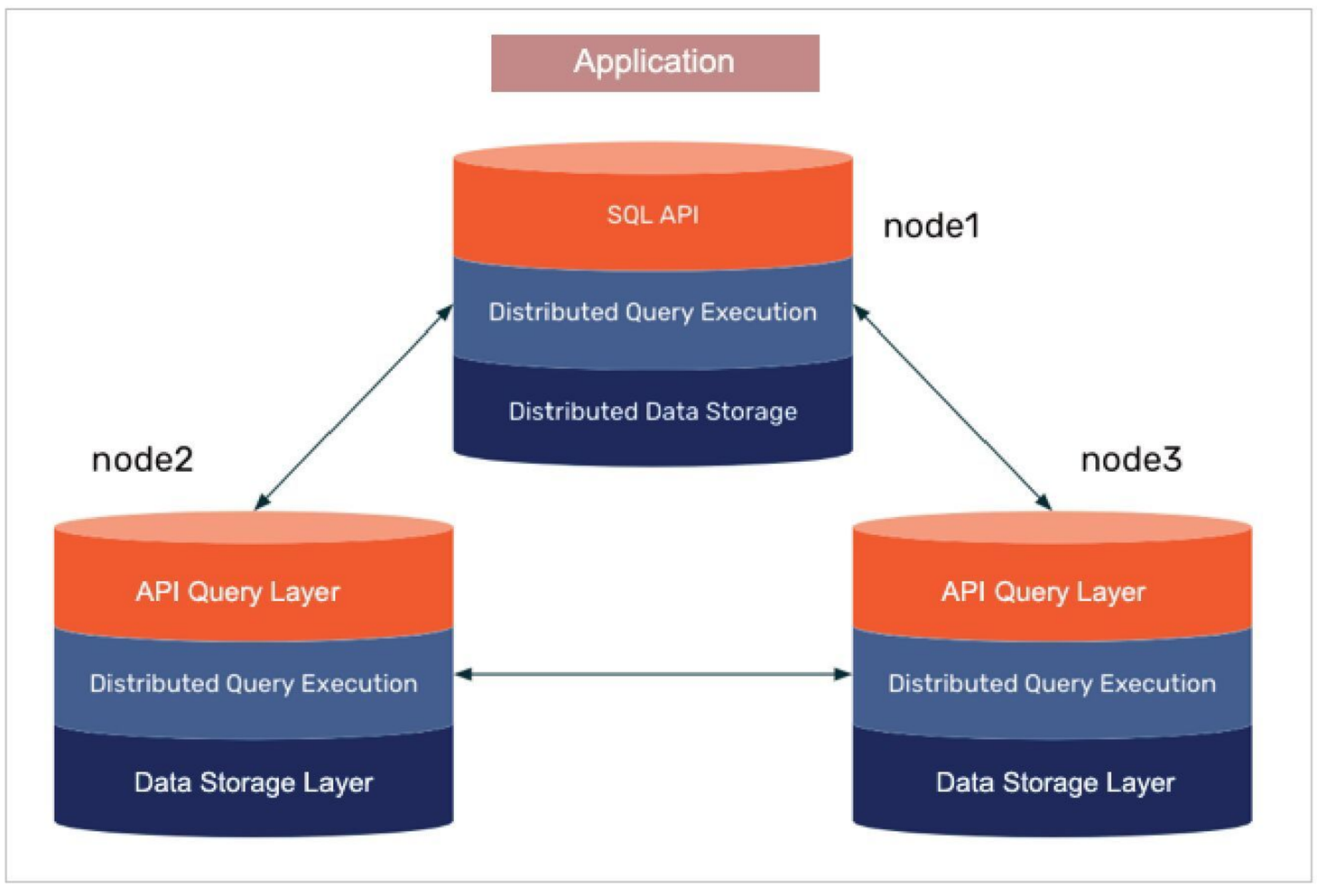


Figure 1: Distributed SQL architecture

and is being used by enterprises of all sizes. For a distributed database to be successful there are typical technology considerations like scale, consistency, resiliency, geo-replication and data locality. Given the growing popularity of multi-cloud, it's also critical for a distributed SQL database to be able to support multi-network deployment. YugabyteDB meets all these criteria.

Layered architecture

YugabyteDB has a layered architecture.

Yugabyte Query Layer (YQL) is the top layer that interfaces with applications interacting directly with it using client drivers. This layer manages API specific aspects for query/command compilation and runtime data representations. YQL is built with extensibility, allowing new APIs to be added as use cases grow.

Yugabyte SQL (YSQL) is a distributed SQL API built on top of the PostgreSQL language layer code. It's stateless SQL query engine is compatible with PostgreSQL.

Yugabyte Cloud QL (YCQL) is a semi-relational language that reflects Cassandra Query Language. Although it is SQL-like language, it's built specifically to be aware of the clustering of data across nodes.

DocDB is the distributed document store that has strong write consistency and is resilient to failures. It has automatic sharding, load balancing and geo aware data placement policies.

Sharding is done to tables stored in DocDB and is transparent to users.

Replication is applied with a configurable replication factor using the Raft consensus algorithm. It is performed at the table level, ensuring single row linearizability in case of failure.

Persistence is achieved with log-structured row/document-oriented storage including many optimisations supporting a growing data set for efficiency.

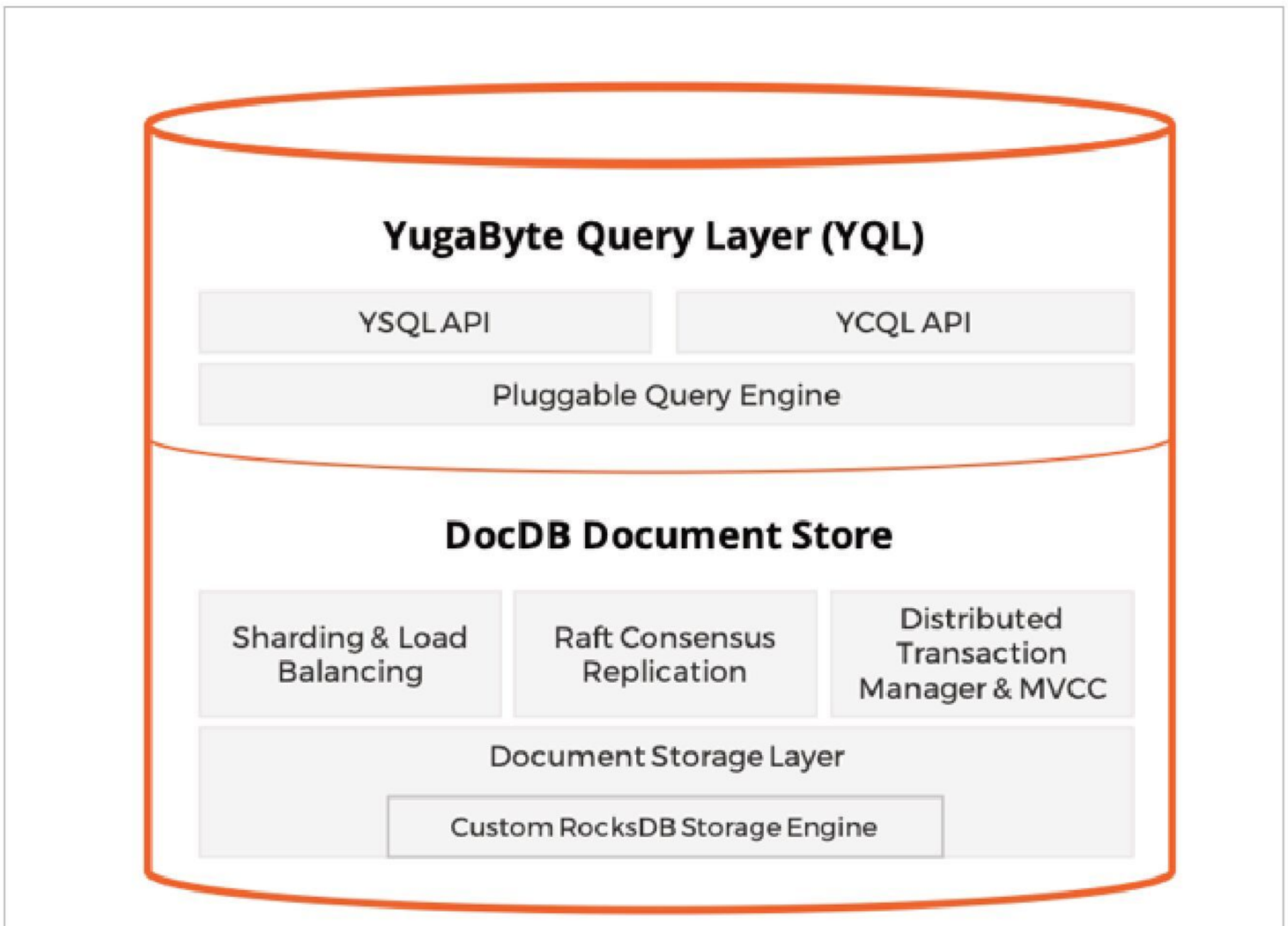


Figure 2: YugabyteDB architecture

Key features

What make YugabyteDB stand apart from other distributed databases are its features:

- Supports single- or multi-row transactions (OLTP apps)
- High performance and linear scalability
- Open source and cloud native with multi-cloud deployability
- Geo-location deployment and global data consistency
- Supports change data capture (CDC) mechanism
- Distributed and multi model
- Offers managed services

The need for data availability, consistency and optimum read/write performance is growing each day. YugaByteDB stands tall amongst all the distributed SQL databases. Its 5x storage sets it apart, given the growing cost of data storage, processing and analytics. END



References

[1] YugaByteDB, *https://www.yugabyte.com/*
[2] YugaByteDB Layered Architecture, *https://docs.yugabyte.com/preview/architecture/layered-architecture/*
[3] Forbes - Can Yugabyte become the de facto database for large scale, cloud native applications, *https://www.forbes.com/sites/brucerogers/2022/05/19/can-yugabyte-become-the-defacto-database-for-large-scale-cloud-native-applications/?sh=f4fc6a947885*

By: Bala Kalavala

The author is a technical architect, evangelist, thought leader and a sought-after keynote speaker. He currently works as a distinguished member of the technical staff and heads the enterprise architecture practice as chief architect at Wipro Ltd. This article expresses his views and not that of Wipro.

Choose the Right Database for Your Project

Selecting the right database is essential for the success of any project. A good data model underpins the overall architecture, provides tremendous insights into the application itself, and builds a solid foundation for a project. There are a number of SQL (RDBMS) and NoSQL based databases available in the market. Which one should you go for?



SQL databases are based on set theory (unions and defined relationships). Here, data is stored in tables (rows and columns). These are ACID (atomicity, consistency, isolation, durability) compliant, have structured data and use SQL as the query language. They support frequent and short transactions that need to be accurate.

On the other hand, NoSQL databases are used when there are large volumes of unstructured data. These typically use cloud storage and support rapid application development. They can work with a wide variety of data models, and data is stored in a distributed fashion. They follow the shared-nothing architecture, and support BASE characteristics (basically available, soft state, eventual consistency). Unstructured data typically includes session data, click-stream data, etc.

There are different types of NoSQL databases:

- Key-value store (Redis, Memcached)
- Column store (Cassandra, HBase)
- Document (Mongo DB, CouchDB)
- Graph (Neo4j)

Evaluation questionnaire

Before deciding the right database (DB) for your application, you must query what is expected from it. You can use the sample set of questions given below, and try and be ready with the answers:

- How much volume of data is the DB going to support (GB/TB/PB, etc)?
- What is the type of data I am going to store (simple or complex data structures, etc)?
- How many concurrent users will be connecting to the database?
- What are the availability requirements for my database (e.g., 99.999%)?
- Does my database need to be highly scalable?
- What is the average response time needed for the queries?
- What is the throughput for the transactions on the database?
- Will I require a real-time consistent database or is some lag permissible?
- Does my data model undergo frequent changes?
- What are the levels of *read* and *write* operations required on the database?

- What is my application architecture that allows easy integration with the database?
- What are my commercial constraints (budget, licences, contracts, etc)?
- What are the typical characteristics of the data to be stored (invalid data, missing data, etc)?
- What are the levels of regulation and compliance that the database needs to adhere to?

The answers to these questions will give you a clear understanding of what you require from a database, helping you to select the right one.

Table 1 lists some of the criteria you should check when selecting a database.

Table 1

Requirement	SQL	NoSQL	Comment
High scalability		Y	Any NoSQL
Consistency and availability	Y		Any DBMS
Consistency and partition tolerance		Y	MongoDB
Availability and partition tolerance		Y	Cassandra
High schema changes		Y	Any NoSQL
Transaction-oriented	Y		Any RDBMS
Inconsistent/incomplete data		Y	NoSQL

Key KPIs

A few KPIs (key performance indicators) that can help you understand the performance of a database are listed below. By tracking these KPIs, you can make a judicious decision of building and maintaining a DBMS that serves the needs of the business.

- Average database availability time
- Average DB usability incidents
- Average service requests per day
- Average service fulfillment time
- Database size, number of instances, etc
- Transactions per second

With the advent of cloud technologies and cloud-first architectures, we now have a plethora of database options to choose from for our project. The cloud operational cost model reduces capital expenditures and allows companies to adopt a 'fail fast, learn faster' approach for whatever database they select. END

By: Phani Kiran

The author has over 18 years of experience in the IT industry and has worked on Big Data technologies.



THE COMPLETE MAGAZINE ON OPEN SOURCE



» To find dealers to buy copy from news-stand, visit: www.ezine.lfymag.com

» To subscribe, visit: pay.efyindia.com

» To buy an ezine edition, visit: www.magzter.com choose Open Source For You

India's No.1 Electronics Magazine
RANKING AS PER ELCINA-IMRB REPORT

"I started reading it when I was a student... ...and I am still reading it, as a student"

—CEO, Design House



electronics FOR YOU
YOURS SINCE 1969

To Subscribe:
https://subscribe.efy.in

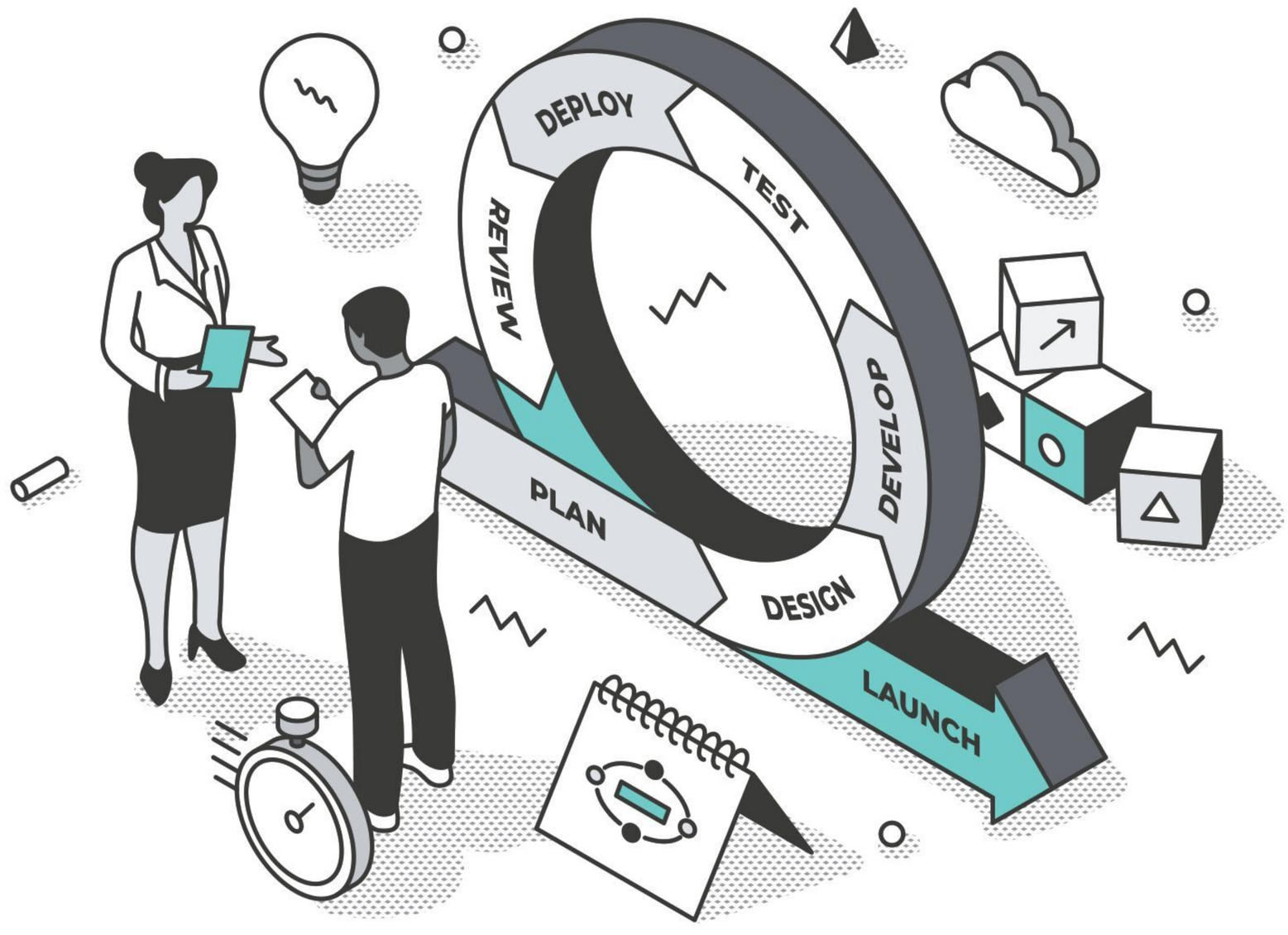
OR

Scan This Code



For any query, call: 011-4059 6600 or email: support@efy.in

Where is DevOps Headed?



Microsoft, Google, Amazon, IBM, and Oracle are today focusing on DevOps on the cloud. IT automation is what these big companies are offering enterprises. However, DevOps is evolving continuously. DevSecOps, AIOps and NoOps are set to be the next buzzwords.

Agile methodology and DevOps have become popular as developers and managers see the business value in delivering good quality software in time. To have flexible release cycles and deliver software with scalable and customisable features is the goal of every enterprise in the world. DevOps has smoothened the release process with CI/CD tools and pipelines being deployed on to the cloud. Polyglot microservices architecture blended with DevOps is helping enterprises to cut down the total cost of ownership. They now have capabilities to upgrade their technology stack with progressive Web apps and the latest UI frameworks. Overall, teams are performing with better efficiency and quality software modules are being developed.

Autonomous DevOps

Containers and DevOps go together with cloud native applications. Kubernetes and Docker are being used as containers and a new term called NoOps is trending now. Orchestration is an important feature of different containers. Container clusters are being created in developer environments to scale the application. There are new containers like Mesos, Swarm, Openshift Rancher and Nomad getting into the cloud native application space. NoOps helps in cutting down the coding cycles in order to monitor and manage the application. Defect fixing and hotfixes are different activities, which are part of NoOps. NoOps helps to improve the synergy between technical teams and business operations personnel. It helps in better monitoring, management, and process automation. NoOps infrastructure enables control of app deployment on the cloud. Enterprises derive benefits like better delivery, service resilience, faster time to market, good quality, and CI/CD automation.

> "Currently, DevOps is more like a philosophical movement, not yet a precise collection of practices, descriptive or prescriptive."
>
> *—Gene Kim, author and researcher*

DevSecOps

DevSecOps is another popular trend related to the security concerns addressed during development operations. Recent issues related to vulnerabilities (log4j), security breaches (Google, Facebook, Microsoft), and security attacks have increased the importance of DevSecOps in enterprises. A shift left approach emphasises the importance of security and quality to be addressed earlier in the software life cycle. Privacy and compliances like GDPR need to be considered at the architectural phase itself. This helps in cutting down costs and improving the speed of the software delivery. Auditing tools and security checklists are part of the DevOps tools and systems, which we call DevSecOps now.

AIOps

AI DevOps is now called AIOps. It is being predicted that AI applications will be managed by AIOps in future. Tools and software related to AIOps are being developed and available as first releases. AI/ML applications deployment and model updates can be handled by AIOps. This will play an important role in Industry 4.0 and data science. There is a school of thought that NoOps will be the end point for AIOps. AIOps consists of data set management, model training, model serving, metadata management, model updates, and service updates. Distributed training is enabled by AIOps, which gives the capabilities for hyper parameter optimisation, workflow management, and 'what if' analysis.

Microservices configuration management

DevOps and microservices are being implemented as standard deployment and architectural blueprints these days. Apps can be scaled at the module level. Microservices help in simplifying the fixing of defects and isolating problem areas. Microservices by design can be scaled by adding more instances of computing power. But when they are not implemented correctly, issues related to data security and management crop up.

Platform as a Product

Software as a Service and Platform as a Product are popular these days on the cloud. DevOps makes these a reality by accelerating the deployment and delivery of features to the platform. CI/CD pipelines help in visualising the app deployment, right from coding to live phases. Continuous delivery, integration and deployment are all part of DevOps. The future is about DevOps assembly lines simulating industry assembly lines.

DevOps is slowly graduating to DevSecOps and AIOps. NoOps for enterprises is the future. The need of the hour is to cut down the occurrence of security related attacks, incidents, and breaches. Data security and privacy is a high priority for enterprises, and these new technologies will all help with that. END

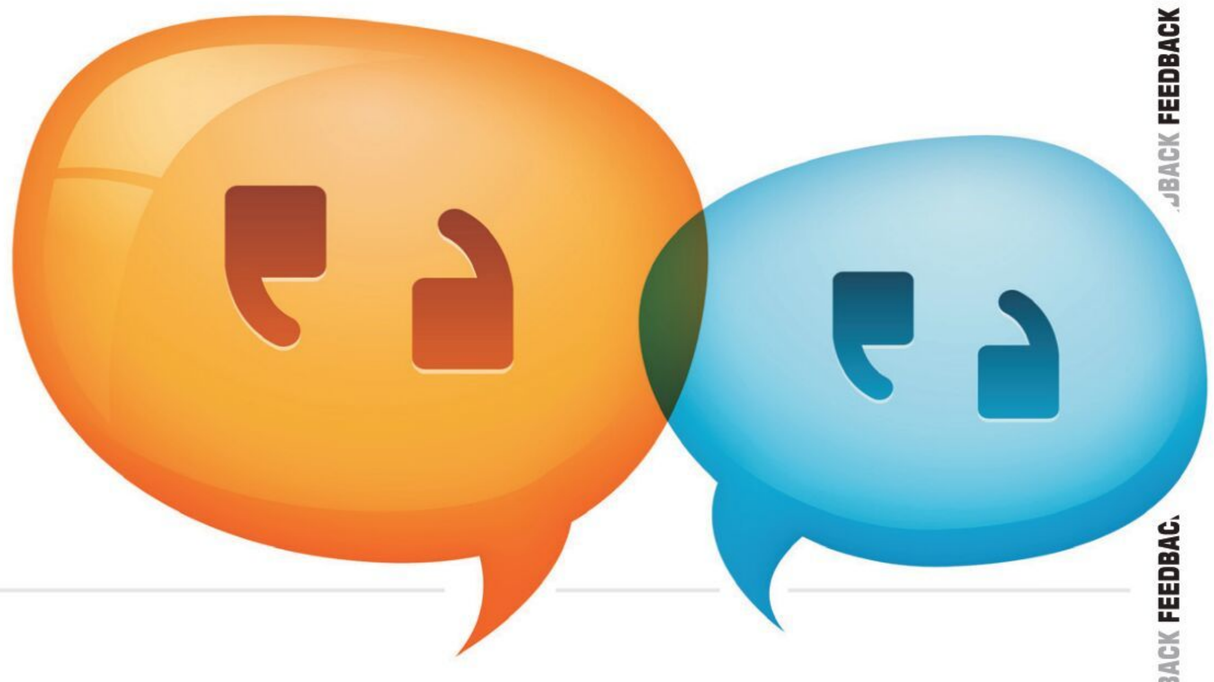
By: Bhagvan Kommadi

The author is founder of 'Quantica Computacao', India's first quantum computing startup. He has 20 years of experience in product development and IT services.

FEEDBACK FEEDBACK FEEDBACK FEEDBACK FEEDBACK FEEDBACK FEEDBACK FEEDBACK FEEDBACK FEEDBACK FEEDBACK FEEDBACK FEEDBACK FEEDBACK FEEDBACK

We Value Your Feedback

We love to hear from you as **OSFY** consistently strives to make its content **informative, interesting** and **engaging.**



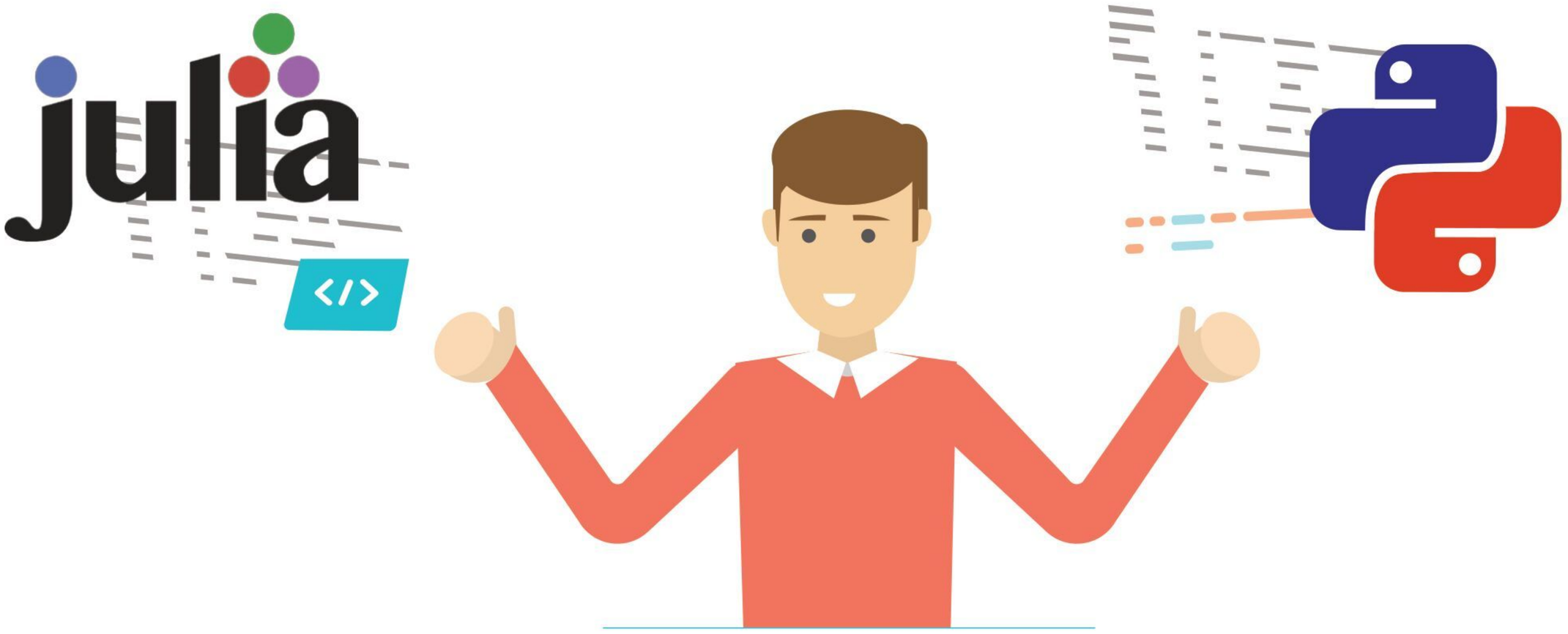
Please share your feedback/ thoughts/ views via email at osfyedit@efyindia.com.

We welcome your comments/ suggestions and encourage you to send them to:
The Editor, D-87/1, Okhla Industrial Area, Phase 1, New Delhi-110020.

OpenSource For You
THE COMPLETE MAGAZINE ON OPEN SOURCE

FEEDBACK FEEDBACK FEEDBACK FEEDBACK FEEDBACK FEEDBACK FEEDBACK FEEDBACK FEEDBACK FEEDBACK FEEDBACK FEEDBACK FEEDBACK FEEDBACK FEEDBACK

Julia and Python: Which Language is Quicker?



Julia is a dynamic programming language with a high level of abstraction. While it is a general-purpose language that may be used to develop any program, it has several characteristics that are ideally suited to numerical analysis and computational research. Python was designed in the early 1990s as a simple object-oriented programming language, but has evolved significantly since then. This article takes a deeper look at the performance of both for neural networks and machine learning.

Julia's architecture features parametric polymorphism in a dynamic programming language, as well as a multiple dispatch programming paradigm as its primary programming model. It allows concurrent, parallel, and distributed computing with or without message passing interface (MPI) or the built-in 'OpenMP-style' threads, as well as the direct invocation of C and FORTRAN libraries without intermediate code. Julia employs a just-in-time (JIT) compiler, which the Julia community refers to as 'just-ahead-of-time' (JAOT) since it compiles all code to machine code by default before running it.

Unlike Python, Julia was designed specifically for use in statistics and machine learning.

Julia can fly through linear algebra, whereas Python can plod through it. This is because Python was never designed to accommodate all of the matrices and equations that machine learning requires. Python isn't bad by any means, especially with NumPy, but Julia is a lot better tailored to this kind of mathematics in terms of a no-package experience. Julia's operand system is a lot more like R's than Python's, which is a significant plus. The majority of linear algebra can be completed in less time and with less effort.

As we know, in recent years Python has dominated the areas of machine learning and data science. Because of the variety of third-party libraries that we can use in Python, it helps to develop machine learning code easily. While there are so many advantages of Python, there is one major drawback — it's an interpreted language, which makes it slow. This is the age of data, and the more data there is the more time it takes to work on it. That's where Julia comes into the picture.

Most of the research work on Julia so far has been on topics like high power computing or the scientific

calculation capabilities of Julia. But here we will talk about how Julia is not only capable of working on complex scientific calculations efficiently but also on commercial-based problems, and can tackle Python in machine learning and neural networks.

Objective and experimentation

Julia is as simple as Python but is a compiled language like C. So let's test how fast Julia is in comparison with Python. For that, we will first test these languages on some simple programs and then move to the main focus of our experiment, which is to test them for machine learning and deep learning.

Julia and Python provide many libraries and open source benchmarking tools. For benchmarking and calculating time in Julia, we used CPUTime and time libraries. Similarly, for Python, we used the time module.

Matrix multiplication

We tried out simple arithmetic operations first, but since these will not generate much difference in time we decided to check out the timing in matrix multiplication. We started with creating two (10 * 10) matrices of random float numbers and performed dot products in these. As we know, Python has a NumPy library, which is famous for working with matrices and vectors. Similarly, Julia has a LinearAlgebra library that works well with matrices and vectors. So we compared the matrix multiplication with and without using their respective libraries. The source code for all the programs used in this article is available in our GitHub repository *(https://github.com/mr-nerdster/Julia_Research.gitsee)*. The 10x10 matrix multiplication program written in Julia is given below:

```
@time LinearAlgebra.mul!(c,x,y)

function MM()
    x = rand(Float64,(10,10))
    y = rand(Float64,(10,10))
    c = zeros(10,10)
    for i in range(1,10)
        for j in range(1,10)
            for k in range(1,10)
                c[i,j] += x[i,k]*y[k,j]
            end
        end
    end
end
@time MM

0.000001 seconds
MM (generic function with 1 method)
```

Julia takes 0.000017 seconds using the library and 0.000001 seconds using loops.

The same matrix multiplication program was written in Python, as shown below. From the results it can be seen that with the library the program takes less time compared to without the library.

```
import numpy as np
import time as t
x = np.random.rand(10,10)
y = np.random.rand(10,10)
start = t.time()
z = np.dot(x, y)
print("Time = ",t.time()-start)
Time =  0.0013160705566406250

import random
import time as t
l = 0
h= 10
cols = 10
rows= 10

choices = list (map(float, range(l,h)))
x = [random.choices (choices , k=cols)
for _ in range(rows)]
y = [random.choices (choices , k=cols)
for _ in range(rows)]

result = [([0]*cols) for i in range
(rows)]

start = t.time()

for i in range(len(x)):
    for j in range(len(y[0])):
        for k in range(len(result)):
            result[i][j] += x[i][k] *
y[k][j]

print(result)
print("Time = ", t.time()-start)

Time =  0.0015912055969238281
```

Python takes 0.0013 seconds using the library and 0.0015 seconds using loops.

Linear search

The next experiment that we performed was a linear search on one hundred thousand randomly generated numbers. We used two methods here — one by using a *for* loop and the other by using an operator. We performed 1000 searches with integers ranging from 1 to 1000, and as you can see in the output below we also printed out how many integers we find in the data set. The output of time by using loops and by using the IN operator is given below. Here we measured time by taking the median CPU time of 3 runs.

The program was written for Julia and the results are shown below.

```
FOR_SEARCH:
        Elapsed CPU time: 0.417767
seconds
            matches: 550
        Elapsed CPU time: 0.41119
seconds
            matches: 550
        Elapsed CPU time: 0.425561
seconds
            matches: 550

    IN:
            Elapsed CPU time: 0.433371
    seconds
                matches: 550
            Elapsed CPU time: 0.44637
    seconds
                matches: 550
            Elapsed CPU time: 0.433301
    seconds
                matches: 550
```

The same program was written for Python and the results are:

```
FOR_SEARCH:
        Elapsed CPU time: 16.420260511
seconds
            matches: 550
        Elapsed CPU time: 16.140975079
seconds
            matches: 550
        Elapsed CPU time: 16.49639576
seconds
            matches: 550

IN:
        Elapsed CPU time: 6.446583343
seconds
            matches: 550
        Elapsed CPU time: 6.216615487
seconds
            matches: 550
        Elapsed CPU time: 6.296716556
seconds
            matches: 550
```

From the above results, it is evident that there are no time differences between using a loop and an operator in Julia. However, the loop takes almost three times more execution time than the IN operator in Python. The interesting point here is that, in both cases, Julia has a much faster execution time than Python.

Linear regression

The next experiments were performed in machine learning algorithms. We first worked on one of the most common and simple machine learning algorithms, i.e., linear regression with a simple data set. We used a 'Head Brain' data set that contains 237 data entries and has two columns [HeadSize, BrainWeight]. In this, we had to calculate the brain weight by using the head size. So we implemented linear regression from scratch, without using any library on this data set in both Python and Julia.

Julia:

```
GC.gc()
@CPUtime begin
    linear_reg()
end
elapsed CPU time: 0.000718 seconds
```

Python:

```
gc.collect()
start = process_time()
linear_reg()
end = process_time()

print(end-start)
elapsed time: 0.007180344000000005
```

The time taken by both Julia and Python is given above.

Logistic regression

Next, we carried out an experiment on the most common type of machine learning algorithm, i.e., logistic regression, by using libraries in both languages. For Python, we used its most commonly used library sklearn while in Julia we used the GLM library. The data set that we used for this is the information about a bank's clients, which contains 10,000 data entries. The target variable is a binary variable that indicates whether the consumer left the bank (closed his or her account) or remained a customer.

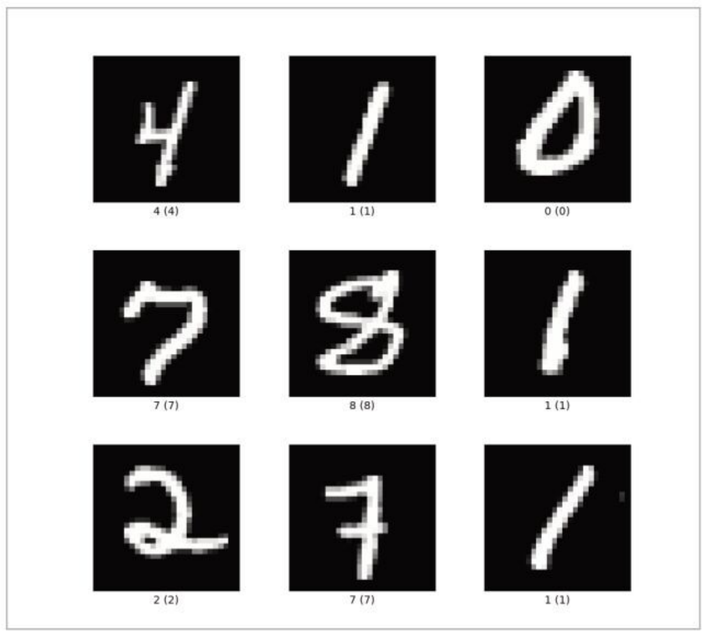


Figure 1: Example of MNIST data set

The time taken by Julia for logistic regression is given below.

```
@time log_rec()
0.027746 seconds (3.32 k allocations:
10.947 MiB)
```

The time taken by Python for logistic regression is also given below.

```
gc.collect()
start = process_time()
LogReg()
end = process_time()
print(end-start)

Accuracy : 0.8068
0.34901400000000005
```

Neural networks

After testing both languages on various programs and data sets, we tested them on neural networks and used the MNIST data set. This data set contains gray-scale images of hand-drawn digits, from zero through nine. Each image is 28x28 pixels. Each pixel value indicates the lightness or darkness of that pixel, and this value is an integer between 0 and 255, both inclusive. The data also contains a label column which represents the digit that was drawn in the respected image.

Figure 1 shows a few examples of the MNIST data set.

We created a simple neural network to test the time taken by both languages. The structure of our neural network is like this:

```
Input ---> Hidden layer ---> Output
```

```
@time Flux.train!(loss, params(model), Iterators.repeated(trainbatch,40), opt; cb = Flux.throttle(callback,10))

loss(trainbatch...) = 2.0350972755858594
  5.766363 seconds (282.95 k allocations: 1.339 GiB, 2.17% gc time, 6.92% compilation time)
```

Figure 2: Julia takes 5.76 seconds in a neural network

```
Epoch 29/40
1875/1875 [==============================] - 3s 2ms/step - loss: 5.1357 - acc: 0.8942 - val_loss: 6.1563 - val_acc: 0.8961
Epoch 30/40
1875/1875 [==============================] - 3s 1ms/step - loss: 5.1226 - acc: 0.8942 - val_loss: 6.0663 - val_acc: 0.8972
Epoch 31/40
1875/1875 [==============================] - 3s 1ms/step - loss: 5.0476 - acc: 0.8953 - val_loss: 6.3820 - val_acc: 0.8876
Epoch 32/40
1875/1875 [==============================] - 3s 1ms/step - loss: 5.1187 - acc: 0.8938 - val_loss: 6.5461 - val_acc: 0.8898
Epoch 33/40
1875/1875 [==============================] - 3s 1ms/step - loss: 5.0417 - acc: 0.8949 - val_loss: 6.2389 - val_acc: 0.8965
Epoch 34/40
1875/1875 [==============================] - 3s 1ms/step - loss: 5.1899 - acc: 0.8936 - val_loss: 6.0297 - val_acc: 0.9060
Epoch 35/40
1875/1875 [==============================] - 3s 1ms/step - loss: 5.1787 - acc: 0.8945 - val_loss: 6.2162 - val_acc: 0.8983
Epoch 36/40
1875/1875 [==============================] - 3s 2ms/step - loss: 5.1070 - acc: 0.8950 - val_loss: 6.2734 - val_acc: 0.8943
Epoch 37/40
1875/1875 [==============================] - 3s 1ms/step - loss: 4.9866 - acc: 0.8953 - val_loss: 6.4508 - val_acc: 0.8888
Epoch 38/40
1875/1875 [==============================] - 3s 1ms/step - loss: 5.1557 - acc: 0.8949 - val_loss: 6.9689 - val_acc: 0.8829
Epoch 39/40
1875/1875 [==============================] - 3s 1ms/step - loss: 5.1465 - acc: 0.8952 - val_loss: 6.3960 - val_acc: 0.9002
Epoch 40/40
1875/1875 [==============================] - 3s 1ms/step - loss: 4.9815 - acc: 0.8967 - val_loss: 6.2203 - val_acc: 0.8968
110.33991527557373
```

Figure 3: Python takes 110.3 seconds in a neural network

It contains an input layer, hidden layer, and output layer. To avoid complexities, we did not use any preprocessing on our data set, and worked on it as it is. We trained this model for 40 iterations and checked the time difference between Julia and Python in it.

For Julia, the Flux library was used to implement the neural network and for Python, the Keras library was used. Figure 2 shows the time taken by Julia in a neural network.

Figure 3 shows the time taken by Python and a few iterations of the model in the neural network.

The results show that there are huge time differences between Julia and Python when it comes to a neural network.

Table 1 summarises all the results of our experiments, and gives the time difference (in percentage) between Julia and Python.

All the experiments we carried out indicated that as the complexity of the program as well as the size of the data set increases, the execution time difference between Julia and Python increases. From the results, we can conclude that Julia is the better programming language for machine learning and neural networks. END

Experiment	Julia (seconds)	Python (seconds)	Time difference (%)
Matrix multiplication (without library)	0.000001	0.0015	99.9
Matrix multiplication (with library)	0.000017	0.0013	98.69
Linear search (using loop)	0.42	16.4	97.43
Linear search (using IN operator)	0.43	6.2	93.06
Linear regression	0.000718	0.00718	90
Logistic regression	0.025	0.34901	92.83
Neural networks	5.76	110.3	94.77

Table 1: Time difference between Julia and Python for all experiments

References

[1] Sascha Hunold and Sebastian Steiner, 'Benchmarking Julia's Communication Performance: Is Julia HPC ready or Full HPC?', 2020 IEEE/ACM Performance Modeling, Benchmarking and Simulation of High-Performance Computer Systems (PMBS) 2021.

[2] Ruizhu Huang, Weijia Xu, Yinzhi Wang, Silvia Liverani, Ann E. Stapleton, 'Performance Comparison of Julia Distributed Implementations of Dirichlet Process Mixture Models', IEEE International Conference on Big Data (Big Data) 2019.

[3] 'Julia Benchmarking Tools', available online at: *https://juliaci.github.io/BenchmarkTools.jl/stable/*

[4] Source code for this experimental work is available in our GitHub repository: *https://github.com/mr-nerdster/Julia_Research.git*

[5] 'Handwritten-Digit-Recognition-using-MNIST-dataset-and-Julia-Flux', available online at: *https://github.com/crhota/Handwritten-Digit-Recognition-using-MNIST-dataset-and-Julia-Flux*

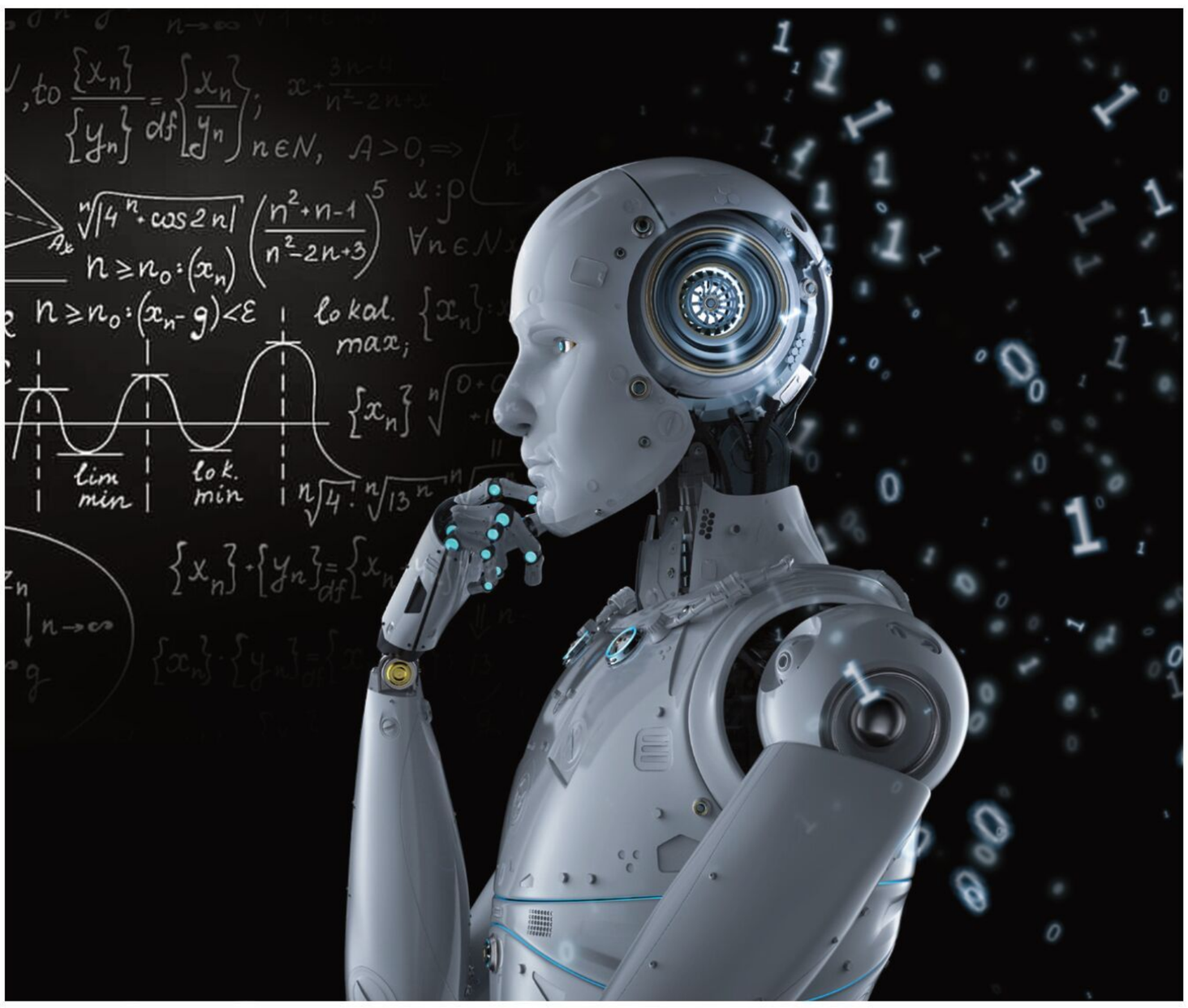
[6] Muhammad Ardi, 'Simple Neural Network on MNIST Handwritten Digit Dataset', available online at: *https://becominghuman.ai/simple-neural-network-on-mnist-handwritten-digit-dataset-61e47702ed25*

By: Akshat Jain, Md Zainul Haque and B. Thangaraju

The authors are associated with the Open Source Technology Lab in the International Institute of Information Technology, Bengaluru.

AI: Some History and a Lot More about Matrices

In the first article on this series on AI (published in *OSFY,* June 2022), we discussed how fields like AI, machine learning, deep learning, data science, etc, are related yet distinct from each other. We also made a few difficult choices regarding the programming language, tools, etc, which will be used throughout this series. Finally, we started discussing matrices. In this article, we will discuss more about matrix theory, the heart of AI. But before we proceed any further, let us discuss the history of AI.



First of all, why is a discussion on the history of AI essential? Well, there were a few times in history when there was great enthusiasm around research on AI, but on many of those occasions the huge expectations about the potential of AI went wrong. So, we need to learn some history of AI to understand whether this time we really are going to do wonders with AI or is it yet another bubble that is about to burst.

One important question that needs to be answered is: when did our fascination for artificial intelligence start? Was it after the invention of digital computers or did it start even before? I believe the quest for an omniscient being capable of answering all questions dates back to the beginning of civilisation. The Oracle of Delphi in the ancient times was one such supposedly divine being capable of answering all the queries asked by people. The quest for creative machines outwitting human intelligence was also fascinating to us from ancient times. There were several failed attempts to make machines capable of playing chess. For example, Mechanical Turk was an infamous fake chess playing machine constructed in the late 18th century. The inventions of logarithm by John Napier, the Pascal's calculator by Blaise Pascal, difference engine by Charles Babbage, etc, were all predecessors to the development of artificial intelligence. Further, if you go through the history of mankind, you will find countless other real and imaginary (fake) moments hoping to achieve intelligence beyond the reach of the human brain. However, irrespective of all these achievements, the search for true AI began with the invention of digital computers.

However, if that is the case, the important question is: what were some of the milestones in the history of AI up until today? As mentioned earlier, the invention of digital computers is the single most significant event in the search for artificial intelligence. Unlike electro-mechanical devices where scalability was dependent on power requirements, digital devices simply benefited technological advancement like the transition from vacuum tubes to transistors to integrated circuits to present day very large-scale integration (VLSI) technology.

Another important phase in the development of AI could be traced back to the first theoretical analysis of AI by Allan Turing. The Turing test proposed by him suggests one of the earliest methods to test artificial intelligence. The test may not be relevant today but it was one of the first attempts to define artificial intelligence. It can be summarised as follows. Behind a curtain, there is either a person named A or a machine named B. You can talk with him/it for say 30 minutes, after which you should be able to identify whether you were conversing with a person or a machine. With the powerful chatbots of today, it is easy to see that such a test will not identify true AI. But in the early 1950s this did give a theoretical framework to understand AI.

In the late 1950s, a programming language called Lisp was developed by John McCarthy. This was one of the earliest high-level programming languages. Until then, machine languages and assembly languages (notoriously difficult to use) were used for computer programming. A very powerful machine, a very powerful language to communicate with it, and computer scientists being the optimists and dreamers made the hope for artificially intelligent machines reasonable for the first time in the history of humanity. In the early 1960s, the hopes for artificially intelligent machines were at their peak. Of course, there were a lot of developments in the computing world, but were there any wonders? Sadly, no. So, the 1960s saw hope and despair about AI for the first time. However, computer science was developing at a pace that is unparalleled in any field of human endeavour.

Then came the 70s and the 80s, where algorithms played a major role. A lot of new and efficient algorithms were developed during this period. The famous textbook 'The Art of Computer Programming' written by Donald Knuth (I request you to read about this person, in the world of computer science he is either Gauss or Euler) had its first volume published in the late 1960s marking the beginning of the algorithmic era. A lot of general and graph algorithms were developed during these years. Further, programming based on artificial neural networks was also introduced during this time. Though McCulloch and Pitts were the pioneers in proposing artificial neural networks as far back as the 1940s, it took decades to become a mainstream technique. Today, deep learning is almost exclusively based on artificial neural networks. Such developments in the field of algorithms led to a resurgence in research in the field of artificial intelligence in the 1980s. However, this time, limitations in communication and computing powers derailed progress in this area. For a second time, AI couldn't live up to the huge expectations it had generated. Then came the 90s, Y2K, and the present day. Once again there is a lot of enthusiasm and hope about the positive impact of AI based applications.

Based on our discussion, it is easy to see that there were at least two other occasions in the digital era when AI was promising. On both those occasions AI didn't live up to its expectations. Is the current enthusiasm around AI a similar one? Of course, this is a very difficult question to answer. But in my personal opinion, this time AI is going to make a huge impact. But what are the factors that made me make such a prediction? First, very powerful computing devices are cheaply available nowadays. Unlike in the 1960s or 1980s, when a handful of such powerful machines were available, we now have them in the millions and billions. Second, there are large data sets available today for training AI and machine learning based applications. Imagine an AI engineer working in the field of digital image processing in the 90s. How many digital images were available for training his/her algorithms? Well, maybe a few thousands or tens of thousands. A cursory search on the Internet will tell you that the data science platform Kaggle (a subsidiary of Google) itself has more than ten thousand data sets. The large amount of data produced on the Internet every day makes training of algorithms a lot easier. Third, high speed connectivity has made it easy to collaborate with large groups. Imagine the difficulty of computer scientists collaborating with each other until the first decade of the 21st century. The present day speed of the Internet has made collaborative AI based projects like Google Colab, Kaggle, Project Jupyter, etc, a reality. Owing to these three factors, I believe this time AI is here for good and will have some excellent applications.

More about matrices

Now that we are somewhat familiar with the history of AI, it is time to revisit one of the main themes of this tutorial on AI — matrices and vectors. In the last article, we began discussing these. This time we are going to dive a little deeper into the world of matrix theory. I will begin by pointing your attention to Figures 1 and 2, which show eight matrices (A, B, C and D, and matrices E, F, G and H). Why so many matrices in a tutorial on AI and machine learning? First, as mentioned in the previous article in this series, matrices are the core of linear algebra, and linear algebra is the heart of machine learning, if not

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 8 \end{bmatrix}$$

$$B = \begin{bmatrix} 1 & 2 & 3 \\ 2 & 4 & 6 \\ 3 & 6 & 9 \end{bmatrix}$$

$$C = \begin{bmatrix} 1 & 2 & 3 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$D = \begin{bmatrix} 11 & 22 & 33 \end{bmatrix}$$

Figure 1: Matrices A, B, C and D

$$E = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}$$

$$F = \begin{bmatrix} 11 & 22 & 33 \\ 77 & 88 & 99 \end{bmatrix}$$

$$G = \begin{bmatrix} 11 \\ 22 \\ 33 \end{bmatrix}$$

$$H = \begin{bmatrix} 1 & 2 & 3 & 4 \end{bmatrix}$$

Figure 2: Matrices E, F, G and H

the brain. Second, each of these matrices does serve a specific purpose in the following discussion.

Let us see how matrices can be represented and how some details about them can be obtained. Figure 3 shows how matrix A is represented with NumPy. Though they are not equivalent, we use the terms matrix and array as if they are synonyms.

I urge you to carefully learn how a matrix is created using the function *array* of NumPy. There is also a function *matrix* offered by NumPy to create two-dimensional arrays and matrices. However, this function will be deprecated in the future and its use is no longer encouraged. We have created matrix A and Figure 3 also shows some details about it. The line of code *A.size* tells us about the number of elements in the array. In this case, it is 9. The line of code *A.nidm* tells the dimension of the array. In this case, it is easy to see that matrix A is two-dimensional. The line of code *A.shape* gives us the order of matrix A. The order of a matrix is the number of rows and columns of that matrix. Notice that A is a 3 x 3 matrix (3 rows and 3 columns). Though I am not explaining any further, knowing the size, dimension, and shape of an array is a little tricky with NumPy. Figure 4 shows you why the size, dimension and shape of a matrix should be identified carefully. Minor changes while defining an array could lead to changes in its size, shape, and dimension. Hence, from Figure 4, it is clear that a programmer should be extra careful while defining matrices.

```
>>> A=np.array([[1,2,3],[4,5,6],[7,8,9]])
>>> A
array([[1, 2, 3],
       [4, 5, 6],
       [7, 8, 9]])
>>> A.size
9
>>> A.ndim
2
>>> A.shape
(3, 3)
```

Figure 3: Matrix A with NumPy

```
>>> X=np.array([[1,2,3],[4,5,6],[7,8]])
>>> X.ndim
1
>>> X=np.array([[1,2,3],[]])
>>> X.shape
(2,)
>>> X.size
2
>>> X=np.array([1,2,3])
>>> X.ndim
1
>>> X=np.array([[1,2,3]])
>>> X.ndim
2
>>> X=np.array([[[1,2,3]]])
>>> X.ndim
3
```

Figure 4: Size, dimension and shape of an array

```
>>> B=np.array([[1,2,3],[2,4,6],[3,6,9]])
>>> A.shape == B.shape
True
>>> A+B
array([[ 2,  4,  6],
       [ 6,  9, 12],
       [10, 14, 17]])
>>> np.add(A,B)
array([[ 2,  4,  6],
       [ 6,  9, 12],
       [10, 14, 17]])
>>> D=np.array([11,22,33])
>>> A+D
array([[12, 24, 36],
       [15, 27, 39],
       [18, 30, 41]])
```

Figure 5: Matrix addition

Now let us perform some basic matrix operations. Figure 5 shows how two matrices A and B are added. Indeed, there are two distinct Pythonic ways to *add* the two matrices, either using the function *add* of NumPy or the operator plus (+) for matrix addition. A very important point to remember is that two matrices can be added only if they are of the same order. For example, two 4 x 3 matrices can be added whereas a 3 x 4 matrix and a 2 x 3 matrix cannot be added. However, since programming is different from mathematics, NumPy does not follow this rule in practice. Figure 5 also shows how matrices A and D are added. Remember that, in the mathematical world, this sort of matrix addition is not valid. A rule called *broadcasting* decides how matrices of different orders should be added. We won't be discussing broadcasting at this point but if you are familiar with C or C++, then typecasting of variables might give you a hint as to what happens when broadcasting rules are applied in Python. So, if you want to make sure that you are performing matrix addition in the true mathematical sense, it would be safe if the following test is performed and the answer is *True*:

```
A.shape == B.shape
```

It can be verified that if you try to add matrices D and H, then it will give you an error.

Please don't think that matrix addition is the only operation we are going to spend time on. Of course, there are a lot more matrix operations. Matrix subtraction and multiplication are illustrated in Figure 6. Notice that subtraction and multiplication also have two distinct Pythonic ways to perform the required operations. The function *subtract* or the operator for matrix subtraction (-) and the function *matmul* or the operator for matrix multiplication (@), can be used for performing matrix subtraction and multiplication, respectively. Execution of the operator * is also shown in Figure 3. Notice that the function *matmul* of NumPy and the operator @ perform matrix multiplication in the mathematical sense. So, be careful with the operator * while dealing with matrices.

Two matrices of order (m x n) and (p x q) can be multiplied if and only if n is equal to p. Further, the product matrix will be of order (m x q). Figure 7 shows more examples of matrices being multiplied. Notice that E@A is possible whereas A@E leads to an error. Carefully go through the examples of *D@G* and *G@D*. Using the shape attribute, identify

```
>>> C=np.array([[1,2,3],[0,0,0],[0,0,0]])
>>> A-C
array([[0, 0, 0],
       [4, 5, 6],
       [7, 8, 8]])
>>> np.subtract(A,C)
array([[0, 0, 0],
       [4, 5, 6],
       [7, 8, 8]])
>>> A@C
array([[ 1,  2,  3],
       [ 4,  8, 12],
       [ 7, 14, 21]])
>>> np.matmul(A,C)
array([[ 1,  2,  3],
       [ 4,  8, 12],
       [ 7, 14, 21]])
>>> A*C
array([[1, 4, 9],
       [0, 0, 0],
       [0, 0, 0]])
```

Figure 6: More matrix operations

```
>>> E@A
array([[30, 36, 39],
       [66, 81, 90]])
>>> A@E
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
ValueError: matmul: Input operand 1 has a mismatch in its core
 dimension 0, with gufunc signature (n?,k),(k,m?)->(n?,m?) (si
ze 2 is different from 3)
>>> D@G
array([[1694]])
>>> G@D
array([[ 121,  242,  363],
       [ 242,  484,  726],
       [ 363,  726, 1089]])
```

Figure 7: More examples of matrix multiplication

which among the eight matrices can be multiplied with each other. Though by the strict mathematical definition, a matrix is two-dimensional, we will be dealing with arrays of higher dimensions. So, just to give an example, let us see how a three-dimensional array is represented using NumPy. The following line of code creates a three-dimensional array named T.

```
T = np.arr ay([[[11,22],[33,44]],[[55,66],[77,88]]])
```

An introduction to Pandas

So far, we were entering matrices from the keyboard. What if we want to read a large matrix from a text file or a data set (very large for that matter) and process it? Here comes another powerful Python library to our rescue, Pandas. Let us begin by reading a small CSV (comma-separated value) file, which is a type of text file (don't worry, soon we will read Microsoft Excel and LibreOffice/OpenOffice Calc files). Figure 8 shows how a CSV file called *cricket.csv* is read, and the first three lines in it are printed on the terminal. More features of Pandas will be discussed in the coming articles in this series.

```
>>> import pandas as pd
>>> dataframe = pd.read_csv('cricket.csv')
>>> dataframe.head(3)
      Name  Centuries
0   Sachin         49
1    Kohli         43
2  Ponting         30
```

Figure 8: Reading a CSV file with Pandas

Rank of a matrix

The rank of a matrix is the *dimension* of the *vector space spanned* by its rows (columns). The three italicised words, dimension, vector space and spanned, may not be familiar to you unless you remember your college level linear algebra. If you fully understand what is meant by the rank of a matrix, I am happy and no further explanation is required. However, if you are not familiar with the terms, think of it as the amount of information contained in the matrix. Again, this is a gross oversimplification and I am not proud of it. Figure 9 shows how the ranks of matrices are found out. Matrix A has rank 3 because none of its rows can be obtained from any other rows in it. Matrix B's rank is just 1, because the second and third rows can be obtained by multiplying the first row ([1, 2, 3]) with 2 and 3, respectively. Matrix C only has one non-zero row and hence has rank 1. Similarly, the ranks of the other matrices can also be understood. Rank is very relevant in our discussion and we will revisit this topic in later articles in this series.

```
>>> np.linalg.matrix_rank(A)
3
>>> np.linalg.matrix_rank(B)
1
>>> np.linalg.matrix_rank(C)
1
>>> np.linalg.matrix_rank(D)
1
>>> np.linalg.matrix_rank(E)
2
>>> np.linalg.matrix_rank(F)
2
>>> np.linalg.matrix_rank(G)
1
>>> np.linalg.matrix_rank(H)
1
```

Figure 9: Finding the rank of a matrix

Now it is time for us to wind up our discussion for this month. In the next article in this series, we will add more tools to our arsenal so that they can be used for developing AI and machine learning based applications. We will also discuss terms like neural networks, supervised learning, unsupervised learning, etc, in more detail. Further, from the next article onwards, we will use JupyterLab in place of the Linux terminal. END

By: Deepu Benson

The author is a free software enthusiast. His area of interest is theoretical computer science. The open source tools of his choice include ns-2 and ns-3. He maintains his own technical blog too.

The AI and ML Services that Microsoft Azure Offers

Humongous amounts of data are being generated each moment today. Artificial intelligence and machine learning services are needed to handle and analyse this Big Data. This is where the AI and ML services of Microsoft Azure come in. Read on to find out how they are helping developers and businesses today.



Artificial intelligence (AI) is the traditional model devised to mimic human brain functions by computers using a technology-driven approach along with mathematical and statistical models (e.g., probability). Machine learning (ML) was conceived later to make programming models of brain functions. It aimed to improve the behaviour of AI models using various learning approaches and storing the results of machine learning models for future predictions. This was inspired by the neural network — the core storage layer of the human brain that stores data using the input layer, the hidden layer and the output layer.

There are many popular machine learning frameworks available in the market, many of which are open source. To choose a machine learning tool or framework, one has to apply the problem scenario, data requirement, data volume to process, training data availability, trainability, algorithm support, visualisation and reporting facility, capability to auto-build the knowledge or training model, processing speed required (CPU/GPU), to name a few.

In ML, building the model is a time consuming task as a lot of training is required to get accuracy for the given problem. For example, for text recognition from images, a lot of training needs to be done in order to build the neural network of data sets for better recognition accuracy. This requires higher compute resources and hence greater scalability. This can be well achieved with cloud based platforms and, hence, machine learning services in cloud platforms are very popular today.

Azure machine learning services, including ML Studio, are being widely used for various business scenarios like anomaly detection, sentiment analysis for marketing research, and predictive analysis for customer relationship management, to name a few. But there are more practical business use cases now being experimented using Azure ML services in the banking industry. These include biometric identification, cheque processing using biometric analysis like face detection, and signature detection using pattern matching.

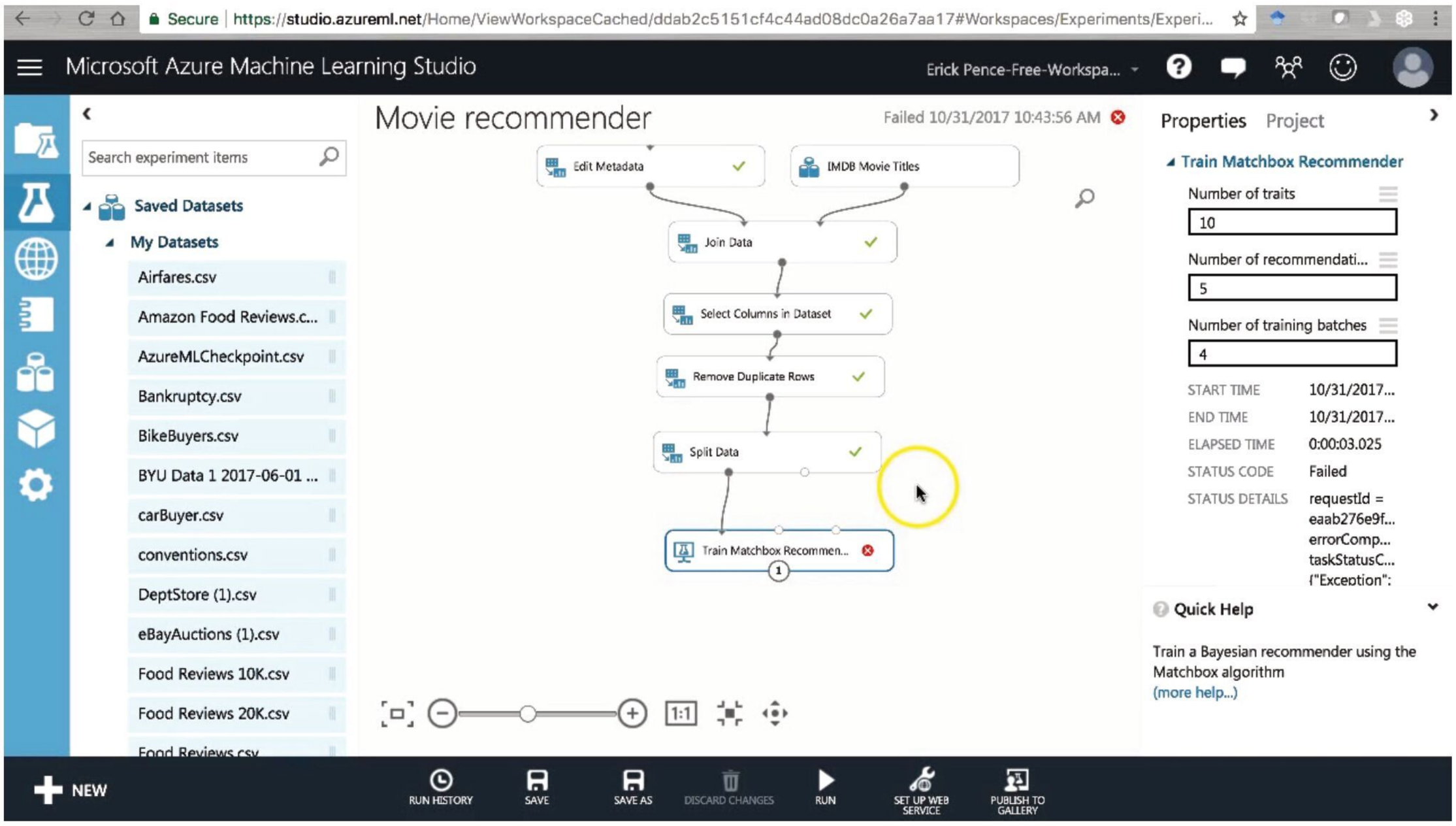


Figure 1: Microsoft Azure Machine Learning Studio

Azure ML Studio

Microsoft Azure Machine Learning (MAML) Studio is an online, collaborative, drag and drop tool for building machine learning models. It helps developers to write common machine learning algorithms as modules rather than write code in languages like Python or R, and build models graphically using data sets. Once the model is built, developers can fine-tune the hyper parameters of the algorithms. And on finalising the model, machine learning models are published as Web services to support custom apps or BI tools like Excel.

The Azure ML Studio menu is divided into three main sections: *Author*, *Assets*, and *Manage*. *Author* deals with the creation of the code and set up of machine learning processes. *Assets* are the resources that are created and stored within the *Author* section, such as the pipelines created in the Designer. They control the entire resources workflow, right from inputting data sets over a pipeline to the endpoints for the output (e.g.,

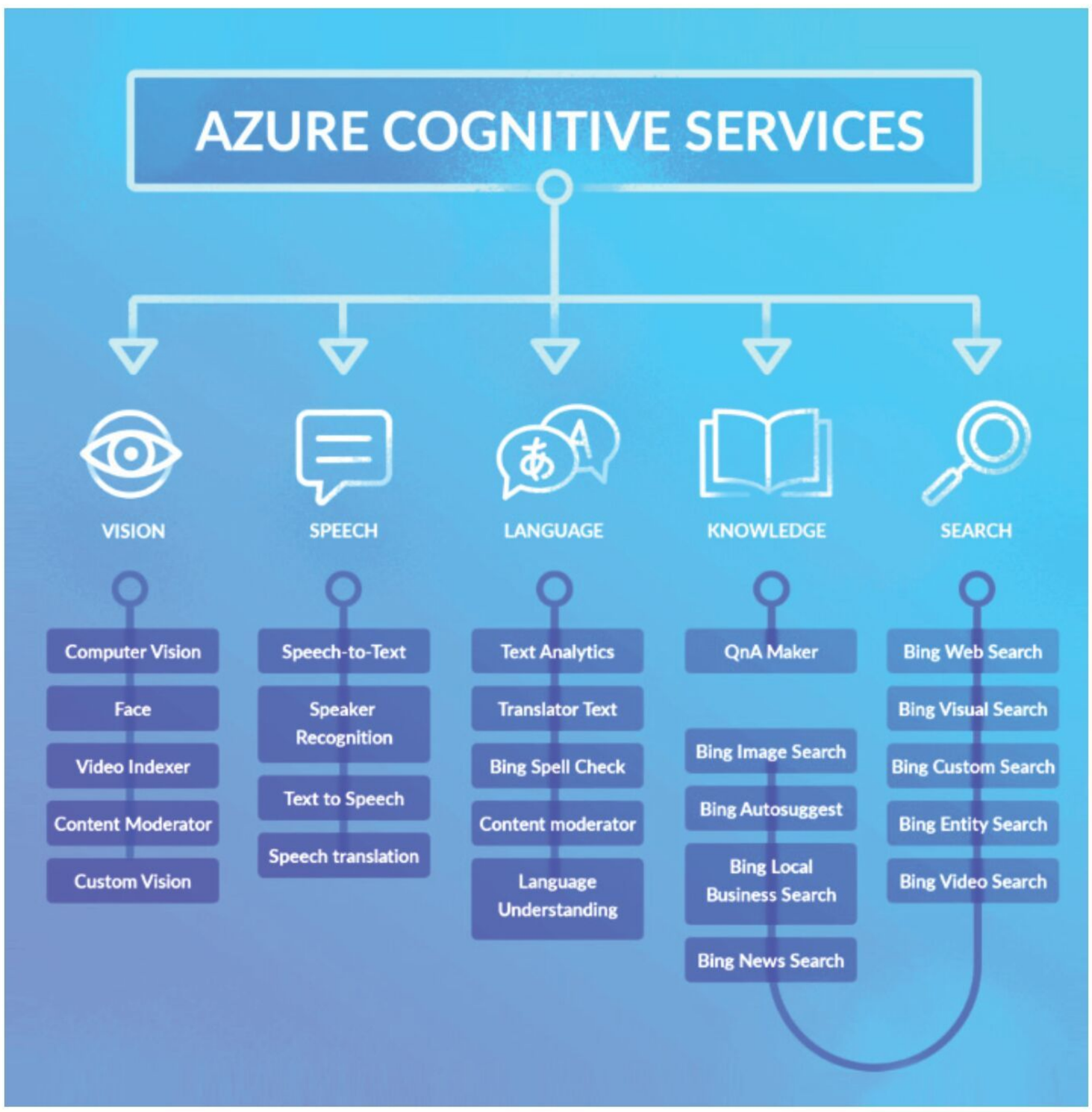


Figure 2: Microsoft Azure Cognitive Services

AI and ML services in the Fintech industry

We live in a data driven world, where it is impossible to secure applications or online services manually. AI based solutions are the need of the hour for anomaly detection in transactions and user services.

AI based solutions in Fintech enable greater security, improved customer experience and return on investment, and faster time to market. A few AI driven solutions being used in the Fintech industry are listed below.

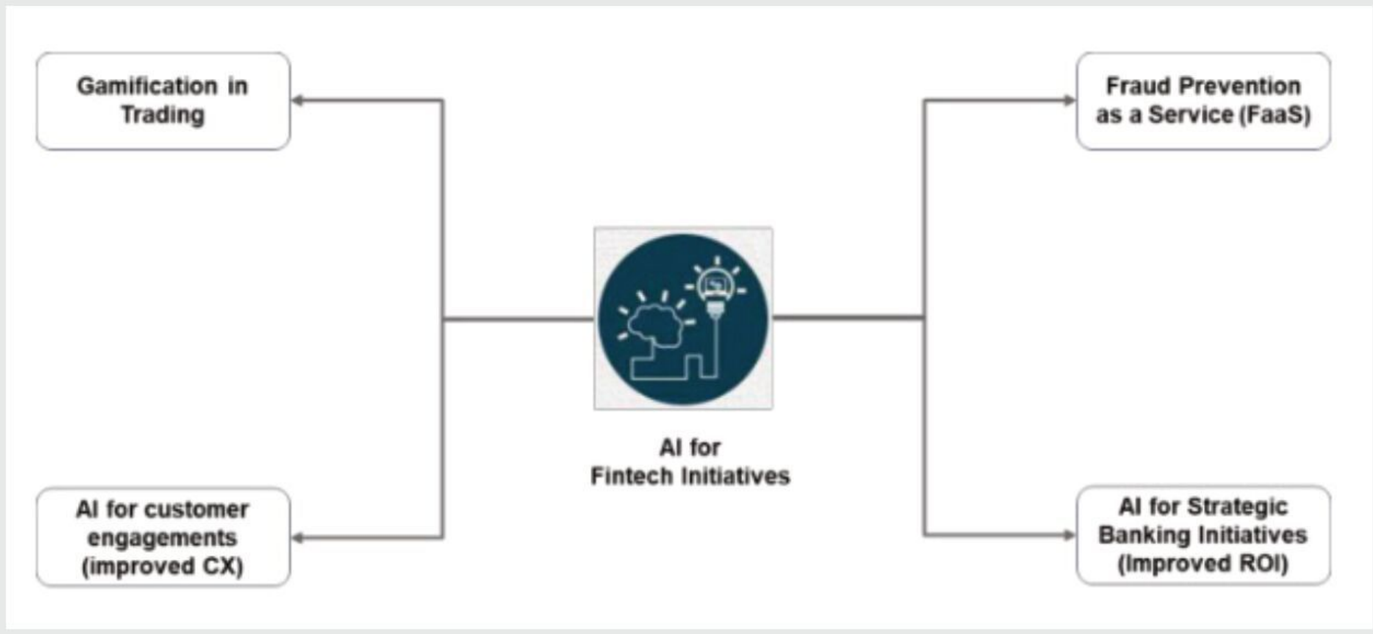


Figure 3: AI for Fintech use cases

1. ***Gamification in investment banking****:* Platforms like Betfair.com and WikiHow use AI based algorithms for intelligent gamification to predict prices for stocks based on various parameters like history, trending conditions, market need and stock movement. This is intelligent gambling in a three-minute cycle to give quotes for trading options.
2. ***Finance-as-a-Service:*** This is a new age service where cloud providers like Azure release Microsoft Cloud for financial services. Amazon FinSpace is another example. Open banking APIs and SaaS solutions for Fintech services (like Mambu) help to quickly get into digital banking services.
3. ***Fraud Prevention-as-a-Service:*** SaaS platforms like Sardine.ai give solutions for fraud detection using AI to analyse credit card transactions, customer credit history, and online shopping related frauds.

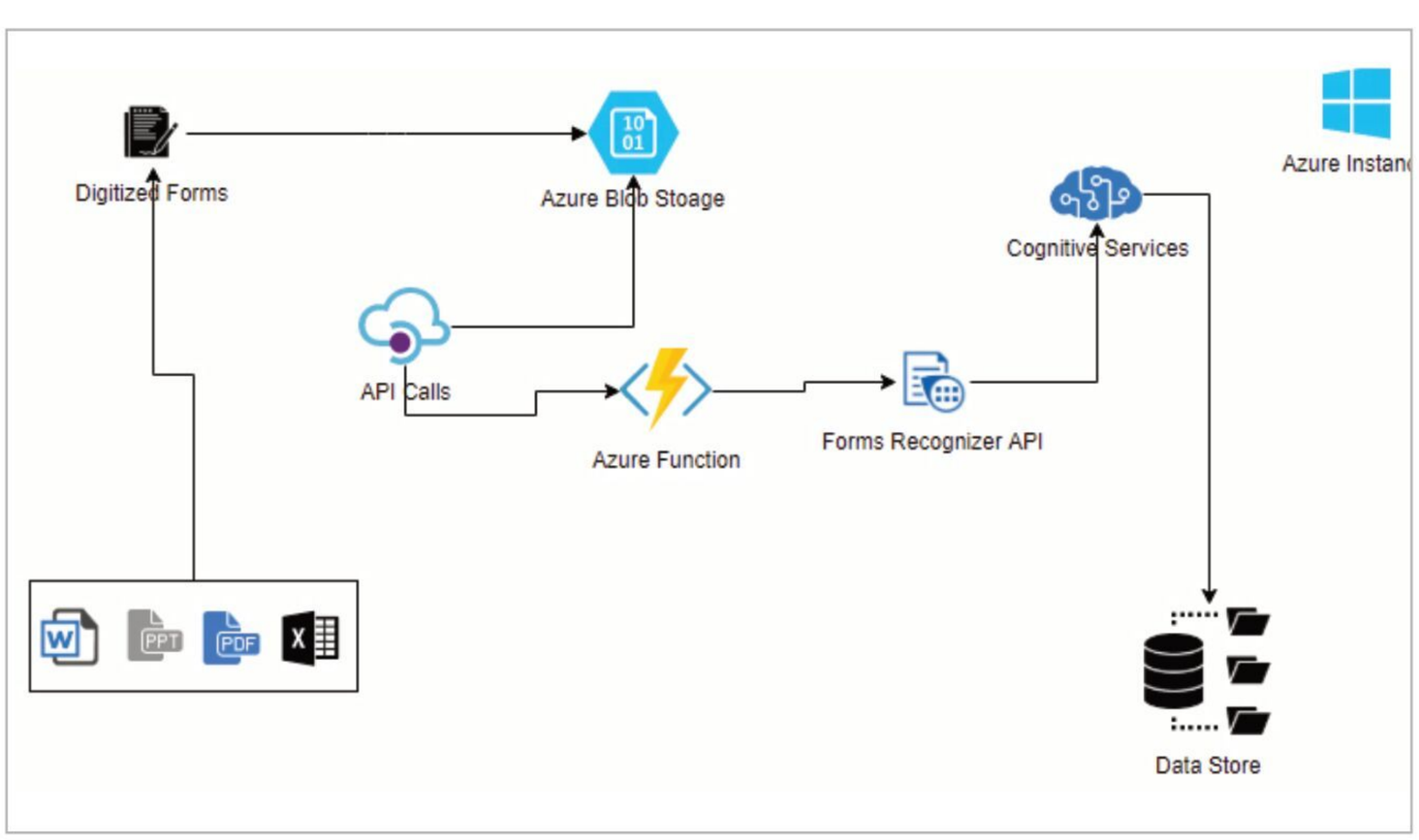


Figure 4: Azure Forms Recognizer solution for data extraction from forms

connection to real systems via REST API). The *Manage* section is for the system behind the scenes. It encompasses the computation clusters and instances, data stores on which the data sets are stored, and the integrations into other systems. It is an invisible layer.

Azure Machine Learning Studio creates a machine learning model in seven steps:

a. ***Data gathering***: Gathering of data starts the process, as data correction leads to perfect prediction models. MAML Studio provides a number of sample data sets, and additional collections can be downloaded from *archives.ics.uci.edu/ml/dataset.html*.
b. ***Data preview***: After uploading the data, the data is previewed to validate the data input.
c. ***Modules integration***: After viewing, modules are added by a simple drag and drop from the panel on the left.
d. ***Data preprocessing:*** This involves adjusting the available data as per nodes. Statistical data is computed from the available data.
e. ***Defining features***: In this step, the features of the data set that are most relevant to the result for prediction are determined.
f. ***Applying algorithms:*** The algorithm is applied to train the model.
g. ***Predicting:*** The prediction is tested and data is visualised in this step.

Azure Cognitive Services

Azure Cognitive Services are cloud based services with APIs (application programming interfaces) and SDKs (software development kits) which allow developers to integrate cognitive intelligence into applications even if they don't have a good knowledge of AI. With Azure Cognitive Services, various features like face recognition,

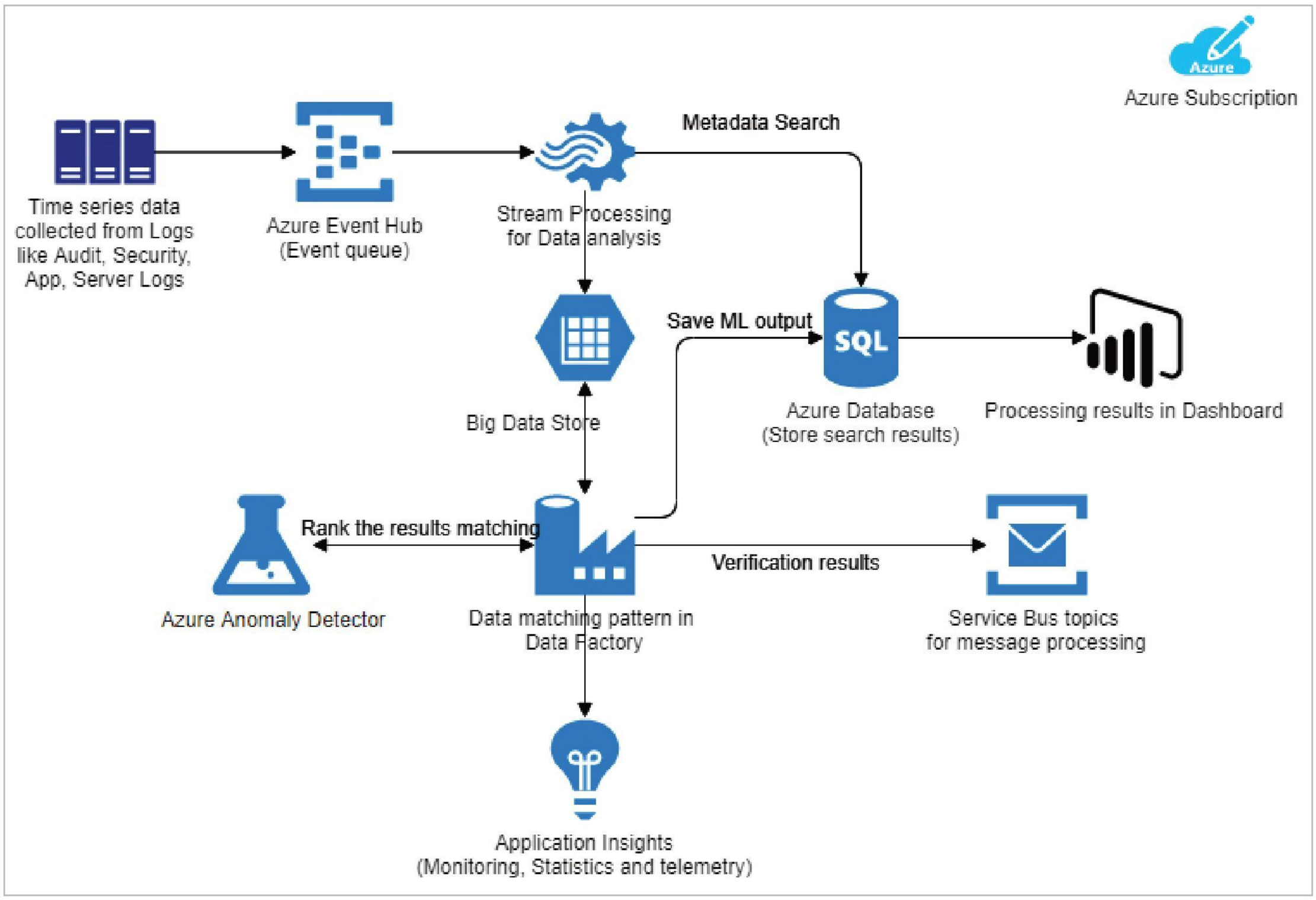


Figure 5: Azure Anomaly Detector architecture

speech recognition, emotion detection, and speech and language based advanced features can be added to applications.

These are available as REST APIs, client library SDKs and user interfaces. Azure Cognitive Services have tons of useful benefits for developers like:

a. Easy integration of pretrained and customised AI models in applications
b. Strong deployment of cognitive services from cloud to edge
c. Improved customer experience with simple and intuitive GUI in applications

Azure Form Recognizer

Azure has introduced a new service called Form Recognizer, which is built on cognitive services and OCR methods. It enables character extraction from pre-printed forms like invoices and bills. Currently, it works for printed character extraction and can be used to extract text from PDFs, office documents, and images. Feature enhancement is in progress for recognition of handwritten forms/ invoices/bills.

Azure ML Studio based OCR APIs is a supervised approach, where training and manual correction is required to improve the recognition rate of extracted text. Forms Recognizer is an unsupervised model, which improves the overall efficiency of the text recognition process.

Azure anomaly detection

Anomaly detection is a Big Data processing solution that aims at finding rare event patterns from a collection of patterns by creating pattern groups based on occurrences. For example, it can identify unusual login behaviour from a collection of audit logs, or unusual server failures from a server log.

In an enterprise, these logs are ever growing in size and have various patterns. Identifying which of these are valid events and which are suspicious ones is a big task.

Azure provides a sophisticated AI based service called Azure Anomaly Detector which uses API calls to use time series data like logs, enabling AI based detection of unusual event patterns. Before Azure introduced the Anomaly Detector, the Azure Machine Learning service was used to identify patterns of events, and use these for training the system to identify unusual behaviours.

With the introduction of anomaly detection, the system can self-train to build the knowledge base of patterns

of events. The configuration can also be customised for fine-tuning the sensitivity and ranking of behaviour.

Microsoft Machine Learning for Apache Spark (MMLSpark)

Microsoft Machine Learning for Apache Spark comprises tons of tools aimed at expanding the distributing computing framework Apache Spark in diverse directions. It adds deep learning and data science tools to Spark and even integrates Spark machine learning pipelines with the Microsoft Cognitive Toolkit (CNTK), LightGBM and OpenCV. This helps application designers to create powerful, scalable, predictive and analytical models for large images and text based data sets.

Some of its features are:

- Preprocessing of image data sets using OpenCV
- Strong integration of images from HDFS into Spark
- Image training using pretrained deep neural nets using CNTK
- Web services in Spark using HTTP on Apache Spark
- Classification and regression models training

Azure Language Understanding (LUIS)

Azure Language Understanding (LUIS) allows an application to understand what a person wants. It uses machine learning to help developers design applications that accept the input in natural language and extract the meaning from it. The application interacts with end users to pass the input into the LUIS app and translate it into relevant information to fetch the results back.

The technologies that LUIS works with are:

- *Bot framework:* This chatbot based API talks with the end user via a text input.
- *Bing Speech API:* This converts the natural language into text and then LUIS processes it.

There are three key concepts on which LUIS is based.

- ***Intents:*** An intent represents the actions a user wants to perform. The intent is a purpose or goal expressed in a user's input, such as booking a flight, paying a bill, or finding a news article. You define and name intents that correspond to these actions. A travel app may define an intent named 'BookFlight'.
- ***Utterances:*** An utterance is text input from the user that your app needs to understand. It may be a sentence, like 'Book a ticket to Paris', or a fragment of a sentence, like 'Booking' or 'Paris flight'. Utterances aren't always well-formed, and there can be many utterance variations for a particular intent.
- ***Entities:*** An entity represents detailed information that is relevant in the utterance. For example, in the utterance 'Book a ticket to Paris', 'Paris' is a location. By recognising and labelling the entities that are mentioned in the user's utterance, LUIS helps you to choose the specific action to take to answer a user's request.

LUIS has two ways to build a model — the Authoring APIs and the LUIS.ai Web app. Both methods give developers and their collaborators control of the LUIS model definition. One can use either LUIS.ai or the Authoring APIs or a combination of both to build a model. This management includes models, versions, collaborators, external APIs, testing, and training. Once the model is built and published, the utterance is passed to LUIS and the JSON object results are received with the endpoint APIs.

Microsoft Azure has introduced a range of AI and ML services in recent years to help developers and businesses tackle the large amounts of data being generated. Do take some time out to learn more about them (check the *References* below) to keep up with these fast changing times. END

References

[1] *https://docs.microsoft.com/en-us/azure/machine-learning/concept-automated-ml*
[2] *https://studio.azureml.net/*
[3] *https://docs.microsoft.com/en-us/azure/cognitive-services/what-are-cognitive-services*
[4] *https://azure.microsoft.com/en-us/services/cognitive-services/*
[5] *https://azure.microsoft.com/en-in/services/cognitive-services/form-recognizer/*
[6] *https://azure.microsoft.com/en-in/services/cognitive-services/anomaly-detector/*
[7] *https://docs.microsoft.com/en-us/azure/cognitive-services/text-analytics/how-tos/text-analytics-how-to-sentiment-analysis?tabs=version-3#opinion-mining*
[8] *https://techcommunity.microsoft.com/t5/educator-developer-blog/microsoft-machine-learning-for-apache-spark/ba-p/379152*
[9] *https://azure.microsoft.com/en-us/services/cognitive-services/conversational-language-understanding/*

By: Dr Anand Nayyar and Dr Magesh Kasthuri

Dr Anand Nayyar is a PhD in wireless sensor networks and swarms intelligence. He works at Duy Tan University, Vietnam, and loves to explore open source technologies, IoT, cloud computing, deep learning and cyber security.

Dr Magesh Kasthuri is a senior distinguished member of the technical staff and principal consultant at Wipro Ltd. This article expresses his views and not that of Wipro.

How to Analyse Sentiments Using Machine Learning

This article will help you understand the concept of sentiment analysis and learn how it is done. It uses different machine learning algorithms for sentiment analysis, and then compares them to decide which one is the best for the particular problem described here.

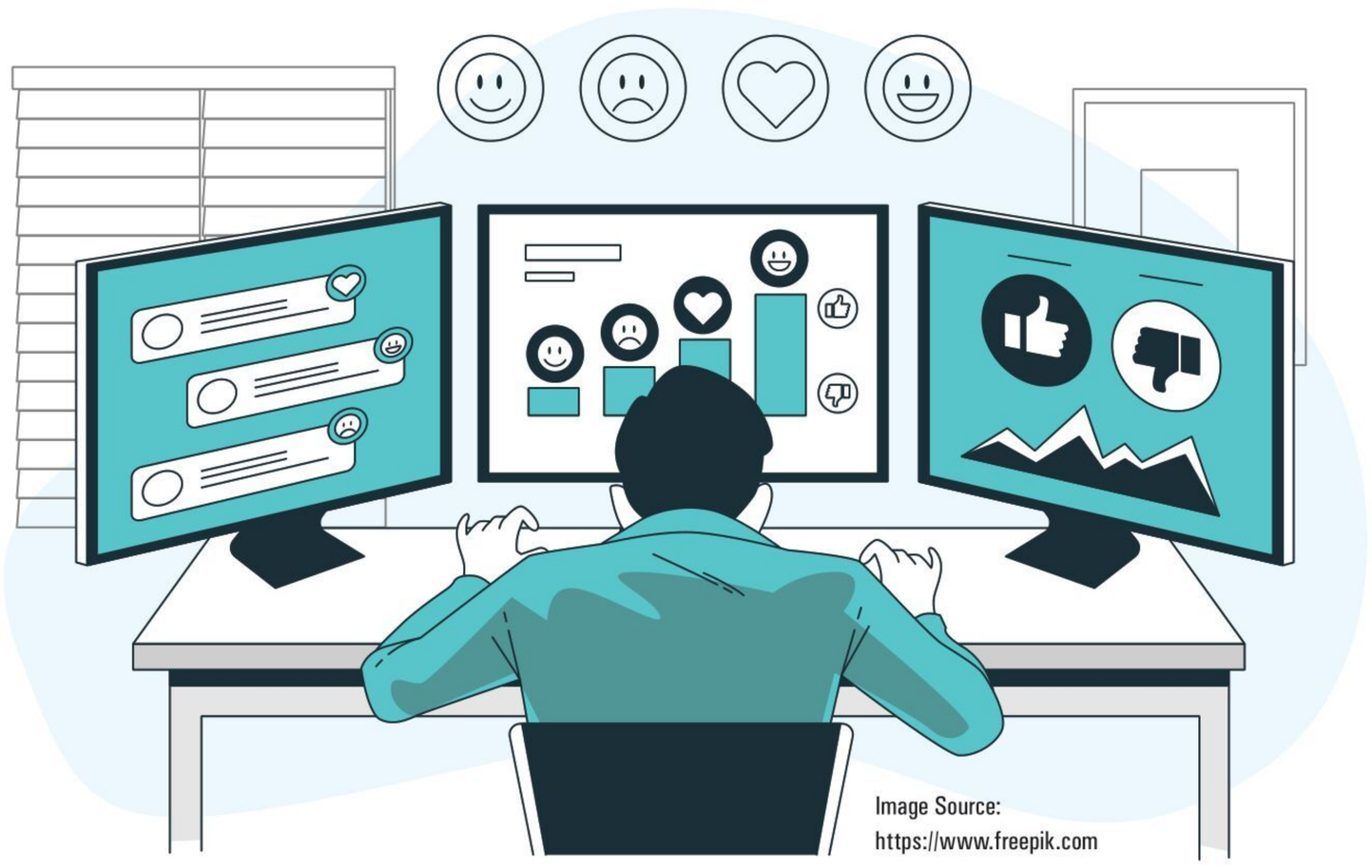


Image Source: https://www.freepik.com

Sentiment analysis is a major area in the field of natural language processing. A sentiment is any opinion or feeling that we have about an event, a product, a situation or anything else. Sentiment analysis is the field of research in which human sentiments are automatically extracted from the text. This field started evolving in the early 90s.

This article will help you understand how machine learning (ML) can be used for sentiment analysis, and compare the different ML algorithms that can be used. It does not try to improve the performance of any of the algorithms or methods.

In today's fast paced world, everything is online and everyone can post their views. A few negative online comments may hurt a company's reputation and, thereby, its sales. Now that everything's online, everyone can post their views and opinions. It becomes very important for companies to go through these to understand what their customers really want. But since there is so much data, it cannot be gone through manually. This is where sentiment analysis comes in.

Let us now start developing a model to do a basic sentiment analysis.

Let's start!

The first step is to select a data set. You can choose from any publicly available reviews or comments such as tweets or movie reviews. The two columns that should definitely be there in the data set are the label and the actual piece of text.

Figure 1 shows a small sample of how the data looks.

Now we need to import the required libraries:

```
import pandas as pd
import numpy as np
from nltk.stem.porter import
PorterStemmer
import re
import string
```

As you can see in the above code, we have imported NumPy and Pandas for processing the data. We will look at the other imported libraries when we use them.

Now that the data set is ready and the libraries are imported, we need to bring

negative	@jamielewislewis you are the worst, I just hate you, how could you do that to me??
positive	Staying positive is really important in life, stay happy!
positive	I really like the new design of your website!
positive	You really look good today, your ress is so beautiful
negative	Had a dream about a walk in fast food resturaunt that sold ice cream and guitar strings.They were sold out of positive2gauge
positive	The new design is awful!
positive	That is so sweet, thanks for doing that for me!

Figure 1: Data sample

the former into our project. The Pandas library is used for this purpose. We bring the data set into the Pandas data frame using the following line of code:

```
sentiment_dataframe = pd.read_csv("/
content/drive/MyDrive/Data/sentiments -
sentiments.tsv",sep = '\t')
```

Now that we have the data set in our project, let us manipulate it so that our algorithm can understand the features better. We begin by giving names to our columns in the data set. This is done by using the line of code given below:

```
sentiment_dataframe.columns =
["label","body_text"]
```

We then assign numerical labels to the classes — negative is replaced with 1 and positive is replaced with 0. Figure 2 shows how the data frame looks at this stage.

The next step is the preprocessing of the data. This is a very important step as it helps us to convert string/text data into numerical data (machine learning algorithms can understand/process numerical data and not text). Also, the redundant and useless data needs to be removed as it may taint our training model. We remove the noisy data, missing values and other non-consistent data in this step.

We will add the features text length and punctuation count in the data frame specifically for this application. We will also do the stemming, i.e., we will convert all similar words (like 'give', 'giving', etc) into a single form. Once this is done, we divide the data set into two -- X and Y -- where X is the *features* and Y is the *prediction* class.

This is done using the following piece of code. Figure 3 shows the data frame after these steps are taken.

```
def count_punct(text):
   count = sum([1 for char in text if
char in string.punctuation])
   return round(count/(len(text) -
text.count(" ")),3)*100

 tokenized_tweet = sentiment_
dataframe['body_text'].apply(lambda x:
x.split())
stemmer = PorterStemmer()
tokenized_tweet = tokenized_tweet.
apply(lambda x: [stemmer.stem(i) for i
in x])
for i in range(len(tokenized_tweet)):
   tokenized_tweet[i] = '
'.join(tokenized_tweet[i])
sentiment_dataframe['body_text'] =
tokenized_tweet
sentiment_dataframe['body_len'] =
sentiment_dataframe['body_text'].
apply(lambda x:len(x) - x.count(" "))
sentiment_dataframe['punct%'] =
sentiment_dataframe['body_text'].
apply(lambda x:count_punct(x))
X = sentiment_dataframe['body_text']
y = sentiment_dataframe['label']
```

We now need to convert the string into numerical data. We use a count vectorizer for this purpose; that is, we get the counts of each word and convert it into a vector.

After this, features such as length of text and punctuation count in the dataframe, i.e., X, are calculated. A sample of X is shown in Figure 4.

Now the data is ready for training. The next step is to determine which algorithms we are going to use for training our model. As has been mentioned before, we are going to try several algorithms and determine the best one for sentiment analysis. Since we are basically trying to do binary classification, the following algorithms can be used:

- K-nearest neighbors (KNN)
- Logistic regression
- Support vector machines (SVMs)
- Stochastic gradient descent
- Naive Bayes
- Decision tree
- Random Forest

We first need to split our data set into testing and training data. This is done by using the *sklearn* library using the following code:

```
from sklearn.model_selection import
train_test_split
X_train, X_test, y_train, y_test =
train_test_split(X,y, test_size = 0.20,
random_state = 99)
```

We will use 20 per cent of the data for testing and 80 per cent for the

[12] sentiment_dataframe.head()

	label	body_text
0	0	Staying positive is really important in life, ...
1	0	I really like the new design of your website!
2	0	You really look good today, your ress is so be...
3	1	Had a dream about a walk in fast food resturau...
4	0	The new design is awful!

Figure 2: Data frame with basic modifications

[32] sentiment_dataframe

	label	body_text	body_len	punct%
0	0	stay posit is realli import in life, stay happy!	40	5.0
1	0	I realli like the new design of your website!	37	2.7
2	0	you realli look good today, your ress is so be...	41	2.4
3	1	had a dream about a walk in fast food resturau...	102	1.0
4	0	the new design is awful!	20	5.0
...	...	...	...	...
2196	0	thi movi ha a lot of comedy, not dark and gord...	324	7.4
2197	1	have not watch kid film for some years, so I m...	1110	6.3
2198	1	you probabl heard thi phrase when it come to t...	928	5.5
2199	0	I wa about thirteen when thi movi came out on ...	464	9.3
2200	1	you are a fat person shitti ugli	26	0.0

2201 rows × 4 columns

Figure 3: Data frame after the division of the data set

[39] X

	body_len	punct%	0	1	2	3	4	5	6	7	...	13165	13166	13167	13168	13169	13170	13171	13172	13173	13174
0	40	5.0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
1	37	2.7	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
2	41	2.4	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
3	102	1.0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
4	20	5.0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
2196	324	7.4	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
2197	1110	6.3	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
2198	928	5.5	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
2199	464	9.3	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
2200	26	0.0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0

2201 rows × 13177 columns

Figure 4: Sample of final features

training part. We will separate the data because we want to test on a new set of data whether our model is working properly or not.

Now let us start with the first model. We will try the KNN algorithm first, and use the *sklearn* library for this. We will first train the model and then assess its performance (all of this can be done using the *sklearn* library in Python itself). The following piece of code does this, and we get an accuracy of around 50 per cent.

```
from sklearn.neighbors import
KNeighborsClassifier
model = KNeighborsClassifier (n_
neighbors=3)
model.fit(X_train, y_train)
model.score (X_test,y_test)
```

```
0.5056689342403629
```

The code is similar in the logistic regression model — we first import the function from the library, fit the model, and then test it. The following piece of code uses the logistic regression algorithm. The output shows we got an accuracy of around 66 per cent.

```
from sklearn.linear_model import
LogisticRegression
model = LogisticRegression()
model.fit (X_train,y_train)
model.score (X_test,y_test)
```

```
0.6621315192743764
```

The following piece of code uses SVM. The output shows we got an accuracy of around 67 per cent.

```
from sklearn import svm
model = svm.SVC(kernel='linear')
model.fit(X_train, y_train)
model.score(X_test,y_test)
```

```
0.6780045351473923
```

The following piece of code uses the Random Forest algorithm, and we get an accuracy of around 69 per cent.

```
from sklearn.ensemble import
RandomForestClassifier
model = RandomForestClassifier()
model.fit(X_train, y_train)
model.score(X_test,y_test)
```

```
0.6938775510204082
```

Next we use the Decision tree algorithm, which gives an accuracy of around 61 per cent.

```
from sklearn.tree import
DecisionTreeClassifier
model = DecisionTreeClassifier()
```

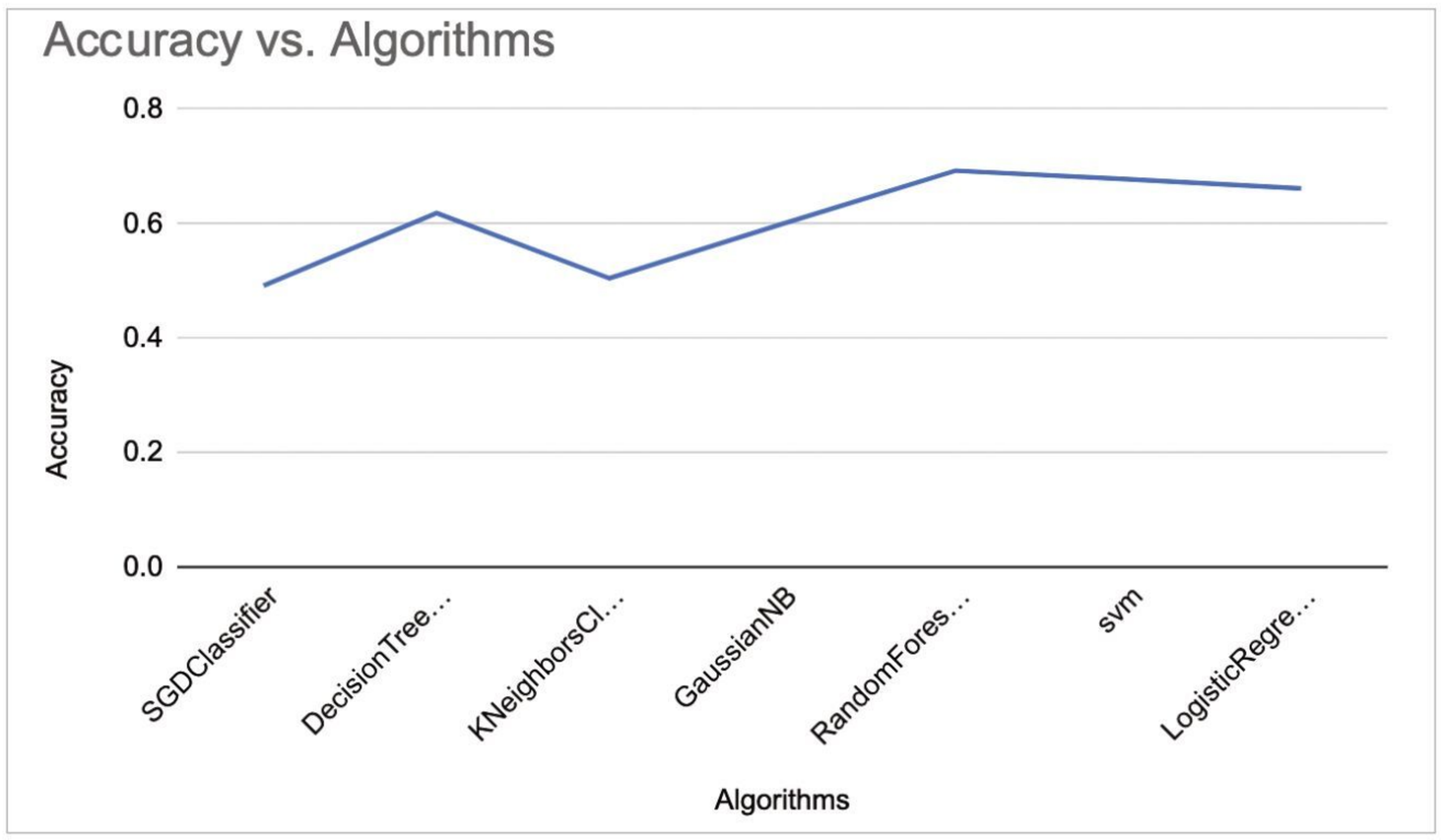


Figure 5: Accuracy performance of the different algorithms

```
[77] data = ["You are bad, you are the worst!"]
     vect = pd.DataFrame(countvect.transform(data).toarray())
     body_len = pd.DataFrame([len(data) - data.count(" ")])
     punct = pd.DataFrame([count_punct(data)])
     total_data = pd.concat([body_len,punct,vect],axis = 1)
     my_prediction = model.predict(total_data)
     print(my_prediction)

     [1]

     data = ["You are amazing !!"]
     vect = pd.DataFrame(countvect.transform(data).toarray())
     body_len = pd.DataFrame([len(data) - data.count(" ")])
     punct = pd.DataFrame([count_punct(data)])
     total_data = pd.concat([body_len,punct,vect],axis = 1)
     my_prediction = model.predict(total_data)
     print(my_prediction)

     [0]
```

Figure 6: Sample predictions made

```
model = model.fit(X_train,y_train)
model.score(X_test,y_test)

0.6190476190476191
```

The following piece of code uses the stochastic gradient descent algorithm. The output shows that we got an accuracy of around 49 per cent.

```
from sklearn.linear_model import
SGDClassifier
model = SGDClassifier()
model = model.fit(X_train,y_train)
model.score(X_test,y_test)

0.49206349206349204
```

The following piece of code uses Naive Bayes. We get an accuracy of around 60 per cent.

```
from sklearn.naive_bayes import
GaussianNB
model = GaussianNB()
model.fit(X_train, y_train)
model.score(X_test,y_test)

0.6009070294784581
```

Now that we have checked out all the algorithms, let us graph their accuracy performance. The graph is shown in Figure 5.

As you can see, the random forest algorithm gave the best accuracy for this problem and we can conclude that it is the best fit for sentiment analysis amongst ML algorithms. We can improve the accuracy much more by getting better features, trying out other vectorising techniques, and using a better data set or newer classification algorithms.

Now that random forest is seen as the best algorithm for this problem, I am going to show you a sample prediction. In Figure 6, you can see that the right predictions are being made! Do try this out to improve upon this project! END

By: Jishnu Saurav Mittapalli

The author is a full stack Web developer, interested in modern technologies like artificial intelligence and machine learning. He is currently working in the field of natural language processing.

OSFY Magazine Attractions During 2022-23

Month	Theme
March 2022	Security, Network Management and Monitoring
April 2022	Open Source Programming (Languages and tools)
May 2022	DevOps Special
June 2022	AI, Deep learning and Machine Learning
July 2022	Database management and Optimisation
August 2022	Cloud Special: Everything from management to implementation
September 2022	Mobile/Web App Development, Optimisation and Security
October 2022	Blockchain and Open Source
November 2022	Open Source and IoT and Edge
December 2022	Data science, Data security, Data Management,Storage and Backup
January 2023	Docker and containers
February 2023	Open Source on Windows and Best in the world of Open Source (Tools and Services)

How to Run a Kubernetes Cluster Locally with Kind

Kind is a tool used for running Kubernetes (k8s) locally using Docker container 'nodes'. Its primary purpose is to test k8s, but it can also be used in your CI system or for local development.



Kind serves various purposes in the Kubernetes ecosystem depending upon your use case. It can be used to learn Kubernetes, to make sure that your Kubernetes application manifests are correct, or to test your deployments locally. In this article, we will see how to use Kind to create a Kubernetes cluster locally and run a Web application using it.

Let's invoke the Kind executable to see what are the available options:

```
> kind
kind creates and manages local Kubernetes clusters using
Docker container 'nodes'

Usage:
  kind [command]

Available Commands:
  build       Build one of [node-image]
  completion  Output shell completion code for the specified
shell (bash, zsh or fish)
  create      Creates one of [cluster]
  delete      Deletes one of [cluster]
  export      Exports one of [kubeconfig, logs]
  get         Gets one of [clusters, nodes, kubeconfig]
  help        Help about any command
  load        Loads images into nodes
  version     Prints the kind CLI version

Flags:
  -h, --help              help for kind
      --loglevel string   DEPRECATED: see -v instead
  -q, --quiet             silence all stderr output
  -v, --verbosity int32   info log verbosity
      --version           version for kind

Use "kind [command] --help" for more information about a
command.
```

Cluster creation and interaction

Now let's create a Kubernetes cluster locally and interact with it:

```
> kind create cluster
Creating cluster "kind" ...
 > Ensuring node image (kindest/node:v1.21.1) >
 > Preparing nodes >
 > Writing configuration >
 > Starting control-plane >
 > Installing CNI >
 > Installing StorageClass >
Set kubectl context to "kind-kind"
```

You can now use your cluster with:

```
kubectl cluster-info --context kind-kind
```

Not sure what to do next? Check out *https://kind.sigs.k8s.io/docs/user/quick-start/*.

The above command will bootstrap a k8s cluster using a pre-built node image. In fact, once the above command completes, you can use Docker to verify that a new image has indeed been downloaded:

```
> docker images | rg kind
kindest/node                                    <none>
32b8b755dee8   8 months ago    1.12GB
```

The default cluster name is *kind*. And you can confirm it using the following command:

```
> kind get clusters
kind
```

You can also get it's *kubeconfig* via the following command:

```
> kind get kubeconfig | head
apiVersion: v1
clusters:
- cluster:
    certificate-authority-data: LS0tLS1CRUdJTiBDRVJUSUZJQ0FUR
S0tLS0tCk1JSUM1ekNDQWMrZ0F3 SUJBZ0lCQURBTkJna3Foa2l
HOXcwQkFRc0ZBREFWTVJNd0VRWURWUVFERXdwcm
RXSmwKY201bGRHVnpNQjRYRF RJeU1ESXdOREUxTlRNeU1sb1hEVE15TURJd0
1qRTFOV<Snipped>
OG9QRDNoYnRraFJ4MQotLS0tL UVORCBDRVJUSUZJQ0FURS0tLS0tCg==
    server: https://127.0.0.1:38471
  name: kind-kind
contexts:
- context:
    cluster: kind-kind
    user: kind-kind
```

You can use the *--wait* flag if you want to wait till the control plane is in a reachable state. Since Kind is mostly used for testing, you cannot actually update it. Let's say we want to have a declarative setup with this configuration for our Kind cluster:

```
kind: Cluster
apiVersion: kind.x-k8s.io/v1alpha4
nodes:
- role: control-plane
  kubeadmConfigPatches:
  - |
    kind: InitConfiguration
    nodeRegistration:
      kubeletExtraArgs:
        node-labels: "ingress-ready=true"
  extraPortMappings:
  - containerPort: 80
    hostPort: 80
    protocol: TCP
  - containerPort: 443
    hostPort: 443
    protocol: TCP
```

The above configuration creates a Kind cluster with extraPortMappings and node-labels. The port mappings are required to allow localhost to make requests to the Ingress controller. Let's try to apply the above manifest (but note that it's going to fail):

```
> kind create cluster --config ./cluster.yaml
ERROR: failed to create cluster: node(s) already exist for
a cluster with the name "kind"
```

So we have to delete and create a new one. Let's delete it first:

```
▫ kind delete cluster
Deleting cluster "kind" ...
```

Let's create the cluster using our declarative setup now:

```
> kind create cluster --config ./cluster.yaml
Creating cluster "kind" ...
 > Ensuring node image (kindest/node:v1.21.1) >
 > Preparing nodes >
 > Writing configuration >
 > Starting control-plane >
 > Installing CNI >
 > Installing StorageClass >
Set kubectl context to "kind-kind"
```

You can now use your cluster with:

```
kubectl cluster-info --context kind-kind
```

You can also specify your k8s version in the above manifest by specifying the proper Docker image, which you can find in its GitHub release page:

```
- role: control-plane
  image: kindest/node:v1.21.1@sha256:69860bda5563ac81e3c005
7d654b5253219618a22ec3a346306239bba8cfa1a6
```

Cluster interaction

We need to interact with the cluster to get the actual job done. This could be deploying your application or doing maintenance operations on your node. You can use the usual *kubectl* to interact with your cluster:

```
> kubectl get nodes
NAME                 STATUS   ROLES                  AGE     VERSION
kind-control-plane   Ready    control-plane,master   95s
v1.21.1
```

Now let's check the various name spaces present:

```
> kubectl get namespaces -A
NAME                 STATUS   AGE
default              Active   3m1s
kube-node-lease      Active   3m2s
kube-public          Active   3m2s
kube-system          Active   3m2s
```

```
local-path-storage   Active   2m58s
```

And let's check the various logs that are running:

```
> kubectl get pods -A
NAMESPACE            NAME
READY   STATUS    RESTARTS   AGE
kube-system          coredns-558bd4d5db-7pzbj
1/1     Running   0          3m25s
kube-system          coredns-558bd4d5db-vgf9l
1/1     Running   0          3m25s
kube-system          etcd-kind-control-plane
1/1     Running   0          3m28s
kube-system          kindnet-jq54g
1/1     Running   0          3m25s
kube-system          kube-apiserver-kind-control-plane
1/1     Running   0          3m28s
kube-system          kube-controller-manager-kind-control-plane
1/1     Running   0          3m28s
kube-system          kube-proxy-bkxql
1/1     Running   0          3m25s
kube-system          kube-scheduler-kind-control-plane
1/1     Running   0          3m28s
local-path-storage   local-path-provisioner-547f784dff-2vkf4
1/1     Running   0          3m25s
```

Loading a Docker image: To load a Docker image, give the following command:

```
kind load docker-image image_1 image_2
```

Note that once you load the image, you still have to create the Deployment k8s resource, which will typically reference the loaded image and then do kubectl apply of the resource.

NGINX Ingress controller: Now let's deploy Ingress NGINX to our cluster:

```
> kubectl apply -f https://raw.githubusercontent.com/
kubernetes/ingress-nginx/main/deploy/static/provider/kind/
deploy.yaml
namespace/ingress-nginx created
serviceaccount/ingress-nginx created
serviceaccount/ingress-nginx-admission created
role.rbac.authorization.k8s.io/ingress-nginx created
role.rbac.authorization.k8s.io/ingress-nginx-admission created
clusterrole.rbac.authorization.k8s.io/ingress-nginx created
clusterrole.rbac.authorization.k8s.io/ingress-nginx-admission
created
rolebinding.rbac.authorization.k8s.io/ingress-nginx created
rolebinding.rbac.authorization.k8s.io/ingress-nginx-admission
created
clusterrolebinding.rbac.authorization.k8s.io/ingress-nginx
created
clusterrolebinding.rbac.authorization.k8s.io/ingress-nginx-
admission created
configmap/ingress-nginx-controller created
service/ingress-nginx-controller created
service/ingress-nginx-controller-admission created
deployment.apps/ingress-nginx-controller created
job.batch/ingress-nginx-admission-create created
job.batch/ingress-nginx-admission-patch created
ingressclass.networking.k8s.io/nginx created
validatingwebhookconfiguration.admissionregistration.k8s.io/
ingress-nginx-admission created
```

Next, let's inspect our NGINX pods:

```
> kubectl get pods -n ingress-nginx
NAME                                READY   STATUS              RESTARTS   AGE
ingress-nginx-admission-create-hrmn5        0/1     Completed
0          47s
ingress-nginx-admission-patch-7ds99         0/1     Completed
1          47s
ingress-nginx-controller-674cb6ff57-tmjbc   0/1
ContainerCreating   0          47s
```

Let's check the service of the NGINX controller:

```
▫ kubectl get services -n ingress-nginx
NAME                                 TYPE        CLUSTER-IP
EXTERNAL-IP   PORT(S)                      AGE
ingress-nginx-controller             NodePort    10.96.65.28
<none>        80:32544/TCP,443:30304/TCP   116s
ingress-nginx-controller-admission   ClusterIP   10.96.207.79
<none>        443/TCP                      116s
```

You can see the NGINX controller is listening on both port 80 and 443. Now let's check what our Ingress class is:

```
▫ kubectl get ingressclass -A
NAME    CONTROLLER             PARAMETERS   AGE
nginx   k8s.io/ingress-nginx   <none>       8m26s
```

Deploying an application

Let's deploy the *https://httpbin.org/* application inside our Kind cluster. I have the manifest for the application in the app_k8s repository: *https://github.com/psibi/app_k8s*.

Rendering the resource for the application will result in this:

```
▫ kustomize build overlays/app_nginx
apiVersion: v1
kind: Namespace
metadata:
```

```
  name: base-app
---
apiVersion: v1
kind: Service
metadata:
  labels:
    app: httpbin
  name: httpbin
  namespace: base-app
spec:
  ports:
  - name: http
    port: 8000
    targetPort: 80
  selector:
    app: httpbin
  type: ClusterIP
---
apiVersion: apps/v1
kind: Deployment
metadata:
  name: httpbin
  namespace: base-app
spec:
  replicas: 1
selector:
  matchLabels:
    app: httpbin
    version: v1
template:
  metadata:
    labels:
      app: httpbin
      version: v1
  spec:
    containers:
    - image: docker.io/kennethreitz/httpbin
      imagePullPolicy: IfNotPresent
      name: httpbin
      ports:
      - containerPort: 80
---
apiVersion: networking.k8s.io/v1
kind: Ingress
metadata:
  name: ingress-httpbin
  namespace: base-app
spec:
  ingressClassName: nginx
  rules:
  - http:
      paths:
  - backend:
      service:
        name: httpbin
        port:
          number: 8000
    path: /
    pathType: Prefix
```

Once done, apply the above application:

```
▫ kubectl apply -f application.yaml
namespace/base-app created
service/httpbin created
deployment.apps/httpbin created
ingress.networking.k8s.io/ingress-httpbin created
```

Now let's check the Ingress:

```
▫ kubectl get ingress -n base-app
NAME              CLASS   HOSTS   ADDRESS     PORTS   AGE
ingress-httpbin   nginx   *       localhost   80      3m17s
```

Let's check that the application is indeed working:

```
▫ curl -s http://localhost | head
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <title>httpbin.org</title>
    <link href="https://fonts.googleapis.com/css?family
=Open+Sans:400,700|Source+Code+Pro:300,600|Titillium+W
eb:400,600,700"
        rel="stylesheet">
    <link rel="stylesheet" type="text/css" href="/flasgger_
static/swagger-ui.css">
    <link rel="icon" type="image/png" href="/static/favicon.
ico" sizes="64x64 32x32 16x16" />
```

This summarises how to use Kind to create a Kubernetes cluster, deploy a NGINX controller, and then use that to deploy a Web application on it. END

References

[1] Sample application repository: *https://github.com/psibi/app_k8s*

[2] Kind homepage: *https://kind.sigs.k8s.io/*

By: Sibi Prabakaran

The author is a senior engineer working at FPComplete.

Get Ready to Embrace the AIOps Era

Businesses, no matter how small or large, must transform themselves into digital firms as technology advances. It is no longer a question of 'choosing' to transform; rather, it is about 'how' to move forward with the transition. This is where AIOps comes in.



There are a lot of challenges that come with transforming an organisation into a digital firm. Lack of dedicated IT skills, organisational change management, evolving customer needs, and hybrid environments are just a few of these. Businesses need to enhance their IT operations (ITOps) to meet these challenges and fulfill customer expectations.

Digital transformation: The way to AIOps

In the future, ITOps will combine algorithmic and human intelligence to make the performance of IT systems transparent and help them deliver a seamless experience.

> "The long-term impact of AIOps on IT operations will be transformative."
>
> – *Gartner*

AIOps will be essential for successful digital transformation to help systems perform at the velocity that modern business demands. This, in turn, will establish how quickly a company acquires and retains market leadership.

Modern AIOps technologies offer IT infrastructure on-premises, on the cloud, or in a hybrid environment. Their automation allows developers to focus on designing next-generation business apps rather than be concerned about the underlying infrastructure.

What is AIOps?

AIOps combines artificial intelligence and machine learning to analyse data for IT operations. It is the process of applying AI to IT operations. It helps organisations detect errors proactively and also helps operations teams prevent issues before they impact end users.

Traditional IT management techniques have been unable to handle digital business transformation. When Gartner coined the term AIOps, the company predicted that there would be significant changes in the process

of IT operations as well as a massive change in how organisations manage their IT ecosystem.

AIOps platforms use Big Data. They collect the data from various IT operations and devices to automatically identify and respond to problems in real-time while still delivering traditional historical analytics. AIOps then executes comprehensive analytics using machine learning for the combined IT data.

Automation-driven insights that lead to continuous improvements and repairs are the outcome. Continuous integration and deployment (CI/CD) for essential IT functions is supported in AIOps.

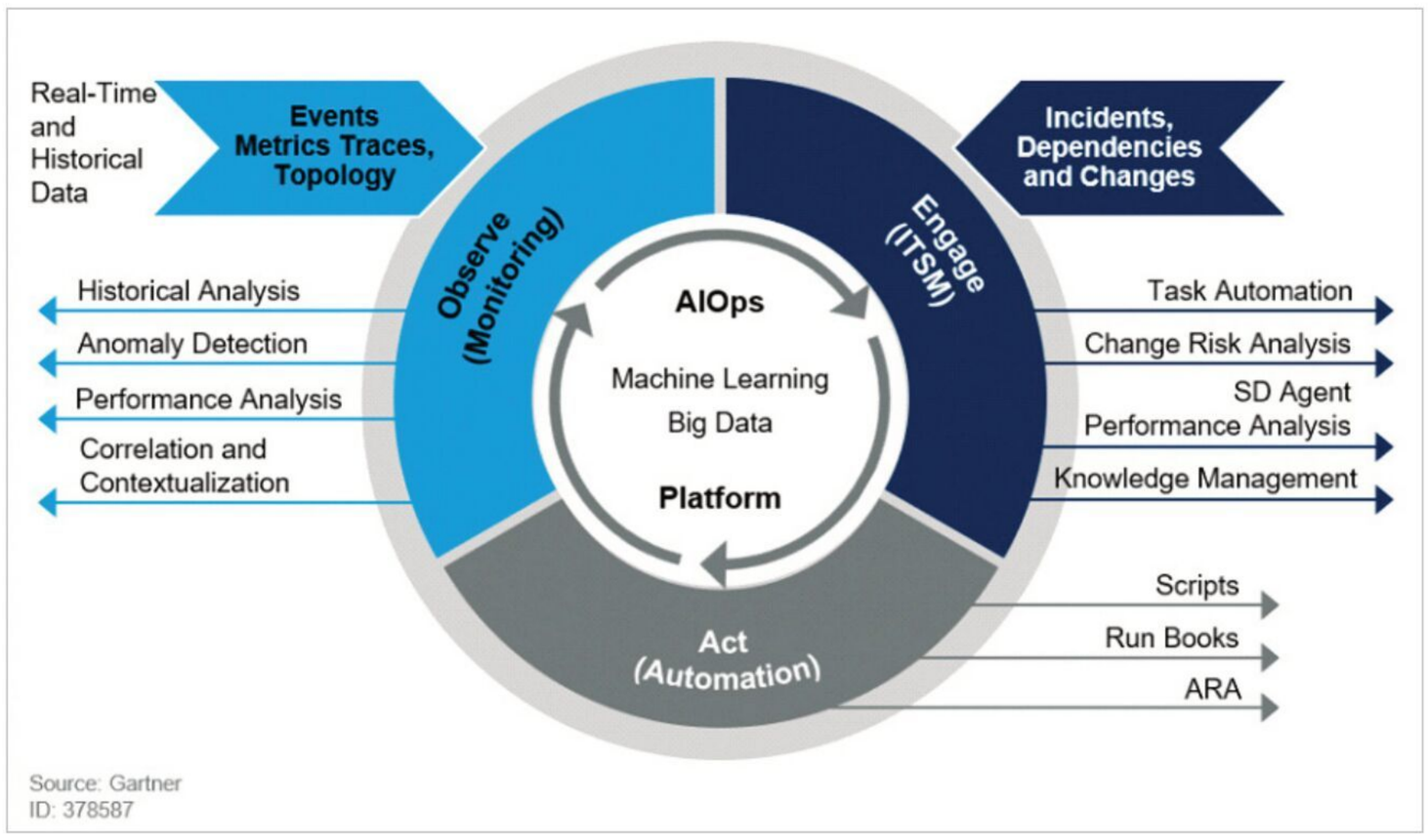


Figure 1: AIOps platform enabling continuous insight across IT operations monitoring (ITOM) (Source: *https://www.gartner.com/en/documents/3971186*)

What is the scope for AIOps?

Getting started with AIOps may seem challenging initially, so it is best to take a step-by-step approach. We need to begin by identifying and understanding IT operational data. AIOps is data-driven at its core. Therefore, it needs access to all relevant operations data, including unstructured data, logs, metrics, real-time data, API outputs, and device data. Structured business data is also needed, such as databases, social activities, and other relational data. The more relevant data AIOps platforms consume, the more accurate their predictions will be.

Businesses need to understand how their data can help solve their biggest problems no matter which industry they are in. They should implement AIOps to review their past failures and identify what data will help them arrive at a solution.

Data analytics can be used to find the source of outages or system slowdowns. Then the AIOps platform can be used for detecting common issues and their remedies. Businesses can use the collected insight to implement machine learning and AI for real-time monitoring and automated response. This entire process helps companies drive AI maturity, effectively fixing errors, avoiding downtime, and enhancing efficiency.

Why is AIOps needed in 2022?

As businesses hasten their digital transformation, applications and system architectures become highly advanced. We can witness this complexity in a few ways:

- Organisations are transferring from traditional application architecture to a cloud-native, flexible, and microservices based containerised application stack.
- These applications are also being deployed on on-premises, hybrid, public, and private cloud platforms.

As applications and IT environments expand, they produce tremendous amounts of data. IT operations teams get exhausted with data they cannot manage. However, AI can consume massive amounts of data. As the volume of data expands, the opportunities for AI to be incorporated into IT processes are much greater.

Anomaly detection, categorisation, and prediction can all be done by using machine learning and deep learning models, which excel at analysing massive volumes of data and delivering insights. Many features of AIOps help firms provide a good user experience through an interactive dashboard.

Businesses implementing AIOps have reported benefits such as frictionless experiences, lower operating expenses, faster customer service, shorter mean time to resolution, and fewer downtimes. AIOps supports IT operations by taking firm decisions based on predictive analytics.

A final note

AIOps is the next step in IT operations analytics (ITOA). AI, cognitive skills, and RPA (robotic process automation) are used to automate the remediation of infrastructure or IT operations' issues before they become a problem. Self-healing systems are the ultimate goal of AIOps. END

By: Amit Shingala

The author is the co-founder and CEO of Motadata (Mindarray Systems Pvt Ltd). He holds over 13 plus years of experience in building, scaling, and selling enterprise products.

EFYGROUP
YOURS SINCE 1969

Thank You IIT Madras

It was in your hostel room that the idea was conceived



The first issue in Jan '69



The Jan '22 Issue

Celebrating 54 years in TECH media

Magazines
- Electronics For You
- Electronics Bazaar
- Open Source For You

Portals
- www.electronicsforu.com
- www.opensourceforu.com
- www.eleb2b.com
- www.electronicsb2b.com

Events
- India Electronics Week
  - IOTshow.in
  - EFY Expo
  - Test & Measurement India
- Open Source India
- Techworldcongress.com

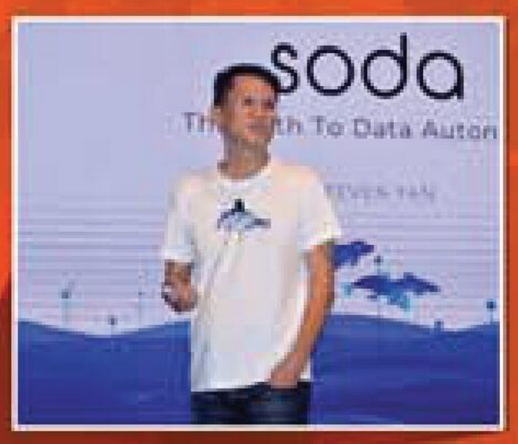
EFY Enterprises Pvt Ltd
D-87/1, Okhla Industrial Area, Phase-1, New Delhi-110020

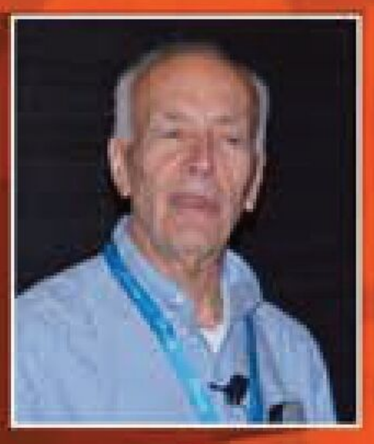
• Phone: +91-98111-55335
• Email: info@efy.in • Website: www.efy.in

R N I No. DELENG/2012/49440, Mailed on 27/28th of Advance month
Published on 27th June 2022
Delhi Postal Regd. No. DL(S)-01/3443/2022-24
Licenced to Post without Pre-Payment Licence No U(S)-56/2022-24
Posted at LPC Delhi RMS Delhi 110006

An EFYGROUP EVENT

ACCELERATING DEVELOPMENT AND DEPLOYMENT OF OPEN SOURCE









19th Edition
OPEN SOURCE INDIA
29-30 September 2022
NIMHANS CONVENTION CENTER, BENGALURU

Asia's #1 Conference on Open Source

Register Now!

www.OpenSourceIndia.in

For more details, call on +91 96677-43666 or write to info@opensourceindia.in

https://opensourceindia.in





