Ansible AWX: The GUI Configuration Management Automation Tool You Will Love To Use

Building Amazon Machine Images With HashiCorp Packer

₹120
ISSN-2456-4885

Open Source For You

Volume: 11 | Issue: 08 | Pages: 100 | June 2023

THE COMPLETE MAGAZINE ON OPEN SOURCE

An EFY GROUP Publication

What's Making Generative AI So Popular?

Use AIOps To Make Your Enterprise Agile

Running Generative AI Models On Your Local Box Is Good Fun

Making A Difference With Generative AI

"For us, stability and scalability are the key aspects of open source platforms"

—**Mani Ganeshan,** global head of engineering, travel distribution and centre head at Amadeus Labs

Electronics ForYou expo

Innovate. Design. Manufacture. Source.

10-11 AUGUST 2023
Manekshaw Centre,
Parade Road, Delhi Cantt., Delhi

delhi.EFYEXPO.com

Delhi's 1st Electronics Event

Focused on New Product Development & Manufacturing

With Conferences on Defence Electronics, Power Electronics, EVs, Batteries, Charging Systems and Energy Storage Systems (ESS) including UPS & Inverters.

A NEW PRODUCT DEVELOPMENT EVENT @ DELHI + NCR

Developing new electronics products and manufacturing them can be a daunting challenge in nations like India, where the electronics eco-system is getting established.

EFY Expo aims to solve this challenge for you by bringing in India's leading suppliers who can assist you in 'new product development' and 'manufacturing them'—even if your initial batch is of ONE unit only!

For more information on sponsoring and exhibiting

Call: Ms Mameeta (+91-95998-14784)
Email: growmybiz@efy.in

MEDIA PARTNERS

electronics FOR YOU | electronicsforu.com It's electronics, it's here | electronicsForYou EXPRESS | electronics B2B.com | Electronics Industry Directory | 99 | 99 E-VEHICLE

CONTENTS

JUNE 2023 | ISSN-2456-4885

FOCUS

28 Power in Your Hands: Running Local Large Language Models for AI Brilliance

35 An Introduction to MLOps

38 AI: Beyond Python

44 Implementing a CNN Deep Learning Model with TensorFlow

47 Use AIOps to Make Your Enterprise Agile

60 A Quick Look at Deep Learning

DEVELOPERS

66 Building a Cross-Platform Mobile Application with Flutter

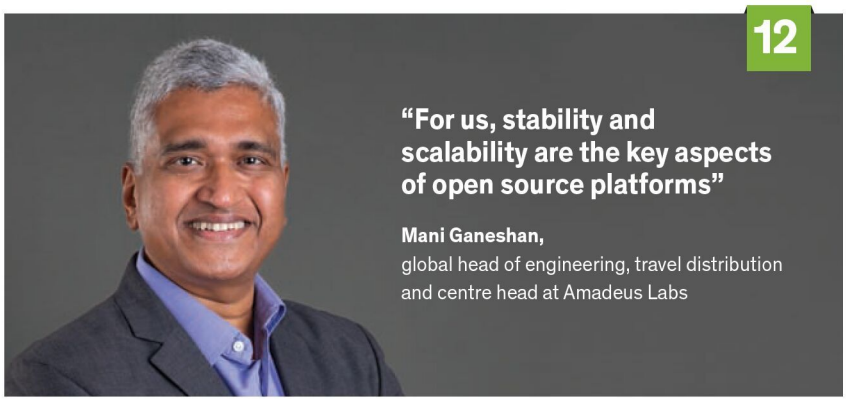
72 R Series: 'dplyr' Package

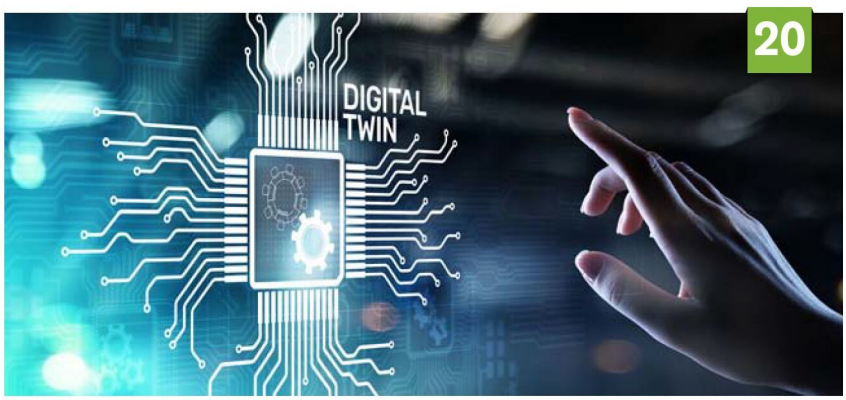
ADMIN

79 Ansible AWX: The GUI Configuration Management Automation Tool You Will Love to Use

88 Using ffmpeg for Intruder

90 The Role of Network Function Virtualization in Telecom Infrastructure





What Digital Twins Bring to the Metaverse

A Quick Look at Free Platforms and Libraries for Quantum Machine Learning

REGULAR FEATURES

07 FossBytes

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

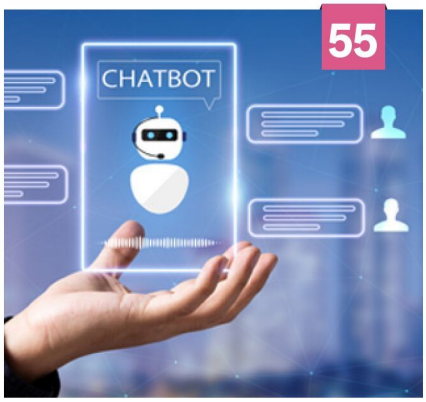
Printed, published and owned by Ramesh Chopra. Printed at Tara Art Printers Pvt Ltd, A-46,47, Sec-5, Noida, on 28th of the previous month, and published from D-87/1, Okhla Industrial Area, Phase I, New Delhi 110020.

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.

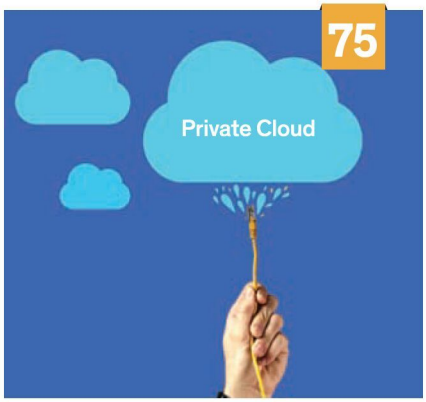
CONTENTS

How Amazon's Services can Ensure You Don't Miss Your Flight



Making a Difference with Generative AI

A Deep Dive into Generative AI and ChatGPT-3



Building a Private Cloud Using OpenStack

Building Amazon Machine Images with HashiCorp Packer

Mattermost enhances open source collaboration with generative AI

Mattermost, a provider of collaboration platforms, has developed its open source namesake technology to help security-conscious organisations better integrate generative artificial intelligence (AI). Organisations can communicate and manage workflows thanks to Mattermost's open source collaboration solution.

The market for collaboration solutions, like almost all others in IT, is now embracing the capabilities of generative AI, and Mattermost is no exception. Better integration with OpenAI will be made possible by Mattermost's new generative AI expansion, enabling businesses to create chatbots and learn about workflows.

There is more than one way AI is impacting the Mattermost platform for enterprise and government customers. The first is generative AI, where Mattermost has developed a platform that enables businesses to integrate various providers, such as OpenAI's ChatGPT and private cloud large language models (LLMs). The second key AI subcategory that Mattermost currently integrates is functional intelligence. Ian Tien, Mattermost CEO and co-founder, cited Mattermost's integration with ServiceNow and its virtual agent as a shining example of this, as it helps improve employee and customer service. Domain specific AI, where AI is used within a particular vertical domain or organisation for privacy and security compliance, is the third category for Mattermost's integrations. The integration with Ask Sage serves this purpose. Ask Sage gives organisations quick data analysis and summarisation tools so they may glean crucial insights as circumstances change.

How the new AI integrations will aid collaboration and workflow will reflect in something as basic as software installation, said Tien. In order to optimally setup and deploy a piece of software, for instance, a system administrator can now employ AI to receive answers and insights. In order to improve the overall user experience, the information the AI generates can come from both public and private sources and be transmitted securely.

Airbyte Open Source gets premium support

Airbyte has announced the launch of its first premium support package for Airbyte Open Source. More than 3,000 businesses currently use Airbyte to fuel their data pipelines, prompting the change in response to the growing need for better, more specialised assistance. With the addition of this premium support, Airbyte offers consumers who need specialised assistance from the company's team of professionals an improved user experience.

Up to this point, Airbyte supported its users via the Slack and Discourse community platforms. These discussion boards have been crucial in creating a thriving user base that shares information, insights, and solutions. However, as the user base has expanded, it has become apparent that a more direct and customised support mechanism is required.

Users can still interact with one another and offer assistance on the community Slack and Discourse platforms. These platforms will now be primarily focused on peer-to-peer interactions and community-driven self-help, while the premium support will be available to users who need help from Airbyte directly.

The new premium support service offers faster response times and personalised assistance. Today, engineers spend time going through past answers and docs. The premium support experience gives them that time back.

Airbyte helps businesses give their consumers access to the appropriate data for analysis and decision-making by making data movement simple and economical across practically any source and destination.

Open source version of FeatureByte SDK now available

The AI business FeatureByte, founded by data professionals, has announced the availability of its open source FeatureByte SDK. With just a few lines of code, the SDK enables data scientists to utilise Python to build cutting-edge features and quickly deploy feature pipelines. In order to perform feature transformations at scale in cloud data platforms like Databricks and Snowflake, FeatureByte automatically creates complex and time-aware SQL.

Organisations can get a number of advantages from the FeatureByte SDK including:

- Accelerated AI innovation: Data scientists can focus on creative problem solving and iterating rapidly on live data, rather than worrying about the 'plumbing'.
- Better business decisions: The FeatureByte SDK delivers better AI data that yields higher performing models, resulting in better business decisions for an organisation.
- Higher productivity with reduced costs: The self-service data environment delivers up to 10x compute efficiency for training, and requires 1/5th of the resources to deploy feature pipelines.

Using tools like Jupyter Notebooks, FeatureByte enables data scientists to quickly convert original concepts into training data for machine learning models that are more accurate.

"At FeatureByte, our goal is to radically simplify feature engineering and management to help enterprises truly scale AI across their organisations," said Razi Raziuddin, CEO and co-founder of FeatureByte. "FeatureByte offers a self-service environment for data scientists so they have a consistent, scalable way to prepare, serve and manage data across the entire life cycle of a model."

Hugging Face introduces HuggingChat, an open source ChatGPT alternative

Hugging Face has launched HuggingChat, a brand new chatbot powered by AI. HuggingChat is capable of carrying out a variety of jobs, including writing articles, resolving code issues, or responding to enquiries, making it an open source competitor to the popular ChatGPT. According to Hugging Face, HuggingChat, which includes 30 billion parameters, is currently the best open source chat model.

HuggingChat is based on the most recent LLaMa model created by the project OpenAssistant. The latter aims to make an AI-based assistant that is efficient and compact enough to run on consumer hardware. The LAION-5B data set, on which Stable Diffusion is based, is one among the open data sets, tools, and models made available by LAION (Large-scale Artificial Intelligence Open Network), a non-profit organisation that manages OpenAssistant.

HuggingChat has a rigorous privacy approach, in which messages are only saved for user viewing and are never shared for training or research. Also, cookies are not used for user authentication or identification. However, this may change in the future to allow users to communicate with researchers.

HuggingChat is the first open source chat project with AI. The source code for the UI is available on GitHub.

PingCAP releases its GitHub Data Explorer tool

PingCAP, a provider of cutting-edge distributed SQL databases, has announced the launch of its GitHub Data Explorer application. This ground-breaking application has been created to aid open source contributors and developers gain better understanding of their GitHub activity, thus streamlining processes and boosting productivity.

Users of PingCAP's GitHub Data Explorer receive a comprehensive dashboard with real-time access to their GitHub operations. With the use of customisable dashboards, developers can monitor project metrics, quickly spot patterns, and gain a deeper knowledge of their open source contributions.

Modern technologies were used in the development of the GitHub Data Explorer, including GH Archive and GitHub event API, TiDB Cloud, SQL generator: Chat2Query, and AI engine: OpenAI.

"PingCAP's ongoing commitment to improving data management and accessibility for everyone has resulted in the development of our GitHub Data Explorer. Today's

organisations depend heavily on data and PingCAP continuously strives to make it more manageable and accessible. By providing our users with this product, we are empowering them to easily gain access and understand their GitHub data and make decisions that will advance their businesses," said Max Liu, CEO of PingCAP.

NVIDIA sets up 'Guardrails' for AI systems

There is always the danger of AI being used to spread incorrect information, leading to security risks. NVIDIA, a proponent of AI, has just unveiled NeMo Guardrails, open source software that addresses some of the most pressing challenges of AI systems.

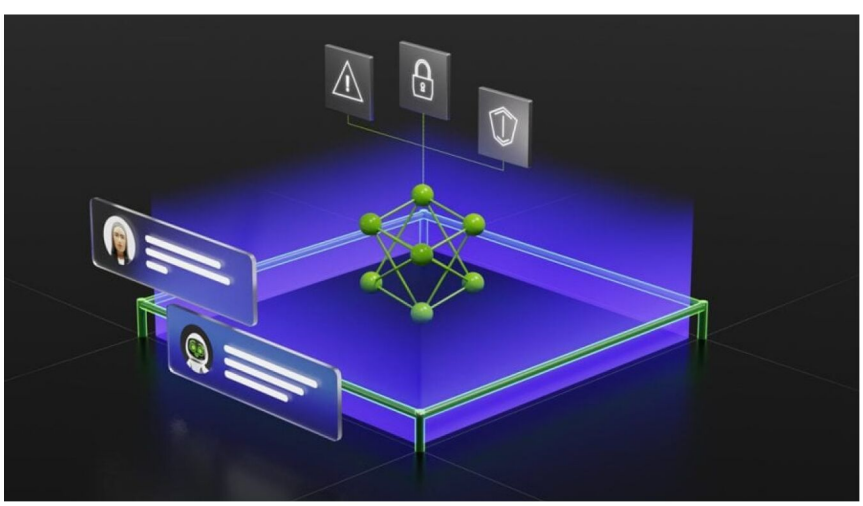
The software helps developers set up three kinds of boundaries: topical, security, and safety.

According to the company, topical guardrails prevent apps from moving into undesired areas. As an example, they keep customer service assistants from answering questions about the weather.

Safety guardrails ensure apps respond with the correct and appropriate information. They can filter out unwanted language and make certain that references are made only to credible sources.

Security guardrails ensure apps make connections only with external third-party applications that are known to be safe.

Any software developer can use NeMo Guardrails. No machine learning experts or data scientists are needed. Developers can create new rules quickly with a few lines of code.



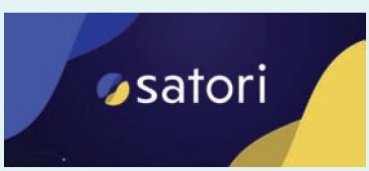
Satori releases data permissions scanner

Satori, a leading data security platform, has unveiled the open source Universal Data Permissions Scanner. This tool can search databases, data lakes, data warehouses, and cloud accounts while analysing permission models to produce a list of people and the level of access they have to files, database tables, and cloud storage buckets. Although other data stores can be added, the free tool now supports Snowflake, Databricks, Amazon S3, Amazon Redshift, Google BigQuery, and MongoDB.

The scanner makes it simpler for businesses to see and manage data store permissions. In addition to the open source version of Universal Data Permissions Scanner, which provides a command-line interface, Satori is offering a fully managed SaaS solution that runs quarterly scans.

"DevOps and data engineers are often tasked with managing the security of the databases, data lakes or warehouses they operate. This usually involves setting permissions to enable users to query the data they need. However, as the number of users and use cases increases, complexity explodes. It's no longer humanly possible to remember who had access to what, how and why, which makes meeting security and compliance requirements impossible," said a company release.

"The root cause of this problem is that permissions to data are usually stored in normalized form, which is great for evaluating permissions but not so great when you want to clearly understand your permissions landscape," it added.



KSOC releases the first-ever KBOM standard

Kubernetes Security Operations Center (KSOC) has released the first-ever Kubernetes Bill of Materials (KBOM) standard. This KBOM, which is available in an open source CLI tool, helps cloud security teams comprehend the extent of third-party tooling in their environment so they can react more quickly to newly discovered vulnerabilities. Despite the substantial third-party tool ecosystem for Kubernetes, compliance rules for the software supply chain have mostly been disregarded.

Numerous Kubernetes tools, including Crossplane, the Jenkins plugin, CubeFS, and Clusternet, now have new vulnerabilities. Although the Software Bill of Materials (SBOM) is now mandatory for federal purchases in the US, this requirement falls short of the deployment stage in the application development life cycle, where Kubernetes comes into play.

A standard for the overall scope and configuration of a cluster is becoming necessary as teams continue their widespread use of Kubernetes. This uniform standard can make understaffed companies more efficient, especially when Kubernetes expertise is already in short supply.

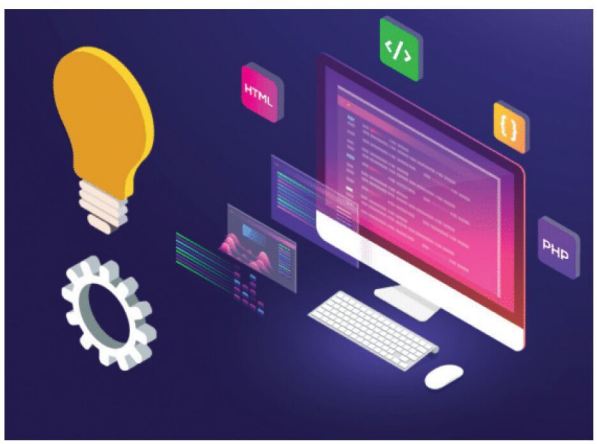
The new KBOM standard provides a quick view of the scope of your Kubernetes cluster, including workload count, cost and type of hosting service, vulnerabilities for both internal and hosted images, third-party customisation, and more.

"Kubernetes is orchestrating the applications of many of the biggest business brands we know and love. Adoption is no longer an excuse, and yet from a security perspective, we continually leave Kubernetes itself out of the conversation when it comes to standards and compliance guidelines, focusing only on activity before application deployment," said KSOC CTO Jimmy Mesta.

CISA takes over the UK government's 'Logging Made Easy' tool

The United Kingdom's National Cyber Security Centre (NCSC-UK) has discontinued funding for an open source program called 'Logging Made Easy' it developed to make it simpler to record security incidents on Windows operating systems. The United States' Cybersecurity and Infrastructure Security Agency (CISA), however, stepped in with a promise to maintain and update the tool not long after the UK declared it would stop providing support for the software.

The open source LME tool was created by NCSC-UK in the UK and released on GitHub in 2018 as a free tool. In order to "focus on the most significant cyber security challenges" and "divert resources to new initiatives designed to help protect the UK's cyber infrastructure," the UK cyber agency announced in January that it would stop supporting the tool as of March 31, this year.



According to a statement by CISA: "Neither agency will maintain code between now and when CISA reconstitutes the tool on their GitHub page. Current users who continue to use LME during this intersessional period must maintain and update the tool independently, and do so at their own risk."

"Our Logging Made Easy project has undeniably delivered results and we are proud to have supported thousands of defenders to keep their networks safe," said Lindy Cameron, CEO, NCSC-UK. "The project's transition to oversight from CISA will mean that existing and new users of the tool will continue to reap the significant benefits that it provides."

Though a specific date has not been determined, the tool will most likely be republished on CISA's GitHub page by the end of the summer.

Cisco launches open source security tools at KubeCon EU

At KubeCon+CloudNativeCon in Amsterdam this April, Cisco Systems, a longtime leader in the open source software community with more than 200 projects to its name, unveiled a collection of fresh open source tools for programmers concerned with application modernisation.

According to the firm, the three new additions—VMClarity, Nasp, and Media Streaming Mesh—have been made to enhance security for Kubernetes and cloud native systems in general. At the conference, Cisco's Emerging Technologies and Incubation division showcased how these tools increase the capabilities of cloud native settings by enhancing security tools, risk inventory in the application stack, and application modernisation.

The purpose of VMClarity is to address any issues that could arise while using virtual machines (VMs) in cloud native environments. "VMs are the No. 1 most-used service on public clouds and the predominant method for hosting containers," Vijoy Pandey, Cisco's senior vice-president of engineering, wrote in a blog post. "The resulting attack paths can be more elaborate than Amsterdam's canal system. We saw a need to provide protection for VMs against security threats such as leaked secrets, malware, and rootkit as well as system misconfigurations and vulnerability scanning, as they are still very much part of how businesses run in the cloud."

Nasp is a lightweight library to expand service mesh capabilities to non-cloud environments. It provides these capabilities to non-cloud endpoints and smaller cloud environments.

Cisco's Media Service Mesh (MSM) runs real-time media applications in cloud native Kubernetes environments. It will be available in a GitHub repository in the upcoming months.

Amazon, Meta and others make important announcements at Open Source Summit

At the Open Source Summit held by the Linux Foundation in Vancouver, Canada, this May, a number of tech giants, including AWS and Meta, made quite a few important announcements.

To begin with, AWS announced that the Cedar policy language and SDK were now open sourced. Cedar helps set permissions in applications using simple policies. It supports both role-based and attribute-based access controls. The SDKs for the language, which include libraries for developing and analysing policies, have also been made available.

AWS also announced a new open source fuzzing framework called Snapchange, which has been created in Rust. It allows programmers to create fuzzers that repeat snapshots of the physical memory in a KVM virtual machine.

Software Package Data Exchange (SPDX) release candidate 3.0 was also made available at the summit. Currently hosted by the Linux Foundation, SPDX is an open source standard that communicates the information in a bill of materials. The inclusion of six new, distinctive profiles in RC 3.0 will help it meet the demands of the market more effectively.

At the event, OpenSSF announced that it has got US$ 2.5 million from Google and US$ 2.5 million from Microsoft through its Alpha-Omega Project. It also said that Hitachi, Lockheed Martin, Salesforce, and SAP have become general members of OpenSSF. Omkhar Arasaratnam and Brian Behlendorf were both named as the foundation's new general manager and chief technology officer, respectively.

Also, Meta announced that it had joined the OpenJS Foundation as a gold member. "The broader JavaScript ecosystem benefits from Meta becoming an OpenJS Foundation member. In fact, we've already been working together in multiple different ways, and this makes official what has already been a great relationship," said Shayne Boyer, board director, OpenJS Foundation.

AMD's AGESA firmware to be replaced by openSIL

At a regional OCP (Open Compute Project) summit hosted in Prague, AMD shared its plans to replace its AMD Generic Encapsulated Software Architecture (AGESA) firmware with an open source Silicon Initialization Library (openSIL). For contemporary computer systems, firmware is an essential component, and for contemporary AMD systems, that crucial code blob is AGESA. The platform's CPU cores, chipset, and memory are among the subsystems that AGESA is in charge of initialising. It is frequently updated to support new hardware and fix faults.

But despite all the benefits that firmware offers, a system's vulnerability to cyber attacks can make it a weak point. In order to increase security, AMD has suggested making the design, architecture, and validation of the Silicon Initialization Firmware open source as part of its new firmware program. OpenSIL is intended to be light, transparent, simple, safe, and scalable. The fourth-generation AMD EPYC (Genoa) processors and associated platforms are currently compatible with openSIL, which AMD is testing in the Proof-of-Concept (POC) stage. The fifth-generation EPYC (Turin) CPUs will also be used in this stage. By 2026, AMD plans to phase out AGESA and make openSIL the standard option for the sixth-generation EPYC series.

AMD acknowledges that openSIL is still a work-in-progress but that it is already quite competitive with AGESA. It might not be available until Zen 6 or even Zen 7 because openSIL won't be ready until 2026, and AMD's most current roadmap lists Zen 5 for 2024. On the client side, AMD has not yet provided a roadmap; however AGESA will eventually be replaced by openSIL on all AMD devices.

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

"For us, stability and scalability are the key aspects of open source platforms"

'Open source is at the heart of what we do.' — this is what Amadeus, a global technology provider for the travel industry, says on its website. **Rahul Chopra** and **Vaishali Yadav** of **EFY Group** recently interacted with **Mani Ganeshan, global head of engineering, travel distribution and centre head at Amadeus Labs**, to understand why open source is important for the company, and what has been done to move in this direction. Ganeshan also talked about the hiring trends and how emerging technologies have led to interesting use cases in the travel industry. Read on to know more...

Mani Ganeshan,
global head of engineering, travel distribution and centre head at Amadeus Labs

Q. Could you briefly give some background about Amadeus and its presence in India?

A. Amadeus is the world's largest technology provider for the travel industry. We're a 37-year-old company, headquartered in Madrid and have more than 19,000 employees working in various offices worldwide. We've developed the technology on which the travel industry transacts its business, be it the sale of tickets for airlines, room nights, car rentals, and so on. We have a vast network of travel agencies on our platform.

Further, we facilitate the inventory being sold to travellers like you and me. And this is one of our largest verticals. We have a second vertical made up of IT business for airlines, hotel chains, car rental companies, and so on. We also play in adjacent businesses like payments for the travel industry, and the IT for airports. So, Amadeus has the solutions for almost everything related to the travel life cycle and its several stages.

The 11-year-old Bengaluru centre is our second largest engineering and R&D centre worldwide. Here, we have over 3000 engineers innovating on several aspects of our platform. And so, we cater to solutions not only for the global travel ecosystem but also act as the engineering hub for the APAC region, based in India.

Q. I want to understand how, despite several startups coming in, Amadeus has been able to survive and thrive in the terminals of almost all travel agents.

A. We are a SaaS platform provider, and our primary and sole customer base is our network of businesses present in the travel industry. So we are a B2B player, or rather, a B2B2C player where we make sure that using our technology and platform, travel providers like the airlines, the hotel chains, the car rental companies, and the travel agencies make it a better experience for travellers. I would say that we provide the backbone. We are not a B2C player like the new dot coms coming in. In the background, the Amadeus platform is being used for every second traveller's journey.

Q. Does that mean that you partner with them rather than them entering into your space?

A. Absolutely, we facilitate online travel agencies, like MakeMyTrip, Expedia, etc, to have more innovative solutions, and thereby cater to the more personalised needs of travellers. While the transaction happens on the online travel agency's portal, in the background, it's Amadeus.

Q. Have none of them ever attempted to come to the B2B side also?

A. For sure, the industry is evolving into a more open platform. It is happening because all the players in the travel space understand that there is a lot of friction here. We understand that the industry works on a complex network. You could be sitting in Delhi booking a ticket between Dallas and Johannesburg, using a Bengaluru-based online travel agency, a US-based payment provider, and two different airlines. And this is a common scenario in the travel industry!

Each player is now trying to find out how to stand apart from the others. So, the question is, how do you make the service a lot more personalised? How does an airline or a travel agency profile its customers, i.e., travellers, better? That's why the industry is evolving into a more open space.

And there are more and more players coming in including the larger hyper scalars and the e-commerce players, because travel is such a wide industry and a basic need. More players coming in necessitates an open platform. And that's the responsibility we carry as a large technology player in the industry—to open up the platform so that more players can come in to create innovative solutions for travellers, and reduce any friction as they flow through.

Q. Am I correct to understand that the more the B2C players come into the industry, the better it is for Amadeus? Because they are your customers in a way and, through them, you are able to reach out to even more customers.

A. You're absolutely right. The more players that are innovating in the travel industry, the better the travellers' experience gets. We have several niche players who solve a point problem for the travel industry. For example, there are startups that use IoT and focus on tracking the baggage in an airport and keeping a traveller informed. As we have our own solutions for ground handlers, we ask, can we partner with one of these startups and make a better solution for travellers? In the end, we bring these point solutions to life and make the overall journey smoother on the tech front.

Q. How would you define the word 'open platform'? Is it different from 'open source'? Please shed some light on that.

A. What we have in Amadeus is a two-sided model. Let me give an example of air travel. What are the airlines looking for? All of them have an inventory of airline seats to sell, and they want to reach out to as many travellers in the world as possible. They do that through a network of travel sellers, who are a part of brick-and-mortar or online travel agencies.

When I say the more open the platform, the better—it's for all the players. It's a two-sided model. Let's say there are more travel agencies on the platform. It obviously increases the reach of travel providers and vice versa. The travel sellers are also

running their own businesses and want to cater to their own set of clientele. The more the airlines, hotel chains and car rental companies are on the platform, the more they can create a comprehensive package for travellers, making travel more interconnected and seamless. That's what I mean when I emphasise the need for an open industry. Every player sees a benefit in being a participant in an open platform, which goes way beyond even open source.

Q. How and at what level is Amadeus using open source?

A. We leverage open source technologies quite significantly because again, in this entire complex and interconnected network, speed, agility, and innovation are the key essence to success for any of the players, including Amadeus, who provides the backbone. Now, if you look at the above said parameters, we are much better off using software that has been developed in an open community rather than trying to have our own proprietary system. That's why we are a big believer in open source and leverage several open source technologies on our platform.

And the good thing is that, now, a lot of these open source platforms are getting more sophisticated and wider in their reach. So, we benefit from that like any other player, and then we leverage these open source software in a responsible way to ensure security and privacy. That's how we play in the open source world.

Q. Where all are you using open source? Are there any broad packages or platforms that Amadeus is a big user of?

A. We have been running our own private cloud in our data centre ever since the company was created. And we are a pioneer in running a SaaS-based platform to scale because we started doing this 37 years back, and we now run this entire platform in a virtualised setup.

So, we use Red Hat, OpenShift and OpenStack extensively. We are one of the premier partners for Red Hat, and we contribute a lot to the evolution of the Red Hat, OpenShift, Kubernetes, and OpenStack platforms. Then, we provide a lot of frameworks on the e-commerce side, which are wide-level products on which travel agencies and airline websites function. For that, we use many open source UI frameworks like Angular and React.

We even created our own proprietary framework, way back in 2009, and then made it open source. It was called ARIA templates (Amadeus Rich Internet Applications). Now, it's actively getting enhanced by the wider developer community. So we have open source frameworks.

We use a lot of Apache Kafka for streaming because it's a volume-intensive platform that we deal with. We use Hadoop Spark, etc, on the data platform that we have created, for gaining insights into the travel industry. So, I would say that open source goes across many spectrums of the travel industry as well as software engineering.

Q. How does your team evaluate which ones to bet upon? Do you carry out proof-of-concepts (PoCs) or some studies? Or is the decision just based on Gartner quadrants?

A. It's a mix of all of that. Of course, we constantly scout to see how the open source community is evolving. Now, more sophisticated open source platforms are coming in. We do our own PoCs once we identify and narrow some of these. And, of course, we participate in industry studies also.

But the key and fundamental aspect we look for is the platform's stability and readiness to scale. Being a large platform with high availability and intensity of transactions, it's extremely important for our software to stand the test of volume, resilience and robustness.

So, we do experiment a lot and our global developer community is working to bring more and more open source into the Amadeus platform. We are extremely careful to make sure that it scales successfully because once we expose that and the entire industry—some 1000 airlines, 300,000 travel agencies, 100,000 hotel properties among others—starts using it, it better behave.

Q. Is there a centre of excellence or some team that looks after this whole domain of open source at Amadeus?

A. We have what we call a DevRel engineering team within the organisation. This is a transversal team that looks at making life better for our engineers and making them more productive. So when any of our engineers find some of these open source libraries or platforms that are available, they eventually funnel them into this DevRel community. It's made up of architects and principal engineers, and we operate it in an inner source kind of model. Once the software is identified, it goes to this community for evaluation, thus enabling the developer network to do PoCs and then come back to the community to report its findings.

The architecture community takes the final call to release it for wider use in two ways — one is just for our engineering community, and the second is for wider production use. The testing rigour is a lot more for the latter, because it's developed for general availability. We need our IP and legal teams to come in, and make sure that we are not violating any patents, etc. If it's for production use, then we bring other departments of the company into the assessment process.

Q. So, essentially, the trigger is from your own developer community, which identifies these platforms, right?

A. If there are startups that are interested in participating in the Amadeus community and exposing their engineering or functional capabilities to the wider travel industry, then we do have forums for that too. We have a platform called Amadeus Next, which constantly looks at onboarding more startups to use the Amadeus API and Amadeus Sandbox, so that they can play around and create more capabilities for the travel industry. Once it gets more sophisticated, we have our Ventures team driven by the Madrid headquarters, which looks at the network of startups to invest in.

Q. Is there an annual cohort or any other system?

A. Typically, it's a system that is up and running and does not operate in a cohort mode. Startups can make a pitch and we evaluate if they should be connected to our customer base—the airlines, the hotel chains and so on. If there is an appetite, we onboard them and make the connections for the startups to use the Amadeus platform, the APIs, and the sandbox to play in and connect to our customer base.

Q. Can you share any success story of a startup, Indian or global, which has gone through this process and then partnered with you?

A. The partnership between Amadeus and Airportr is a great example of how we support startups in their journey to bring innovative solutions to the travel industry. By partnering with Airportr, a London-based startup that offers luggage pickup and delivery services for air travellers, Amadeus was able to integrate their technology into Amadeus' booking and check-in system, making it possible for passengers to book Airportr's services at the same time they book their flight.

Through this partnership, Airportr got access to our extensive network of travel industry experts, which helped it to refine its product and expand its reach. Also, our expertise helped it to seamlessly integrate its solution into the booking and check-in process, making it easier for travellers to take advantage of Airportr's services.

Q. Any major platforms other than Kubernetes and the ARIA framework in which Amadeus is an aggressive contributor or driver of the forum, or is leading the communities?

A. We have partnered with Red Hat for OpenStack and OpenShift. Recently, we have also partnered with the Green Software Foundation, which is a foundation created by companies like Accenture, GitHub, Microsoft and ThoughtWorks. The spirit of this foundation is to make sure that all the software is developed in a sustainable way, leaving a minimal carbon footprint in the technology industry. Amadeus is an active participant along with these founding members.

Q. Is IoT also an area of interest for Amadeus and is any work happening there?

A. There are some use cases of IoT, like that of baggage, or where we try to map an airport so that there can be a seamless guiding of passengers to gates. There are solutions that we are working on and experimenting with, like using your mobile phone as a key for the hotel. NFC communication and some level of IoT come into play there too.

Q. And what is your take on the metaverse?

A. A lot of experimentation! It's a cool toy that many companies are experimenting with. We, in the travel industry, are experimenting to see how the metaverse can probably give a sneak peek of the travel experience before the actual travel. Though we are experimenting, large industry-scale use is yet to happen.

The metaverse is at a stage where we need to wait and watch how it gains traction beyond office productivity into production-grade use. There are some areas where it naturally fits in, like the training of pilots. This is an intense and time-consuming process, and we expect the metaverse to come in and accelerate it. But, ultimately, time will tell.

Q. How about artificial intelligence? Are there any use cases being developed? Or any interesting projects that you can share with us?

A. Everyone wants to know about ChatGPT these days. So yes, we do leverage AI technologies quite intensely within the Amadeus platform. We have some use cases, like how to search the various options that you have to reach from source to destination. If you put London and New York as your source and destination, there could be many options for a traveller. Now, the aim is to use AI and ML techniques to profile the traveller better so that you provide extremely pointed, let's say, 10 options with a high likelihood that the traveller will pick one of them, rather than providing 200 options where the traveller gets lost and doesn't book at all.

We also leverage AI/ML techniques to learn from past travel patterns, demographics, seasonality patterns, and so on to provide more custom solutions. As one example, we are leveraging AI to see how we can troubleshoot the production platform better so that even before a customer reports a problem, we can sense it ourselves and take corrective action.

We are leveraging AI solutions on our customer helpdesk also, to see how quickly we can scan our database of tickets when a customer faces a challenge, so that even before an engineer is asked to get involved, we

have provided a near-final solution to one of our customer agents.

Q. Any plans to set up something like the government of India's DigiLocker?

A. There are many innovations going on, some of which are now production grade. We are investing heavily in using biometrics—be it your fingerprint or facial recognition—as your identity so as to enable travelling through an airport without having to show a single piece of paper including your passport. Our solutions are in the pilot stage at the Delhi, Hyderabad and Goa international airports.

There are innovations parallel to the DigiLocker that are happening. During the pandemic, almost all travellers were concerned if it was safe to travel through airports, what were the regulations prevalent in destination cities/areas, and how to find a healthcare network in case some testing was needed. So, Amadeus created a safe travel ecosystem for a traveller, a kind of a document repository system, creating a network of healthcare providers, government regulations and travel players, so that the traveller felt confident to come out and start travelling. It has seen significant uptake in the travel industry over the last two years, and multiple airlines are getting millions of documents processed on this platform.

Q. That brings me to the pandemic. How did you handle that period? And what was your message to your team during those tough days?

A. There is no beating around the bush there. It was a very, very tough time for the entire industry when, in March 2020, world travel almost came to a standstill. Because we run such a large travel platform for the industry, we have seen disruptions in different pockets or some regions. The SARS virus, bird flu, etc, had disrupted travelling in some areas of the world in the past, but world travel still used to happen.

This was perhaps the first case where the entire travel ecosystem came to a standstill worldwide. It did impact our entire customer base. It impacted Amadeus significantly, with a sharp dip in our revenues worldwide, maybe the sharpest ever.

But there was a strong resolve in our executives and in the leadership that if this industry has to recover, it is the technology that is going to make a difference by creating a safe ecosystem that connects the players a lot better and thereby, brings the confidence back into the industry. And sure enough, that's exactly what happened. We knew that it was the technology that would keep the industry going. And that's one way we found to motivate our engineers to innovate. We still believe that the need for technology is almost the highest ever in the industry now.

I'm sure you're seeing that now travel has recovered almost to pre-pandemic levels, or even crossed that level in some areas. We are still continuing that impetus of innovation in the industry.

Q. What is your message to engineering students or even to those who are working professionally, about building a skillset in the open source space?

A. I would say this is an absolutely great time to be in the technology industry. Because 20 years back, open source was not as sophisticated and proprietary software was ruling the roost. I think we are now reaching a level of maturity in this industry where there is a balance between proprietary packaged software and open source. I believe the use of open source is growing at a double-digit CAGR of around 15-16% year-on-year.

This is the time to hone our skills and make sure that we stay relevant by participating in the open source wave that the world is in. And again, companies like Amadeus, which run large technology centres, are responsible and take accountability to contribute back to this community that we are benefiting a lot from. So, we motivate our engineers to inner source that software so that other engineers within the company benefit from the technology that they developed, and collectively expose that to the travel industry and thereby contribute back.

Q. How do you recognise contributions by Amadeus engineers giving back to the community? Is there any kind of motivation, reward or recognition system?

A. We have a well-recognised rewards and recognition framework within the company. We have a patent wall in our offices where you will see all the IPs that have been generated by the engineers in our organisation. We showcase and promote that. And there is also financial remuneration that we provide to our engineers when they file a patent, or when they log a disclosure and knowhow. We have a central team that helps them create the paperwork for the patent, and so on. The open source contributions that our engineers make are also a part of that.

Q. Is there a data centre in India also? How are your data centres deployed across the globe?

A. Our primary data centre is in Germany, in the south of Munich, but now, we have declared that we are on a public cloud transformation within the company. We have partnered with Microsoft and at this point, the transformation is on. We are looking to increase the payload that we are reaching into Microsoft Azure, and now that the partnership is set and is in motion, eventually, the reach of the platform will be worldwide.

And while the initial data centres we are setting up are going to be in

Europe on Microsoft, eventually, we will see this reaching worldwide including India.

Q. Is there any law in India that requires you to gradually shift the data here? Is there any timeline, etc, related to it?

A. Data sovereignty, of course, is a big challenge in the world. More and more governments are expecting data to get processed within their geographical boundaries. This was one of the key objectives we had when we partnered with Microsoft.

While there is no specific rule as such, the Indian government of course is doing its own due diligence around data privacy and data protection acts, etc. We believe that our partnership with Microsoft will automatically help us take care of the challenges.

Q. Can you give us a broad outline as to what are the different roles at the Bengaluru centre?

A. We set up the Bengaluru centre in 2012. And every year, we have been scaling the centre, getting deeper and deeper into the complex domain of travel. Our engineers are working on critical programs within the company. And so, primarily, it's a technology centre. Out of the 3000-odd engineers, around 65% are actually writing code, testing software, requirements, specifications, and so on.

Then we have a large vertical to manage and successfully run the platform after the software is ready for deployment. The platform that we run is SaaS-based and has many players and a worldwide customer base. It deals with extremely high volumes, even to the tune of 30,000-40,000 transactions a second. So, its software needs to function in high availability mode, even to a tune of 99.9999% availability uptime. So we have a large team here, that works with our other engineering centre to make sure that the platform availability is at the highest level.

We do have other smaller teams that are into providing IT helpdesk and support services for our customers. The rest is customer support and corporate functions. Of course, we have a large corporate team. We have a security operation centre in Bengaluru, which manages to keep our platform secure from day-to-day threats. So, we have a security operations centre (SOC) here that protects the Amadeus platform worldwide.

Q. Are you hiring as of now? What is the current status of the resignation wave?

A. We do acknowledge that, in the technology industry, there is some turbulence going on. Different companies are looking at their investment strategies, etc. We are in an industry that is in recovery mode. And as I mentioned earlier, the need for innovation in this industry is at its highest. So, Amadeus continues to invest in the technology platform to transform the industry. We recently expanded to a new office in Pune, to tap into the rich talent that the market offers. Currently, we are looking to fill more than 100 roles by next year at the centre.

Q. Does it help the job applicants at Amadeus to have an open source contribution? How do the engineering leaders make sure that the keywords on their resumes have some real weight?

A. Oh, absolutely. When we are searching for engineers and talents, I'd say not only open source but we see if they have that technological, geeky frame of mind. Other things come out during the discussions and interviews. We have partners who constantly scout open source platforms to see the contributions. There are sophisticated hiring portals that have already done this kind of profiling when they source a candidate for us.

Q. Any technologies that you would suggest to professionals, especially developers, to hone their skills on, and some which are now of the past and they may forget about?

A. I never hazard the risk to say any technology is in the past. Because history tells us that any technology that gets developed, eventually lasts a long time. For example, you would think that mainframes are legacy as they were created 40-50 years back, and now, hardly anyone uses mainframes. But, if you do a survey in the industry, I'm sure there are many sectors that are still on the mainframes. The travel industry is on mainframes in many aspects. Amadeus is one of the first technology providers in the travel industry that got off mainframes and ran fully on open source mainframe systems. But there are many travel providers who are still running mainframes in their data centres. So, I never hazard the risk to say that any software has become redundant. For example, C++ still remains very popular and we leverage it vigorously.

But if you look to the future, Python is getting popular; streaming software like Kafka and sophisticated UI frameworks like React—these are software that we leverage heavily on our platform.

Q. What message would you convey to the open source community at large?

A. Again, we would be happy to welcome more and more engineers into the open source bandwagon and make sure that the solutions that we develop are sophisticated, supportable, secure, and take care of various security considerations across the world. So, the more the merrier. Working in the open source community provides unique opportunities to make a large impact in the technology ecosystem and I would recommend young engineers to pursue it. END

What Digital Twins Bring to the Metaverse

The metaverse, where real life converges with the digital world, has given birth to many new age business models and use cases. Digital twins within the metaverse offer new ways of interacting with the real world. Let's see how...

One of the key technologies driving the evolution of the metaverse is that of digital twins. Most popular in the retail industry, a digital twin is a virtual replica of a physical object or system, offering a comprehensive digital representation that can be used for various purposes. When combined with the metaverse, digital twins can enable new ways of experiencing and interacting with the real world.

Architecture of metaverse-based digital twins

The architecture of a metaverse-based digital twin consists of several layers. At the bottom layer is the physical asset or system that the digital twin is replicating. This could be anything from a building, to a manufacturing plant, to a vehicle. The physical asset is equipped with sensors and other IoT devices that collect data on its operations, condition, and performance. This data is then transmitted to the digital twin, which is hosted in the cloud or on-premises.

The next layer is the data processing layer, which consists of several components that work together to process the data collected from the physical asset. This layer typically includes data ingestion and storage, data processing and analysis, and data visualisation and reporting. Data ingestion and storage involves collecting data from various sources, and storing it in a scalable and secure data store. Data processing and analysis involves applying advanced analytics techniques, such as machine learning and artificial intelligence, to the collected data to extract insights and identify patterns. Data visualisation and reporting involves presenting the results of the analysis in an easy-to-understand format.

The third layer is the digital twin layer, which is responsible for creating a virtual representation of the physical asset or system. The digital twin layer consists of several components, including the virtualization engine, the digital twin model, and the digital twin APIs. The virtualization engine is responsible for creating and managing the digital twin, while the digital twin model defines the relationships between the physical asset and its virtual counterpart. The digital twin APIs provide an interface for interacting with the digital twin, enabling users to view and manipulate the virtual representation of the physical asset.

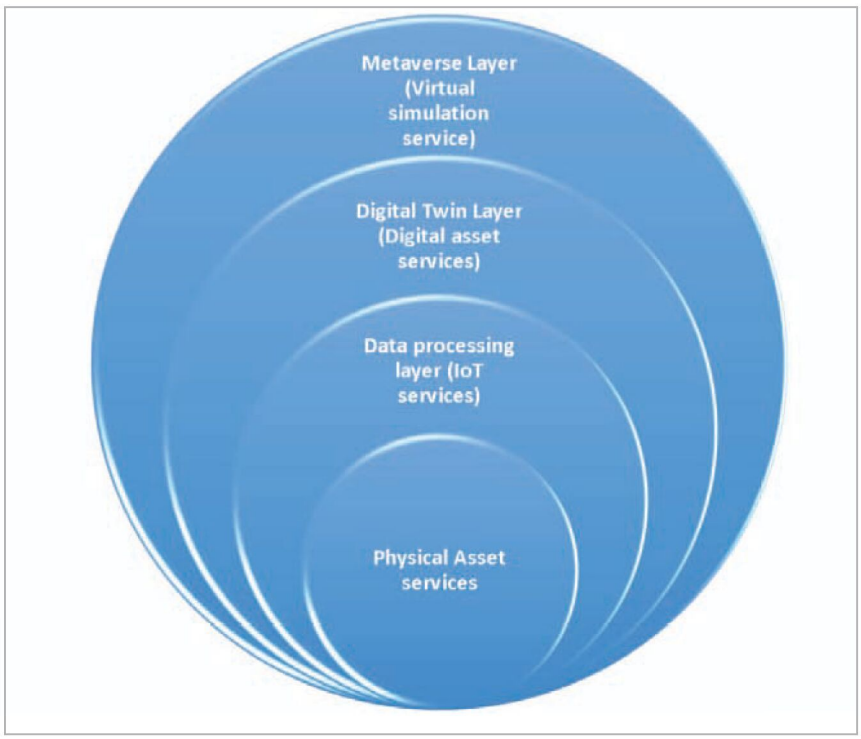


Figure 1: Architecture of digital twin-based metaverse platform

The fourth layer is the metaverse layer, which is where the digital twin is brought to life. The metaverse layer consists of several components, including the immersive environment, the social interaction engine, and the real-time simulation engine. The immersive environment is the virtual space where the digital twin is presented, while the social interaction engine enables users to interact with each other in the virtual world. The real-time simulation engine is responsible for simulating the behaviour of the physical asset in real-time, based on the data collected from the IoT devices.

Components of metaverse-based digital twins

The components of a metaverse-based digital twin can be grouped into several categories. These categories include the physical asset, the data collection and processing components, the digital twin model, the virtualization engine, and the metaverse components.

The physical asset component consists of the physical object or system that the digital twin is replicating. This component includes all the sensors and IoT devices that collect data on the physical asset's operations, condition, and performance.

The data collection and processing components are responsible for collecting, storing, and analysing the data collected from the physical asset. These components typically include data ingestion and storage, data processing and analysis, and data visualisation and reporting.

The digital twin model component defines the relationships between the physical asset and its virtual counterpart. This component includes the digital twin model, which is a comprehensive representation of the physical asset, and the digital twin APIs, which provide an interface for interacting with the digital twin.

The integration of digital twins with metaverse solutions has proven to be advantageous in several ways. Digital twins offer a virtual replica of physical entities, and metaverse solutions provide a platform to integrate them into a virtual environment. This combination enables businesses to harness the benefits of both technologies to enhance their operations and offerings.

One significant advantage of digital twins with metaverse solutions is improved efficiency. By creating a virtual replica of physical entities, businesses can simulate operations and analyse data to identify bottlenecks, inefficiencies, and areas for improvement. This process can lead to better resource allocation, reduced downtime, and optimised operations. For example, in the manufacturing industry, digital twins with metaverse solutions can be used to simulate production processes and identify areas for optimisation, leading to increased production rates and reduced waste.

Another advantage of this solution is improved customer experience. By creating virtual environments that allow customers to interact with digital twins, businesses can offer a more immersive and engaging experience. For instance, in the retail industry, customers can virtually try on clothes or see how furniture would look in their homes before making a purchase.

Digital twins with metaverse solutions also offer improved remote collaboration. By creating virtual workspaces, businesses can allow employees to collaborate on projects, regardless of their physical location. This approach can reduce the need for travel and increase productivity.

Furthermore, digital twins with metaverse solutions offer improved safety and risk management. By simulating scenarios and predicting potential risks, businesses can identify and mitigate safety hazards before they occur. For example, in the oil and gas industry, digital twins with metaverse solutions can be used to simulate drilling operations and identify potential risks.

The integration of digital twins with metaverse solutions offers several advantages, including improved efficiency, enhanced customer experience, improved remote collaboration, and improved safety and risk management. As these technologies continue to evolve, the potential for innovative applications and new advantages will continue to emerge.

Industrial applications of metaverse-based digital twins

Metaverse technology with digital twins is rapidly evolving and transforming various industrial sectors by enhancing customer experiences, optimising business operations, and improving overall efficiency. This technology has the potential to revolutionise how industries operate.

Retail industry: In the retail sector, digital twins within the metaverse help create an immersive shopping experience. Virtual stores set up with digital twins technology enable customers to experience products in a 3D environment. Customers can view products, interact with them, and get an accurate representation of how these look and feel. This technology can also help retailers in optimising inventory management and supply chain management.

Healthcare industry: Metaverse with digital twins can enable healthcare professionals to monitor patients remotely and provide more personalised treatments. The technology can help doctors simulate patient conditions, analyse treatment options, and improve patient outcomes. It can also aid in creating realistic surgical simulations that can help train surgeons.

Manufacturing industry: In the manufacturing sector, digital twins within the metaverse help create a digital thread for products. This allows manufacturers to track products throughout their life cycle from design to production, delivery, and even end-of-life disposal. Digital twins technology can also aid in optimising production processes by simulating different scenarios and analysing the results to identify the most efficient process.

Telecom industry: In the telecom sector, metaverse and digital twins enable the creation of a virtual network. Digital twins technology can help in creating a virtual representation of the entire network, including all devices and connections. This virtual network can help telecom companies to improve network performance and identify potential network issues.

Automotive industry: In the automotive sector, metaverse and digital twins can enable a more personalised driving experience. Digital twins technology can help to create a virtual representation of the car, including all its components, systems, and software. This representation can help in optimising the car's performance, identifying potential issues, and providing a more personalised driving experience.

Common challenges of developing digital twins with the metaverse

Developing digital twins with the metaverse can present various challenges due to the complexity and emerging nature of these technologies. Here are some common problems that developers may encounter.

Data integration and interoperability: Integrating real-time data from physical objects or systems into digital twins within the metaverse can be challenging. Ensuring compatibility and interoperability between different data formats, protocols, and devices can pose technical hurdles. Developing robust data pipelines and standardisation methods is crucial to ensure seamless data integration.

Scalability and performance: Building digital twins that can handle a large number of concurrent users and complex interactions within the metaverse can strain system resources. Scaling the infrastructure to support a growing user base and maintaining optimal performance, such as low latency and high responsiveness, can be demanding tasks.

Security and privacy: Digital twins often involve sensitive data, such as information about physical assets or personal user data. Ensuring the security and privacy of this data throughout the digital twin-metaverse ecosystem is critical. Implementing robust security measures including authentication, encryption, and access controls is essential to protect against unauthorised access, data breaches, and privacy violations.

Realism and immersion: Achieving a high level of realism and immersion within the metaverse can be a significant challenge. Creating realistic and detailed virtual environments that accurately represent the physical counterparts of digital twins requires advanced rendering techniques, physics simulations, and detailed asset modelling. Striving for a seamless integration of the physical and virtual worlds is essential for an immersive experience.

User experience and interaction: Designing intuitive and engaging user experiences within the metaverse is crucial for user adoption. Ensuring smooth and natural interactions between users and their digital twins, as well as with other users, requires thoughtful interface design, haptic feedback, natural language processing, and gesture recognition. Balancing usability, functionality, and immersion is a key consideration.

Standardisation and interoperability: With multiple platforms and technologies emerging in the space, lack of standardisation and interoperability can be a challenge. Ensuring compatibility and seamless integration between different digital twin platforms, metaverse environments, and devices is essential for a cohesive and connected ecosystem. The development of industry-wide standards and protocols can help address this challenge.

Cost and infrastructure requirements: Developing and maintaining digital twins with the metaverse can involve significant costs. Creating high-fidelity virtual environments, managing real-time data streams, and ensuring robust infrastructure can require substantial investments. Managing the cost-effectiveness of development and deployment, while providing a quality user experience, can be a delicate balance.

Ethical considerations: As digital twins become more prevalent and interconnected within the metaverse, ethical considerations arise. Issues such as data ownership, data privacy, algorithmic bias, and potential misuse of digital twins need to be addressed. Establishing ethical guidelines and frameworks to govern the development and use of digital twins within the metaverse is important.

Addressing these challenges requires collaboration between various stakeholders, including developers, researchers, industry experts, and policymakers. As technology advances and the field matures, solutions and best practices will continue to evolve to overcome these obstacles.

Popular platforms and solutions for digital twins within the metaverse

Unity3D: This is a widely used game development platform that has expanded its capabilities to support the creation of virtual and augmented reality experiences. It provides tools and features for building immersive environments and integrating digital twins with the metaverse.

Unreal Engine: Developed by Epic Games, this is another popular game engine that offers robust capabilities for creating interactive and realistic virtual worlds. It can be utilised to develop metaverse experiences and integrate digital twins into these environments.

NVIDIA Omniverse: NVIDIA Omniverse is a platform that aims to connect virtual worlds, simulations, and collaborative environments. It provides a framework for creating digital twins, and enables real-time rendering and interaction within the metaverse.

Microsoft Azure Digital Twins: This cloud-based platform is designed for building comprehensive digital representations of physical environments. It allows users to model, simulate, and analyse real-world systems, and can be integrated with metaverse applications to create immersive experiences.

Siemens MindSphere: Siemens MindSphere is an industrial IoT platform that supports the development of digital twins. It enables the integration of real-time data from physical assets with virtual representations, and can be combined with metaverse technologies to create immersive industrial simulations and analytics.

Autodesk Forge: This cloud-based development platform provides tools for creating and managing digital twins. It offers capabilities for visualising and interacting with digital representations of physical assets, which can be leveraged in the context of the metaverse.

IBM Maximo: IBM Maximo is an enterprise asset management platform that includes digital twin functionality. It enables the creation and management of digital twins for assets and infrastructure, and can be integrated with metaverse solutions to provide a holistic view of assets and environments.

PTC ThingWorx: This industrial IoT platform supports the development of digital twins and the integration of real-time data from sensors and devices. It can be used to create digital twins that can interact with the metaverse and provide real-time insights.

What the future holds

Digital twins and the metaverse are two emerging technologies that are expected to shape the future of our digital world. When combined, they can bring about significant advancements and transformative experiences. Here are some potential future trends.

Enhanced virtual collaboration: Digital twins in the metaverse can enable immersive and collaborative experiences. Users will be able to interact with each other and their digital representations in a shared virtual environment. This can revolutionise remote collaboration, allowing teams from around the world to work together seamlessly.

Real-time data integration: Digital twins can collect real-time data from physical objects or systems and integrate it into the metaverse. This will provide a dynamic and up-to-date representation of the physical world, enabling real-time monitoring, analysis, and simulations. For example, a digital twin of a smart city could incorporate live data on traffic, energy usage, and weather conditions.

Personalised virtual spaces: With the metaverse, individuals can have their own customisable virtual spaces. These spaces can be connected to their digital twins, allowing them to personalise their virtual environment based on their preferences, needs, and interests. Users can also interact with their digital twins within these spaces, creating a personalised and immersive experience.

IoT integration: Internet of Things (IoT) devices can be connected to digital twins within the metaverse, enabling enhanced functionality and control. For instance, a smart home digital twin could integrate with IoT devices such as thermostats, cameras, and appliances, allowing users to remotely monitor and control their physical environment from the metaverse.

AI-driven simulations and predictions: The combination of digital twins and the metaverse can facilitate advanced simulations and predictions powered by artificial intelligence. By leveraging historical data and machine learning algorithms, digital twins can simulate scenarios, predict outcomes,

and optimise performance in various domains such as manufacturing, healthcare, and urban planning.

Virtual commerce and experiences: The metaverse can serve as a platform for virtual commerce, where users can engage in virtual shopping experiences, try out products before purchasing, and interact with virtual representations of real-world stores. Digital twins can enhance these experiences by providing accurate and realistic representations of products and environments.

Cross-platform interoperability: As the metaverse evolves, interoperability between different platforms and systems will become crucial. Digital twins can act as a bridge between various virtual worlds and environments, enabling seamless transfer of data and experiences across different platforms and applications.

Digital twin economies: The metaverse can give rise to new economic models centred around digital twins. Users could create, own, and trade digital twins representing physical assets, virtual goods, or even intellectual property. This could lead to the emergence of digital twin marketplaces and economies, where individuals and businesses can monetise their digital assets.

These trends demonstrate the potential of combining digital twins with the metaverse to create immersive, interconnected, and intelligent digital experiences. However, it's important to note that these trends are speculative, and the actual implementation and adoption of these technologies may vary as they continue to develop.

In conclusion, the metaverse with digital twins has the potential to revolutionise different industrial sectors but we need to keep in mind certain roadblocks like infrastructure cost, security challenges, compliance issues, and so on. END

References

- *https://www.mckinsey.com/capabilities/mckinsey-digital/our-insights/digital-twins-the-foundation-of-the-enterprise-metaverse*
- *https://www.forbes.com/sites/forbestechcouncil/2022/06/01/what-digital-twins-and-the-metaverse-mean-for-our-infrastructure/?sh=4d4b6248683a*
- *https://www.forbes.com/sites/forbestechcouncil/2022/05/03/using-digital-twins-and-preparing-for-the-metaverse/?sh=177b690929e2*
- *https://www.pymnts.com/metaverse/2022/linking-digital-twins-to-make-an-industrial-metaverse/*

By: Dr Magesh Kasthuri and Dr Anand Nayyar

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.

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.

OSFY Magazine Attractions During 2023-24

Month	Theme
March 2023	Security, Network Management and Monitoring
April 2023	Open Source Programming (Languages and tools)
May 2023	Cloud Special: Everything from management to implementation
June 2023	AI, Deep learning and Machine Learning
July 2023	Database management and Optimisation
August 2023	Mobile/Web App Development, Optimisation and Security
September 2023	DevOps Special
October 2023	Blockchain and Open Source
November 2023	Open Source and IoT and Edge
December 2023	All About Data Management
January 2024	Containers and Managing Containers
February 2024	Open Source on Windows and Best in the world of Open Source (Tools and Services)

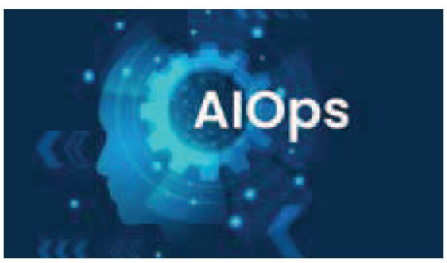
FOCUS

Generative AI

A Quick Look at Free Platforms and Libraries for Quantum Machine Learning

Quantum computing, due to its ability to calculate at an immense speed, has the potential to solve many problems that classical computers find difficult to address. Quantum machine learning or QML is a new field that explores the intersection between quantum computing and machine learning. Several libraries and platforms facilitate the development of QML algorithms and applications. A few popular ones are discussed in this article.. *Read more on page.....30*

Use AIOps to Make Your Enterprise Agile



Artificial intelligence for IT operations or AIOps is a boon for the modern day enterprise. Read on to find out its benefits, and get to know the popular AIOps platforms.

Read more on page.....47

Making a Difference with Generative AI

Everyone seems to be using ChatGPT for something or the other. Here, our experts suggest some serious, game-changing applications that the technology can be put to—and the pitfalls to avoid.

Read more on page.....55

AI: Beyond Python

In this final article of the 12-part series on AI, we will briefly review the topics covered thus far and suggest how to continue improving our skills. Additionally, we will discuss the main topics that were not adequately covered in the series and offer guidance on how to master them. However, the primary focus of this article will be on programming languages, other than Python, that are widely used for developing machine learning-based applications.

Read more on page.....38

How Amazon's Services can Ensure You Don't Miss Your Flight

Large airports and long queues can sometimes put even the most disciplined passenger at the risk of missing a flight. Amazon has a range of AI and MLbased services that can be used at airports to ensure that passengers who reach the airport on time never miss a flight.

Read more on page.....50

A Deep Dive into Generative AI and ChatGPT-3

ChatGPT has taken the world by storm. But there is a lot more happening in the world of AI. We take a look at that, and also peek into the strengths and a few shortcomings of ChatGPT-3.

Read more on page.....62

ACCELERATING DEVELOPMENT AND DEPLOYMENT OF OPEN SOURCE

20th edition

Power in Your Hands: Running Local Large Language Models for AI Brilliance

This step-by-step guide provides a good understanding of the Dalai Library and how to get started with the LLaMA and Alpaca language models on your own machine. Once you have done that, the applications are limited only by your imagination.

Figure 1: Image generated by Midjourney and further edited in Canva

Have you ever thought about running cutting-edge language models like ChatGPT on your own computer? You're not alone! In this adventure, we'll explore how to make it happen with the LLaMA and Alpaca models, using the Dalai Library. Get ready for a detailed, step-by-step, and entertaining journey into the world of local AI! At the end of it you will, hopefully, get a good understanding of how generative AI can reside in your machine.

The local AI dilemma: Our own private sanctuary

Let's face it, we're all excited about the possibilities generative AI models bring. But with great power comes great responsibility (thanks, Uncle Ben). One of the biggest concerns we have is data privacy. Centralised models like OpenAI and Microsoft's offerings are fantastic, but do we really want to hand over our data on a silver platter?

Imagine if Batman had to share his Batcave location with everyone. Not cool, right? That's where running an AI model on your local machine comes into play. It's like having your very own Batcave (minus the cool gadgets and bat-themed vehicles, of course).

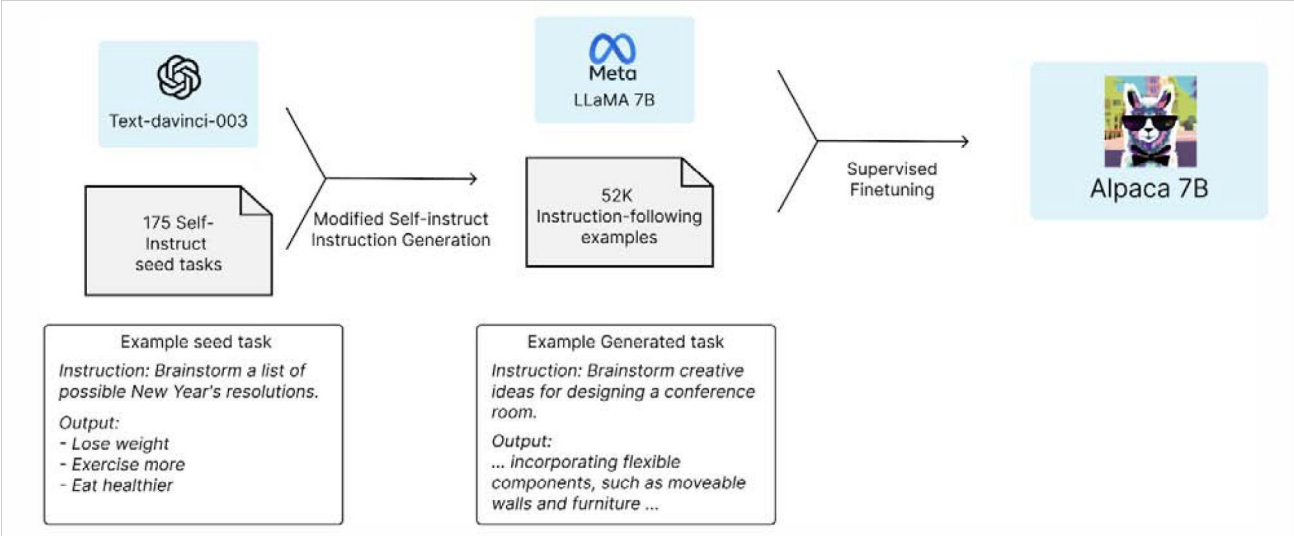


Figure 2: Paper published in the Center for Research on Foundation Models (CRFM) at Stanford

LLaMA: The compact powerhouse created by Meta AI

LLaMA is a foundational language model that has managed to achieve something incredible. Despite being 13x smaller than the colossal GPT-3, it outperforms the latter on most benchmarks! This compact powerhouse is capable of running on local machines – one daring individual even managed to get it working on a Raspberry!

Now, thanks to some unforeseen circumstances, LLaMA is available for non-commercial use. Developed by the talented team at Meta AI, LLaMA and its sibling Alpaca are making local AI usage more accessible than ever before. So, without further ado, let's get started on this great ride!

Installing and running LLaMA

Clone the repo and install the necessary prerequisites from *https://github.com/cocktailpeanut/dalai*.

To kick things off, run the command:

```
npx dalai llama install 7B
```

Before you proceed, though, be aware that LLaMA-7B needs around 31GB of storage. So make sure there's enough space on your computer for this small yet mighty guest. I struggled with this a lot!!

To run LLaMA, simply type:

```
npx dalai serves
```

..and you've done it! You now have a large language model running locally. Give yourself a pat on the back! Why? Because, I had to go through a lot of trouble to get this running like specific versions of Python and Node. You can find the details in the Readme file at *https://github.com/cocktailpeanut/dalai*.

However, when I ran this, there were a lot of meaningless machine letters that came in answer to a simple question "I feel like having some snacks because . ." Upon further research, it looks like this is a well-known issue. Someone in a discussion channel suggested: "Check Alpaca model," so I did that.

Alpaca: The instruction-following marvel

Alpaca is a fine-tuned version of LLaMA that's been designed to follow instructions, much like ChatGPT. The jaw-dropping part is that the entire fine-tuning process cost less than US$ 600! When you compare that to GPT-3's staggering US$ 5,000,000 price tag, it's an absolute steal!

So, how did they do it? Well, OpenAI's text-davinci-003 model lent a helping hand unwittingly by transforming 175 self-instruction tasks into a whopping 52,000 instruction-following examples for supervised fine-tuning. Talk about a clever workaround! The brains behind this amazing model are Rohan Taori, et al. It's so creative — they basically used da-vinci-03 as a teacher for LLaMA to produce AIpaca!! You can find the entire paper on this at *https://crfm.stanford.edu/2023/03/13/alpaca.html*.

Installing and running Alpaca

To install Alpaca, all you have to do is run:

```
npx dalai alpaca install 7B
```

Continued to page...34

A Quick Look at Free Platforms and Libraries for Quantum Machine Learning

Quantum computing, due to its ability to calculate at an immense speed, has the potential to solve many problems that classical computers find difficult to address. Quantum machine learning or QML is a new field that explores the intersection between quantum computing and machine learning. Several libraries and platforms facilitate the development of QML algorithms and applications. A few popular ones are discussed in this article.

Quantum computing uses quantum mechanics to perform calculations. While classical computers use bits, which can represent either 0 or 1, quantum computers use qubits, which can exist in multiple states simultaneously. This allows quantum computers to perform certain types of calculations much faster than classical computers, especially those related to optimisation, machine learning, and cryptography.

However, building quantum computers is a significant technical challenge, as qubits are highly sensitive to environmental noise and require sophisticated error-correction techniques. Despite these challenges, there is significant interest in the potential applications of quantum computing in areas such as drug discovery, materials science, and artificial intelligence.

Quantum computing has the potential to revolutionise many fields by solving problems that are currently intractable using classical computing. There is a huge scope for quantum computing in assorted domains, and it has the potential to impact nearly every field of science

and technology. However, the technology is still in its early stages of development, and there are significant challenges to be addressed in terms of hardware limitations, error correction, and algorithm development, as well as scalability and reliability. Nonetheless, with continued research and development, quantum computing could lead to significant breakthroughs and advancements in many areas of science and technology.

In 2020, Google claimed to have achieved quantum supremacy with its Sycamore quantum computer, which took just 200 seconds to complete a calculation that would have taken the world's fastest supercomputer 10,000 years to complete.

Other major players in the quantum computing field include IBM, Microsoft, Intel, and Honeywell, all of whom are developing their own quantum computing technologies and making them available to researchers and developers through cloud based services.

Key advantages of quantum computing

Speed: Quantum computers can solve certain problems much faster than classical computers, especially those related to data optimisation, machine learning, and cryptography.

Parallelism: Quantum computing allows for massive parallelism, which means that many calculations can be performed at the same time.

Quantum superposition: Quantum computing can make use of quantum superposition, which allows quantum bits (qubits) to exist in multiple states simultaneously. This allows for more complex calculations and faster problem-solving.

Quantum entanglement: Quantum computing can also make use of quantum entanglement, which allows qubits to be connected in such a way that the state of one qubit affects the state of the others. This can be used to perform certain types of calculations much faster than classical computing.

Improved accuracy: Quantum computing can offer improved accuracy over classical computing in certain calculations, such as in the simulation of chemical reactions and the modelling of financial markets.

Security: Quantum computing can potentially offer improved security over classical computing in areas such as cryptography, as certain algorithms that are difficult to break using classical computing can be easily broken by quantum computing.

Innovative applications: Quantum computing is a new and rapidly evolving field, with the potential for a wide range of innovative applications in areas such as drug discovery, materials science, and artificial intelligence.

Some of the potential applications of quantum computing are listed below.

- ***Cryptography:*** Quantum computers have the potential to break many of the cryptographic algorithms that are used to secure sensitive data. However, they can also be used to develop new, more secure encryption methods.
- ***Optimisation:*** Many real-world optimisation problems, such as supply chain management and logistics, are extremely difficult to solve with classical computers. Quantum computers can provide faster and more efficient solutions to these problems.
- ***Machine learning:*** Quantum machine learning algorithms can be used to analyse and classify large amounts of data more efficiently than classical algorithms.
- ***Chemistry:*** Quantum computers can simulate the behaviour of molecules and chemical reactions more accurately than classical computers, which can lead to the development of new materials and drugs.
- ***Finance:*** Quantum computing can be used to optimise portfolios, risk assessment, and other financial calculations.
- ***Weather forecasting:*** Quantum computing can provide more accurate and precise weather forecasts by simulating complex weather patterns and climate models.
- ***Particle physics:*** Quantum computing can be used to simulate particle interactions, and accelerate the development of new theories and technologies in particle physics.

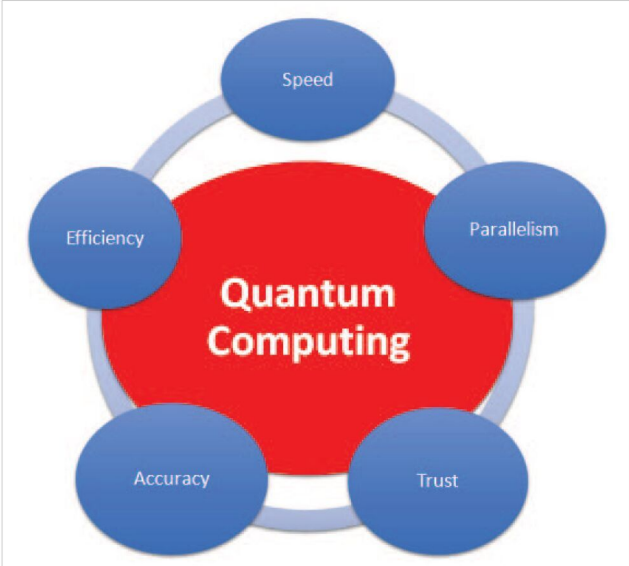


Figure 1: Advantages of quantum computing

Machine learning integration with quantum computing

Quantum machine learning (QML) is a field that explores the intersection between quantum computing and machine learning. It is focused on developing algorithms and techniques that can leverage the unique properties of quantum computing to improve the efficiency and accuracy of machine learning tasks.

Quantum computers use qubits (quantum bits) to perform operations that can solve certain problems exponentially faster than classical computers. This speed can be particularly advantageous for large scale data analysis tasks, such as those encountered in machine learning.

One of the main goals of quantum machine learning is to develop quantum algorithms that outperform classical machine learning algorithms for tasks such as classification, clustering, and regression. Proposed quantum machine learning algorithms include the quantum support vector machine (QSVM), quantum principal component analysis (QPCA), and quantum k-means. One example of a quantum machine learning algorithm is the quantum approximate optimisation algorithm (QAOA), which is used to solve optimisation problems. QAOA is a hybrid algorithm that combines classical optimisation with quantum operations to find the optimal solution to a problem.

Another example of a quantum machine learning technique is a quantum-inspired classical algorithm. These algorithms are designed to mimic the behaviour of quantum systems using classical computers, with the potential for improved performance in certain tasks.

Platforms and libraries for quantum machine learning

As already stated, QML is an interdisciplinary research area at the intersection of quantum computing and machine learning. In recent years, several libraries and platforms have emerged to facilitate the development of QML algorithms and applications. Here are some popular ones.

TensorFlow Quantum (TFQ)

https://www.tensorflow.org/quantum

TFQ is a library developed by Google that enables the creation of quantum machine learning models in TensorFlow. It provides a high-level interface for constructing quantum circuits and integrating them into classical machine learning models.

PennyLane

https://pennylane.ai/

PennyLane is an open source software library for building and training quantum machine learning models. It provides a unified interface to different quantum hardware and simulators, allowing researchers to develop and test their algorithms on a range of platforms.

Qiskit Machine Learning

https://qiskit.org/ecosystem/machine-learning/

Qiskit is an open source framework for programming quantum computers, and Qiskit Machine Learning is an extension that adds quantum machine learning algorithms to the toolkit. It provides a range of machine learning tools, including classical machine learning models that can be trained on quantum data.

Pyquil

https://pyquil-docs.rigetti.com/en/stable/

Pyquil is a library for quantum programming in Python, developed by Rigetti Computing. It provides a simple interface for constructing and simulating quantum circuits and allows for the creation of hybrid quantum-classical models

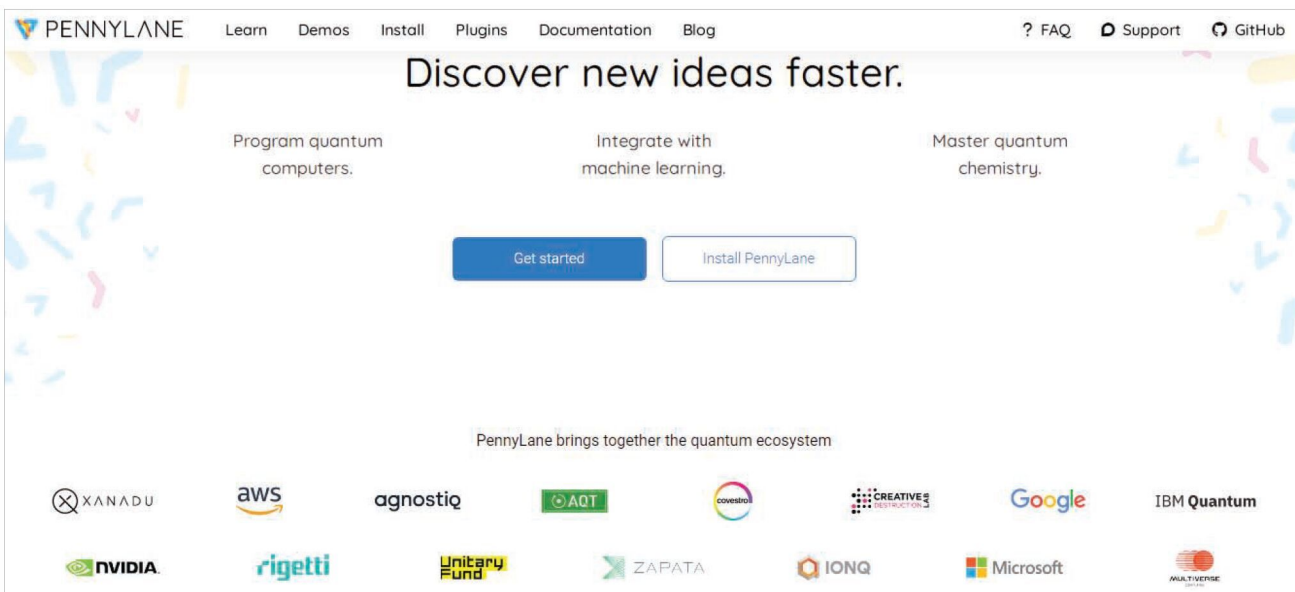


Figure 2: PennyLane framework for quantum machine learning

for machine learning. Forest is a suite of software tools for developing and running quantum applications, also developed by Rigetti Computing. It includes Pyquil and other tools for quantum programming, as well as a cloud based platform for running quantum simulations and experiments.

IBM Q Experience

https://quantum-computing.ibm.com/

IBM Q Experience is a cloud based platform for programming and running quantum circuits on IBM's quantum computers. It includes a range of tools for building and testing quantum algorithms, including quantum machine learning algorithms.

These are just some of the platforms and libraries available for quantum machine learning. As the field continues to grow, we can expect to see more tools and platforms emerge to support this exciting field of research.

Implementation scenarios

The following code demonstrates how to use a quantum circuit to classify two data points as either class 0 or class 1, based on a training data set.

```
from qiskit import QuantumCircuit, execute, Aer
from qiskit.aqua.components.feature_maps import
RawFeatureVector

# Define training data and labels
training_data = [[1, 0], [0, 1]]
training_labels = [0, 1]

# Define feature map circuit
feature_map = RawFeatureVector(feature_dimension=2, data_map_
func=lambda x: x)

# Define quantum circuit
qc = QuantumCircuit(2, 1)
qc.append(feature_map, [0, 1])
qc.h(0)
qc.cx(0, 1)

# Measure qubit 0 to obtain classification result
qc.measure(0, 0)

# Execute circuit on local simulator
backend = Aer.get_backend('qasm_simulator')
job = execute(qc, backend, shots=1024)
result = job.result()

# Print classification result
counts = result.get_counts()
print(counts)
```

In this code, we first define a training data set with two data points and their corresponding labels. We then define a feature map circuit that maps each data point to a quantum state. In this case, we use the RawFeatureVector feature map that maps each data point to a 2-qubit state.

We then define a quantum circuit with two qubits and one classical bit. We apply the feature map to the two qubits, followed by a Hadamard gate on the first qubit and a CNOT gate between the two qubits. Finally, we measure the first qubit to obtain the classification result.

We execute the circuit on a local simulator and obtain the counts of the measurement outcomes. The output will be a dictionary of measurement outcomes and their corresponding counts, such as: {'0': 483, '1': 541}.

This result indicates that the first data point was classified as belonging to class 0 with a probability of approximately 47 per cent, and belonging to class 1 with a probability of approximately 53 per cent. The actual classification depends on the threshold value used to interpret the measurement outcome.

Example of quantum machine learning for image classification

Here is an example of quantum machine learning applied to image classification using the PennyLane and TensorFlow quantum libraries:

```
import pennylane as qml
import tensorflow as tf
import tensorflow_quantum as tfq
from tensorflow.keras import layers

n_qubits = 4  # number of qubits to use in quantum
circuit
n_classes = 3  # number of classes to classify images
into

# Define a quantum circuit that will act as the classifier
dev = qml.device("default.qubit", wires=n_qubits)

@qml.qnode(dev)
def circuit(inputs, weights):
    # Encoding the input data as quantum states
    for i in range(n_qubits):
        qml.RY(inputs[i], wires=i)

    # Apply the trainable weights to the circuit
    for i in range(n_qubits):
        qml.Rot(*weights[i], wires=i)

    # Measure the qubits to get the output probabilities
    return [qml.probs(wires=i) for i in range(n_qubits)]
```

```
# Define the model using TensorFlow Quantum
inputs = tf.keras.Input(shape=(n_qubits,))
weights = tf.Variable(tf.random.uniform((n_qubits, 3)))
outputs = tfq.layers.PQC(circuit, weights)(inputs)
model = tf.keras.Model(inputs=inputs, outputs=outputs)

# Prepare the image classification data
(train_images, train_labels), (test_images, test_labels) =
tf.keras.datasets.mnist.load_data()
train_images = train_images.reshape(-1, n_qubits)
test_images = test_images.reshape(-1, n_qubits)
train_labels = tf.keras.utils.to_categorical(train_labels,
num_classes=n_classes)
test_labels = tf.keras.utils.to_categorical(test_labels,
num_classes=n_classes)

# Train the model
model.compile(optimizer=tf.keras.optimizers.Adam(learning_
rate=0.01),
              loss=tf.keras.losses.
CategoricalCrossentropy())
model.fit(train_images, train_labels, epochs=5)

# Evaluate the model on test data
test_loss, test_acc = model.evaluate(test_images, test_
labels)
print("Test accuracy:", test_acc)
```

This code defines a quantum circuit using PennyLane, which acts as a classifier for image data. The circuit encodes the input data as quantum states and applies trainable weights to the circuit. The qubits are then measured to get the output probabilities, which are used to classify the images.

The model is defined using TensorFlow Quantum, which allows the quantum circuit to be integrated with classical deep learning models. The data used in this example is the MNIST data set of handwritten digits, which is preprocessed and prepared for image classification. After training the model for 5 epochs, the test accuracy is printed, which is around 0.87 in this example. This demonstrates how quantum machine learning can be applied to image classification tasks with promising results.

Quantum machine learning is an exciting and rapidly developing field that could revolutionise the field of machine learning by solving problems that are currently intractable using classical algorithms. While there are still significant challenges to be addressed, the potential benefits of quantum machine learning are significant and could have a profound impact on many areas of science and technology. 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.

Continued from page...29

Alpaca is a lightweight champ, requiring only 4GB of storage, so it won't take up much space on your computer.

To run Alpaca, just repeat the command:

```
npx dalai serve
```

..and you've got it — your very own ChatGPT-like model, ready to serve!

The versatile Dalai API

The fun doesn't end here – the Dalai Library also offers an API that enables you to integrate both LLaMA and Alpaca into your own applications. This opens up a world of possibilities for innovative projects and experiments on your local machine.

Think about creating your own AI-powered chatbot, building a smart writing assistant, or even developing an AI tutor for your favourite subject! Now that you are not limited to 32k words, think about feeding all the classics written by your favourite author (who is no longer there to give us more magical creations), and create the books you always craved for! The only limit is your imagination, and I would love to hear about the creative ways you're using LLaMA and Alpaca in your projects.

So, go ahead and explore the potential of these powerful yet accessible AI models. Just remember, both LLaMA and Alpaca are intended for non-commercial use only. Happy experimenting, and don't forget to share your groundbreaking ideas with the community! END

References

- *https://ai.facebook.com/blog/large-language-model-llama-meta-ai/*
- *https://twitter.com/miolini/status/1634982361757790209*
- *https://crfm.stanford.edu/2023/03/13/alpaca.html*

By: Jaydeep Chakrabarty

The author is a contributor to the open source community and an application security evangelist for the last 15+ years. He works at Thoughtworks Technologies India.

An Introduction to MLOps

There is virtually no industry segment that is untouched by machine learning today. However, with the rapid growth of machine learning based projects, there is the need to build a process and system to manage them. This is where MLOps (machine learning operations) comes in.

MLOps focuses on deploying and maintaining machine learning models in production. It includes:

- Data engineering operations for gathering and preparing data
- Model development operations for multiple iterations of train, test, validate, refine and release

Simply put, MLOps brings the rigour of agile principles to machine learning projects.

Challenges of not using MLOps

Fundamentally, machine learning (ML) projects are different from typical software projects. Organisations run into a myriad of issues to effectively manage ML projects. Data scientists typically need to work on multiple iterations of machine learning models. Creating and managing models by following processes that are not optimised for ML can be a cumbersome process.

Collaboration between different groups is another big challenge. Developing a good model requires collaborative effort between data scientists, DevOps and IT (including data engineers and ML engineers) and often, these different groups have different perspectives.

Managing technical debt between various ML models is another challenge.

MLOps addresses all these challenges effectively.

Key steps followed in MLOps

- Data collection
  - Connects to the source system and collects the required data.
- Data analysis
  - Understands the data, which includes the data names, types, meaning and context of data.
- Data refinement
  - Removes bad data, fills missing data and manages duplicates. Formats the data to get consistency.
- Data preparation
  - Creates the usable data set on which the model can be applied.
- Model creation and training
  - Chooses the correct algorithm and model for the given business problem and trains it with the data.
- Model validation
  - Validates the results from the model with the expected outcome. Manages variances and minimises the error.
- Model deployment
  - Deploys the model to various environments (Dev, QA, UAT, production, etc).
- Model evaluation
  - Evaluates the model by using and applying the output of the ML model and finds recommendations for future improvement. Tunes the parameters as required.
- Model re-training
  - Modifies the model to improve its effectiveness and continue the loop of train, test and evaluate.

Benefits of using MLOps

- Improved quality
  - Because of increased monitoring and reliable operations, as well as a model-drift, fail-fast and faster retry approach, the output is of good quality.
- Increased efficiency
  - There is an increase in the productivity and efficiency of data scientists and ML/data engineers.
- Better coordination
  - Improved and streamlined processes result in better understanding and coordination across various teams.
- Better business outcomes
  - Businesses can apply the ML outcomes and reap the benefits. END

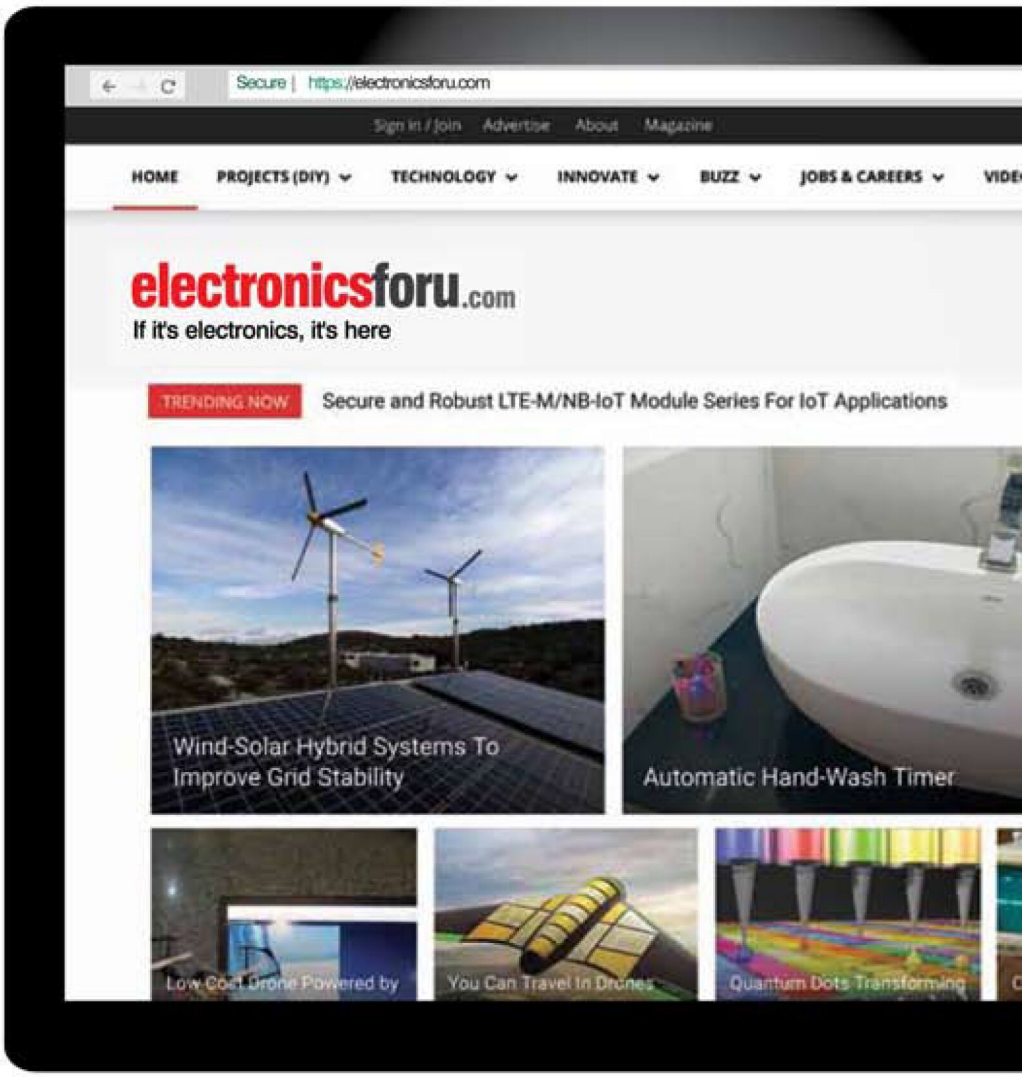
Reference

- *Ml-ops.org*

By: Phani Kiran

The author has over 19 years of experience in the IT industry, and has worked on digital transformation initiatives.

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

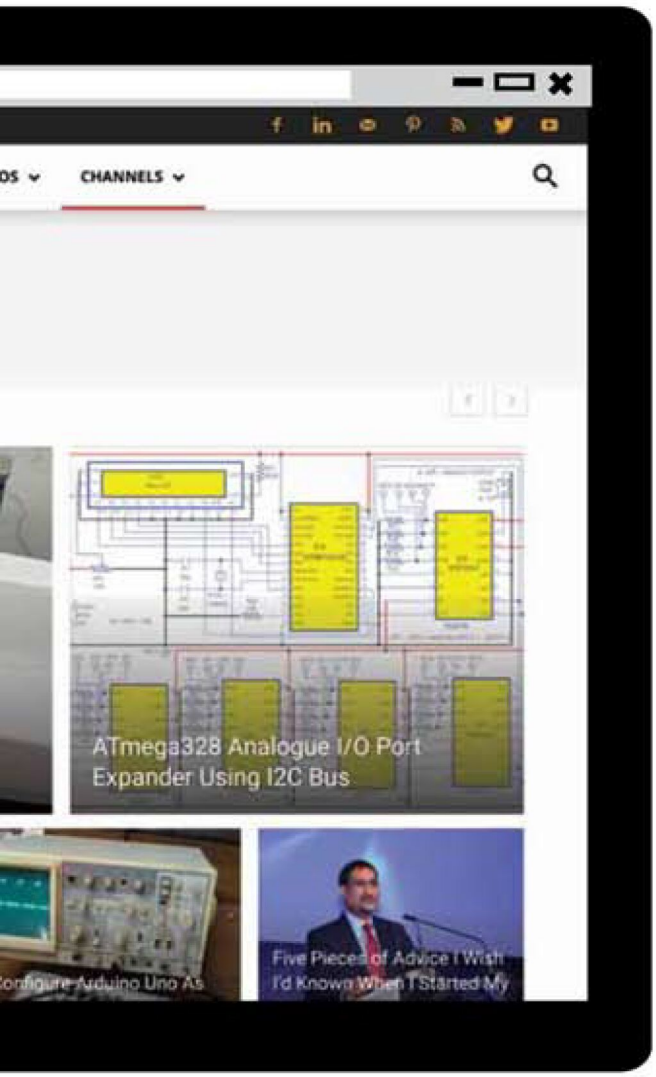
READERS

- 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...



Amazing DIY Projects. Latest Tech trends.
The hang-out for electronics enthusiasts.

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.

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

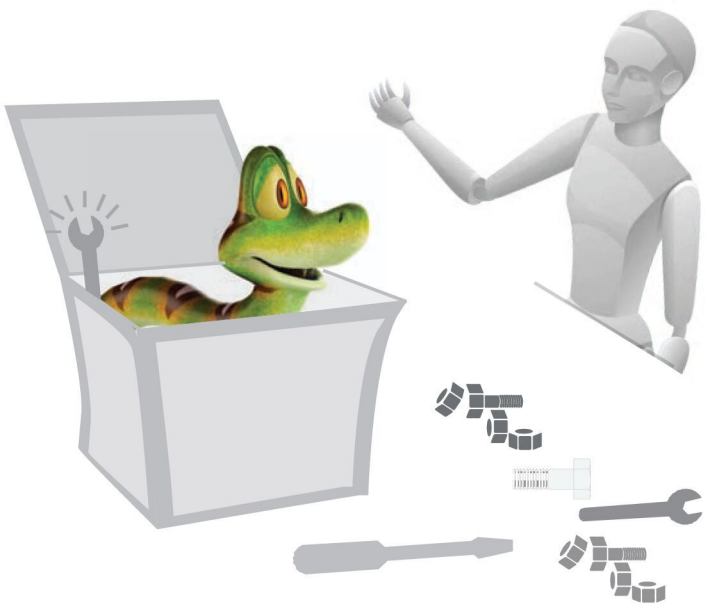
RESPONSE GUARANTEED SOLUTIONS

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

AI: Beyond Python

In this final article of the 12-part series on AI, we will briefly review the topics covered thus far and suggest how to continue improving our skills. Additionally, we will discuss the main topics that were not adequately covered in the series and offer guidance on how to master them. However, the primary focus of this article will be on programming languages, other than Python, that are widely used for developing machine learning-based applications.



As we arrive at the final article in this series on AI and machine learning, I am reminded of a quote by Winston Churchill that aptly applies to the journey of a newcomer introduced to AI through this series: “This is not the end, this is not even the beginning of the end, this is just perhaps the end of the beginning.” While we have covered a great deal in this series and gained valuable knowledge, this is only the start of our journey in the vast ocean of AI. To navigate these challenging waters, we must still acquire a multitude of skills.

Let us begin this final article by taking a moment to review the latest trends in the fields of AI and machine learning.

For a while, there were various online and offline tools that claimed to use AI to perform diverse tasks. However, not many people took these tools seriously. For example, the AI-based image generation model DALL-E 2 has been active since January 2021. However, tools of this kind gained prominence after ChatGPT was launched. Suddenly, hundreds of AI-based tools came into the spotlight. Let us take a look at some of them.

The first one is Bard, developed by Google and launched in March 2023. It is a chatbot based on the LaMDA family of large language models and aims to compete with ChatGPT. However, as of April 2023, Bard is not yet available in India. In my observation, ChatGPT provides only subpar performance while answering advanced mathematical questions — not simple arithmetic. Therefore, I am eagerly waiting to compare Bard with ChatGPT. Bard is just the beginning — almost every tech giant has announced or introduced chatbots to compete with ChatGPT. Let us discuss a few more of them. LLaMA (Large Language Model Meta AI) is a large language model developed by Meta (Facebook). Although launched in February 2023, you have to request access and wait to use LLaMA. This is another tool I am eagerly waiting to try out.

IBM Watson from IBM and Ernie Bot from Baidu are two other potential competitors to ChatGPT. Additionally, tools like Amazon

Lex can be used to develop your own chatbots. There is also an open source language model called GPT-J, which is similar to OpenAI's GPT-3. Notice that ChatGPT uses GPT-3.5 (used by the free product) and GPT-4 (used by the premium product). Microsoft's search engine called Bing also has features based on GPT-4. Other chatbots available online include Perplexity, Jasper, Chatsonic (Writesonic), Replika, Tome, etc. However, many of these services are not free like ChatGPT. Interestingly, some of these tools even provide the option to rewrite text to bypass AI content detection.

A few programming languages for AI

Now, let us talk about some programming languages that are useful for developing AI and machine learning applications. Before we dive in, we need to address two important questions. First, why are we discussing other programming languages towards the end of this series when we have spent a lot of time learning Python-based techniques? Well, we are doing so because these languages provide benefits such as faster processing, better portability, greater low-level control, and improved scalability for developing AI and machine learning applications. Additionally, we will only focus on programming languages that have characteristics distinct from Python to make this discussion worthwhile.

The second question is: What should you do when you have a strong affinity for a programming language that is not listed among the popular options for developing AI and machine learning applications? Fortunately, times have changed, and no programming language or development community can ignore the growing importance of AI. Almost every programming language has some features to support AI-based development. For instance, if you prefer JavaScript, a popular language for web development, there are now certain libraries and frameworks available to support AI and machine learning, such as TensorFlow.js, Brain.js, and ConvNetJS. These libraries enable developers to perform machine learning tasks directly in the browser, which is useful for creating interactive applications that rely on real-time predictions.

```
# sbcl
This is SBCL 2.0.1.debian, an implementation of ANSI Common Lisp.
More information about SBCL is available at <http://www.sbcl.org/>.

SBCL is free software, provided as is, with absolutely no warranty.
It is mostly in the public domain; some portions are provided under
BSD-style licenses.  See the CREDITS and COPYING files in the
distribution for more information.
* (load "add.lisp")
Program Loaded
T
* (add 111 222)
333
```

Figure 1: Execution of the Lisp program

Now, let us explore some of the key programming languages that are used for developing AI.

Prolog is a logic programming language that is well-suited for certain types of AI applications, such as expert systems and natural language processing. Released in 1971, Prolog is actively used today, though not at the same scale as many other programming languages discussed in this series. For example, one of the languages used to develop IBM Watson is Prolog. You can execute Prolog programs in your system by using a compiler called GNU Prolog. It can be installed in Ubuntu by using the command, '*sudo apt install gprolog*'. Prolog is worth considering as a potential candidate for AI development due to its unique programming paradigm. However, keep in mind that this language is used sporadically, and the chances of coming into contact with a professional Prolog programmer are slim.

Lisp is another historically (being the second oldest surviving high-level programming language) and practically relevant programming language for AI development. Although Lisp is not as widely used as some other programming languages, it continues to have a dedicated community of developers who value its unique features and capabilities. Nowadays, Lisp has several dialects available, such as Clojure, Scheme, Racket, and Common Lisp. Common Lisp is presently one of the most popular dialects of Lisp, and SBCL (Steel Bank Common Lisp) is the most commonly used implementation of it. SBCL is an open source implementation known for its quick performance and high level of conformity to the Common Lisp standard. To install SBCL on Ubuntu, run the command, '*sudo apt-get install sbcl*' in your terminal. Now, consider the Lisp program '*add.lisp*', shown below, which adds two numbers. Figure 1 shows the execution of this Lisp program. Notice that executing the instruction '*sbcl*' initiates the operation of the SBCL compiler.

```
(format t "Program Loaded")
(defun add (a b)
  (+ a b))
```

Haskell, a true functional programming language, and Scala, which provides many functional programming capabilities, are also frequently utilised for creating AI and

machine learning applications. Lisp, Haskell, and Scala are significant programming languages in the domain of AI because of the programming paradigm they support. Functional programming features are extremely beneficial when developing AI and machine learning applications. Although Python does support certain functional programming features, such as lambda functions, it is not enough, which underscores the critical importance of programming languages like Lisp, Haskell, and Scala.

Programming languages like C++ and Java are considered for developing AI when requirements like speed, low-level control, and scalability come into the picture. Since C++ and Java are quite well-known, let us move on to discussing other languages.

R is a programming language for statistical computing and is preferred over Python when strong statistical computing capabilities are required. For more details, I urge you to go through the ongoing series on R currently featured in *Open Source For You.*

In my opinion, a comprehensive discussion on the development of AI-based software should include references to proprietary programming languages and tools like MATLAB, Wolfram Mathematica, etc. However, I am aware that many advocates of free and open source software consider such tools as unacceptable. Personally, I align with the moderate faction of the free and open source software community, and do not entirely discourage the use of proprietary software. However, to avoid any controversy, I will talk about Scilab and GNU Octave, two programming languages that have the ability to mimic MATLAB effectively.

Before we proceed any further, I would like to introduce a textbook titled 'Neural Data Science: A Primer with MATLAB and Python' by Nylen and Wallisch. This textbook provides examples in both Python and MATLAB, which makes it easier to switch between the two programming languages. Furthermore, I have tested the code in this textbook using both Scilab and GNU Octave interpreters to ensure their conformity with MATLAB. These comparisons revealed that Scilab has lower syntactic compatibility with MATLAB than GNU Octave.

Now, let us discuss Scilab. It is a numerically-oriented programming language and a free and open source numerical computational package. Scilab is an excellent tool for developing AI and machine learning-based applications due to its ease of learning and rich set of libraries. To install Scilab on Ubuntu, run the command '*sudo apt-get install scilab*' in your terminal, and execute the command '*scilab*' to run the Scilab interpreter. Figure 2 shows a simple example of multiplying two matrices using Scilab. Note that matrix A is of order 1x3 (1 row and 3 columns) and matrix B is of order 3x1 (3 rows and 1 column). Therefore, the product matrix AB has an order of 1x1, with the single element being 66, while the product matrix BA has an order of 3x3. This is a valid MATLAB code as well, but even basic tasks expose the syntactic incompatibility of Scilab with MATLAB. For instance, the perfectly valid MATLAB code '*C = transpose(B)*' produces the following error in Scilab: '*Undefined variable: transpose*'. As a result, potential users of Scilab should exercise extreme caution when using MATLAB code.

```
-->A = [1 1 1]
 A  =

   1.   1.   1.

-->B = [11; 22; 33]
 B  =

   11.
   22.
   33.

-->A*B
 ans  =

   66.

-->B*A
 ans  =

   11.   11.   11.
   22.   22.   22.
   33.   33.   33.
```

Figure 2: Matrix multiplication using Scilab

```
octave:1> A = [1 2; 3 4 ; 5 6]
A =

   1   2
   3   4
   5   6

octave:2> transpose(A)
ans =

   1   3   5
   2   4   6
```

Figure 3: Transpose of a matrix using GNU Octave

Let us now discuss GNU Octave, a free and open source programming language primarily used for scientific computing and numerical computation. Its compatibility with MATLAB, as well as its comprehensive set of tools, make it a powerful programming language for developing AI and machine learning-based applications. To install GNU Octave on Ubuntu, simply run the command '*sudo snap install octave*' in your terminal, and to run the GNU Octave interpreter, execute the command '*octave*'. In Figure 3, a simple example of finding the transpose of a matrix using GNU Octave is shown. Note that matrix A has order 3x2 (3 rows and 2 columns) and therefore, the transpose of matrix A has order 2x3 (2 rows and 3 columns). It is worth noting that, unlike Scilab, this code (which is compatible with MATLAB) works perfectly fine with GNU Octave.

In fact, MATLAB-compatible code works almost flawlessly with GNU Octave, unlike Scilab. The syntactical differences between MATLAB and GNU Octave are relatively rare, but there are a few that might cause trouble. Figure 4 shows some minor syntactic differences

between MATLAB (on the left side of the figure) and GNU Octave (on the right side of the figure). The first difference shown in Figure 4 is due to the different ways in which character strings and character arrays are treated by the two languages. The second difference is due to GNU Octave supporting the post-increment operator available in programming languages like C, C++, Java, etc. From Figure 4, it is clear that MATLAB does not support this operator, whereas its behaviour is similar in C, C++, Java, and GNU Octave. Keep in mind that these are not the only syntactical differences between MATLAB and GNU Octave. As the differences can be more subtle, GNU Octave users should be careful when using MATLAB code.

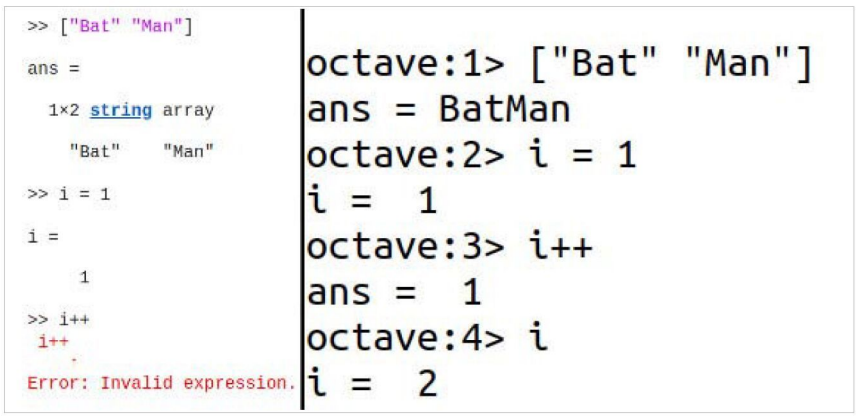


Figure 4: Syntactic differences between MATLAB and GNU Octave

Introduction to Julia

Now, it is time to introduce Julia, a programming language that might pose the greatest threat to the dominance of Python in the development of AI and machine learning-based applications. Julia is a relatively new, high-level programming language designed for computational science. It was first released in 2012, and its features make it well-suited for developing AI and machine learning-based applications. Python and Julia can be compared using the analogy of a mobile phone camera and a digital camera. Python, being a widely-used programming language, is as versatile as a mobile phone and can be utilised for various applications, including AI-based development. In contrast, Julia is like a digital camera specialised in a particular task. Julia is designed for high-performance computing and numerical analysis. It is a relatively new language that has gained prominence in the scientific computing community due to its speed and efficiency. Julia's syntax is somewhat similar to Python's, but it is optimised for speed and efficiency, making it ideal for complex numerical computations and simulations. Thus, speed is one of the important reasons why you should prefer Julia over Python.

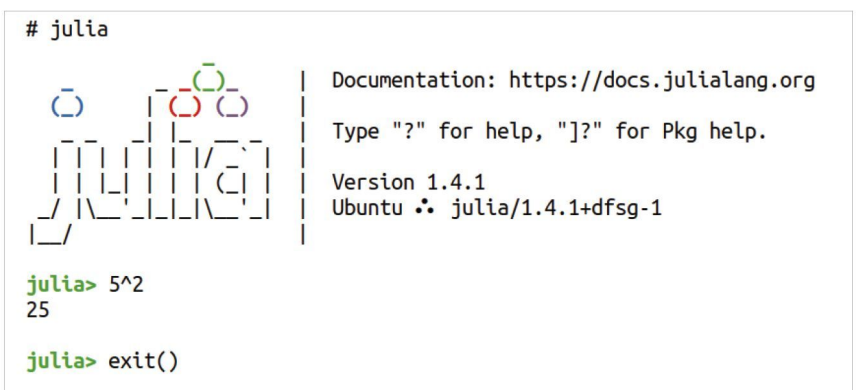


Figure 5: Julia REPL

As Julia shows great potential as a programming language for AI in the future, let us delve into it further. To install Julia on Ubuntu, simply enter the command '*sudo apt-get install julia*' in your terminal. Then use the command '*julia*' to launch the Julia REPL (Read-Eval-Print Loop), a command-line interface for working with Julia. To exit the Julia REPL, simply type '*exit( )*' and press *Enter*. In Figure 5, you can see an example of the Julia REPL running a mathematical expression (5^2 = 25).

Now, let us write and execute a Julia program. Notice that Julia programs have a '*.jl*' extension. Consider the Julia program called '*pgm1.jl*' shown below, which generates a random array of integers, and calculates its mean and standard deviation.

```
using Statistics
Arr = rand(1:100, 10)
μ = mean(Arr)
σ = std(Arr)
println("Array: $Arr")
println("Mean: $μ")
println("Standard Deviation: $σ")
```

The line of code '*using Statistics*' loads the package *Statistics*, which provides several statistical functions, including mean and standard deviation. The line of code '*Arr = rand(1:100, 10)*' creates a one-dimensional array of length 10, filled with random integers between 1 and 100 (inclusive). The rest of the code calculates the mean and standard deviation of the array *Arr*, and prints the array *Arr* and the calculated results. Julia is a programming language intended for scientific computing and allows the use of mathematical symbols as variable

names. For example, the above program uses the symbols μ and σ as variable names to store the mean and standard deviation, respectively. In the Julia REPL, you can get Unicode math symbols by typing the symbol names in LaTeX followed by typing the *tab* key. For example, the variable name μ can be generated by typing '*\mu*' (LaTeX for μ) followed by the *tab* key in the Julia REPL. These types of variable names make the code more expressive, easier to read, and closer to the mathematical notation used in formulas and equations. The Julia program can be executed with the command '*julia pgm1.jl*'. Figure 6 shows the execution and the output of the Julia program *pgm1.jl*.

By now, we are quite familiar with JupyterLab and Jupyter Notebooks. One huge benefit of Julia is that it can be used with Jupyter Notebooks, allowing users to leverage the strengths of both technologies. IJulia is a package that enables the use of Julia within a Jupyter Notebook environment, making it easy to work with data, perform computations, and visualise results. In this series, we have already installed and used JupyterLab. Now, let us explore how IJulia can be installed and used as a kernel in Jupyter Notebooks. Once you have installed Julia, open the Julia REPL and enter the following two commands on the Julia prompt to add the IJulia package: '*using Pkg*' and '*Pkg.add("IJulia")*'. The execution of these two commands will install the IJulia package on your system. Now, execute the command '*Jupyter lab*' to access Julia in a Jupyter Notebook. Figure 7 shows the option to choose Julia (marked inside a blue box) as the kernel for a Jupyter Notebook. As mentioned earlier in this series, JupyterLab can support various kernels. As you can see, my system supports not only Julia but also Python and SageMath kernels.

```
# julia pgm1.jl
Array: [71, 99, 22, 59, 28, 84, 70, 25, 71, 79]
Mean: 60.8
Standard Deviation: 26.848546412132713
```

Figure 6: Output of the Julia program *pgm1.jl*

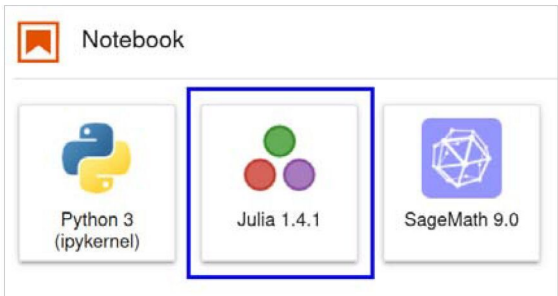


Figure 7: Julia in JupyterLab

```
start = time()
s = factorial(BigInt(200000))
finish = time()
println("Julia: ", finish - start, " Seconds")
#println(s)

Julia: 0.032067060470581055 Seconds

import time
import math
start = time.time()
s = math.factorial(200000)
finish = time.time()
print("Python: ", finish - start, "Seconds")
#print(s)

Python:  0.3572051525115967 Seconds
```

Figure 8: Comparing the execution speeds of Julia and Python

Let us wrap up our discussion of Julia by comparing the execution speeds of Python and Julia.

Figure 8 shows programs to find the factorial of the number 200000, using both Julia (shown at the top of the figure) and Python (shown at the bottom of the figure). Just in case you forgot what a factorial is, the factorial of 5 (denoted as 5!) is 120 (5x4x3x2x1). The programs are straightforward because both Julia and Python have a function called *factorial( )* to find the factorial of a number. I commented out the line of code to print the answer obtained because the final answer has nearly a million digits (973,351 to be exact), which would require 235 A4-size pages to print. Figure 8 also shows the amount of time that the Julia and Python programs took to complete the calculation. It is evident that Python took ten times longer than Julia. While there are strategies to optimise the code, this is the most apparent code that comes to the mind of most programmers, and Julia is the undisputed victor here.

In conclusion, I would like to emphasise the speed of Julia — it produced a number with almost a million digits in less than 33 milliseconds on my laptop (which has modest specifications).

Let us now pause for a moment to consider the remarkable advances in computing power and scientific progress over the last century. Although it is a bit saddening that we can only imagine the technology that will exist 100 years hence, it brings us peace and happiness to know that our children will have a brighter future due to the rapid progress in science and technology.

A quick recap

Now, let us briefly go over the topics covered in this series, along with some key topics that were either left out or given only brief coverage. In this series, I attempted to introduce maths, particularly linear algebra and probability, as needed. However, it is important to note that our discussions were not highly formal or extensive. I want to emphasise once again that understanding maths is crucial for mastering the techniques required for developing AI-based solutions,

without which you will be stuck copying code from Stack Overflow forever. To gain a thorough and formal understanding of these topics, I recommend the books 'Linear Algebra Done Right' by Sheldon Axler and 'A First Course in Probability' by Sheldon Ross. Please keep in mind that these are my personal recommendations, and there are many other textbooks available that cover these subjects in a way that may be more suitable for you.

Apart from the mathematical aspects, we have also done a lot of coding as part of the series. We did not approach the series in a programming language-agnostic way; rather, we relied mostly on Python for coding. While programming with Python, we used tools like JupyterLab and Anaconda Navigator. We covered almost all the major Python libraries and packages, such as TensorFlow, Keras, PyTorch, scikit-learn, Pandas, OpenCV, Matplotlib, etc, which are used for developing AI and machine learning-based applications. Identifying and utilising relevant Python libraries is crucial due to the vast number of options available, and I hope you will give it serious thought. In every article in this series, we also discussed the theoretical aspects of AI and machine learning. I hope that you will continue to update yourselves with the most recent developments in the AI industry.

Let us now examine the various AI and machine learning paradigms that we explored throughout this series. We provided comprehensive coverage of supervised learning and delved into the specifics of classification and regression using different Python libraries. However, to truly enhance your understanding of supervised learning, it is essential to work through more examples. While support vector machines (SVMs) are a critical component of supervised learning, our coverage of them was minimal, and we did not engage in any theoretical discussion regarding them. As a result, I highly recommend dedicating time to master both the theory and application of SVMs.

Our coverage of unsupervised learning was relatively limited compared to supervised learning. We presented an example of how clustering could be implemented, but there is still much more to learn about unsupervised learning, even with clustering. Additionally, we entirely omitted the topic of dimensionality reduction, a crucial aspect of unsupervised learning. By doing so, we disregarded techniques such as principal component analysis (PCA) and independent component analysis (ICA). Therefore, I suggest taking the time to study these techniques in-depth.

We briefly touched on the two lesser-known paradigms of machine learning, semi-supervised learning and reinforcement learning. Acquiring knowledge in these areas could prove beneficial in the long run for your journey through AI and machine learning. Neural networks are the fundamental building blocks for AI-based applications. Learning the specifics of neural networks formally, rather than informally as we treated them, could have a tremendous impact on your understanding of AI.

As we come to the end of this 12-part series on AI, I am reminded of a quote by the famous poet W.H. Auden, "A poem is never finished, only abandoned." Looking back, I feel the same way about this series. There were many things I could have done differently, such as adding or excluding topics or improving the presentation. However, I am happy with the journey we have taken, which I feel has served its purpose of introducing eager learners to the field of AI. Although I may have deviated from the original plan on occasions, I have remained faithful to the goal of introducing the maths and programming behind AI-based applications, albeit less formally than originally intended. Indeed, what sets this series apart, in my opinion, is our unique approach of blending the theoretical, mathematical, and programming aspects of AI.

During our one-year journey through AI, we witnessed a growing interest in the subject from the general public, which reached its peak with the introduction of ChatGPT. As an example of this, ChatGPT was featured on the cover page of the March 2023 issue of *Frontline* magazine. I cannot recall any other software or tool making such headlines in such a short amount of time. It would not be surprising if ChatGPT is chosen as *Time* magazine's 'Person of the Year' for 2023, much like 'The Computer' was chosen in 1982.

As we end this series at a time when there is a major breakthrough in the field of AI, I offer these parting words of advice: formally learn the mathematical concepts introduced in this series, delve deeper into the Python libraries, and start developing a project that interests you which is of intermediate size and complexity, in order to practice the skillsets you have gained through this series.

I end with the hope that the series has been beneficial to you, and I extend my best wishes for an enjoyable and thrilling journey in the domain of AI. END

By: Dr Deepu Benson

The author is a free software enthusiast and his area of interest is theoretical computer science. He maintains a technical blog.

Implementing a CNN Deep Learning Model with TensorFlow

Deep learning has impressive data learning and prediction capabilities, making it very useful for a wide range of industries. TensorFlow is a popular open source library for deep learning applications because it is versatile, scalable, and can integrate with other tools. This article demonstrates how to implement a convolutional neural network (CNN) model with TensorFlow.

When it comes to developing and training machine learning models, TensorFlow is an extremely useful and versatile deep learning library. Its popularity and large community also make it a good choice for those looking to learn and apply deep learning in their work. Figure 1 shows several reasons why TensorFlow is a popular choice for deep learning frameworks.

Comparison of TensorFlow, PyTorch, and Keras

Three widely used deep learning frameworks are Keras, TensorFlow, and PyTorch. Keras' application programming interface (API) is a set of high-level building blocks that is intuitive, flexible, and easy to use. TensorFlow provides a low-level interface to create a neural network and high-level API (such as Keras) for easy and efficient model building. PyTorch's dynamic computational network enables more flexible and efficient model construction than TensorFlow.

Table 1 presents a comparison of these three deep learning frameworks. Overall, each framework has its own set of strengths and drawbacks, and their selection is determined by the use case.

Key programming elements in TensorFlow

Tensors, variables, and placeholders are the key programming elements in TensorFlow, making it a powerful platform.

Tensors: Tensors are TensorFlow's fundamental data structure. They are multidimensional arrays, similar to NumPy arrays, but with additional features such as support for GPU acceleration and automatic differentiation. In the TensorFlow library, a tensor can be created

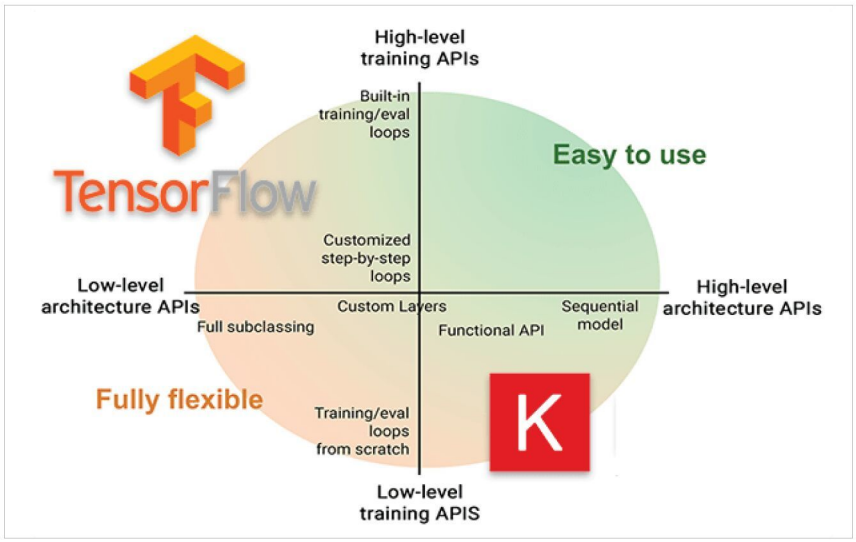


Figure 1: Why TensorFlow is popular for deep learning frameworks

```
# example of loading and plotting the mnist dataset
from tensorflow.keras.datasets.mnist import load_data
from matplotlib import pyplot
# load dataset
(trainX, trainy), (testX, testy) = load_data()
# summarize loaded dataset
print('Train: X=%s, y=%s' % (trainX.shape, trainy.shape))
print('Test: X=%s, y=%s' % (testX.shape, testy.shape))
# plot first few images
for i in range(25):
 # define subplot
 pyplot.subplot(5, 5, i+1)
 # plot raw pixel data
 pyplot.imshow(trainX[i], cmap=pyplot.get_cmap('gray'))
# show the figure
pyplot.show()
```

Figure 2: TensorFlow Python code to implement CNN deep learning model

Table 1: TensorFlow vs PyTorch vs Keras

Criteria	TensorFlow	PyTorch	Keras
Developed by	Google	Meta (Facebook)	Google
Written in	C++, Python, CUDA	Lua	Python
Compatibility	Large data set	Large data set	Small data set
Debugging capability	Difficult	Good	Not required
Choice of use	Scalability and performance	Flexibility and dynamic nature	Ease of use and simplicity
Popular industry use cases	Image classification, language translation, speech recognition, etc	Computer vision, NLP, recommendation system, etc	Speech recognition, healthcare, sentiment analysis, text classification

using the following command:

```
import tensorflow as tf
# Creating a tensor
a = tf.constant([5, 7])
print (a)
```

Variables: Parameters of a model, such as a neural network's weights and biases, are stored and updated in variables. They are created using the *tf.Variable()* function, which takes an initial value as its argument. The following command can be used to create variables in TensorFlow:

```
import tensorflow as tf
# Creating a variable
b = tf.Variable(23)
print (b)
```

Placeholders: When training or inferencing a TensorFlow model, placeholders are frequently used to inject input data into the model. You can use the *tf.placeholder()* function to create a placeholder in TensorFlow. This function takes two arguments: the data type of the tensor that will be fed into the placeholder and the shape of the tensor (which can be optional). Here is an example of creating a placeholder for a 2D tensor of floats:

```
import tensorflow as tf
# create a placeholder for a 2D tensor
of floats
x = tf.placeholder(tf.float32,
shape=(None, 2))
```

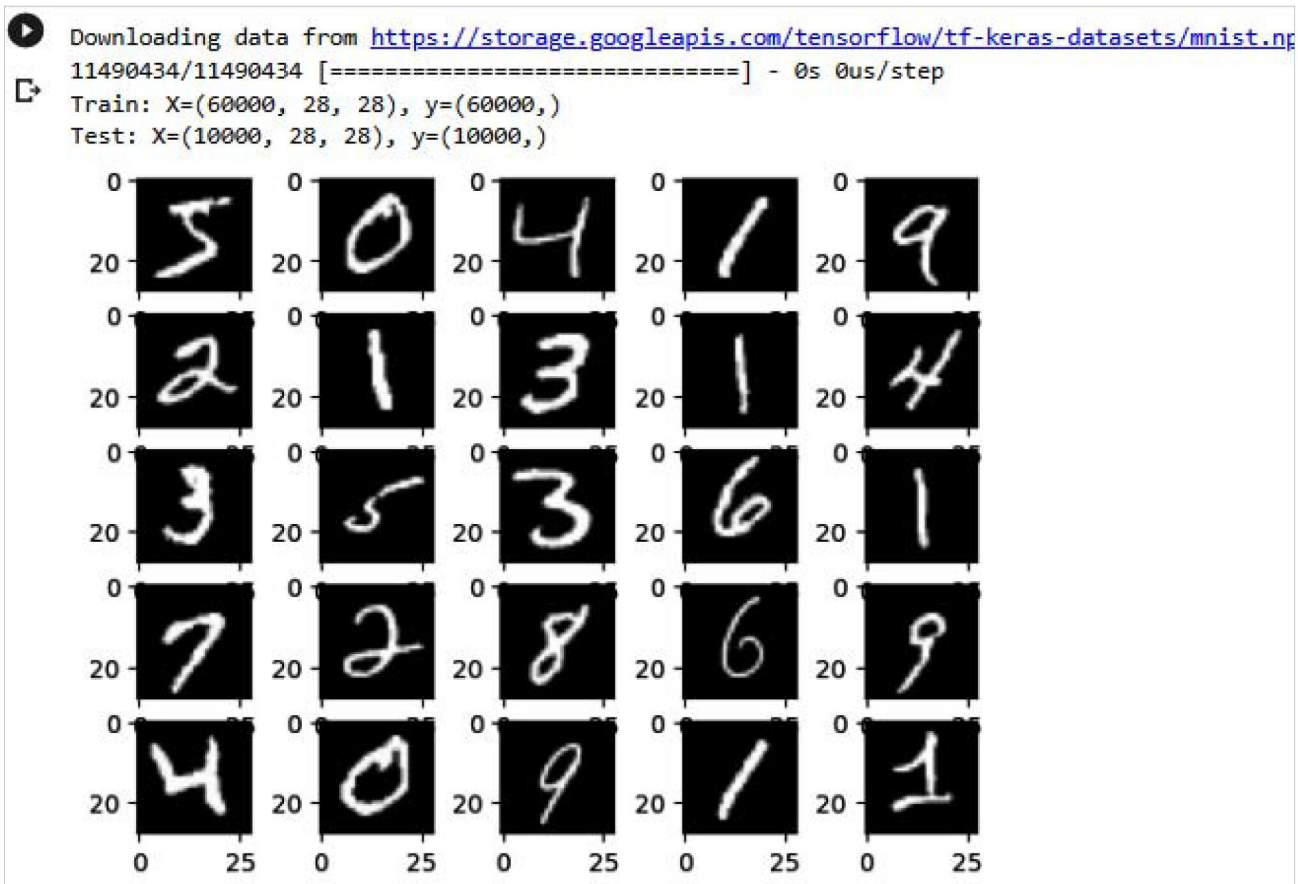


Figure 3: Output of CNN model implemented using MNIST data set

Implementing a CNN deep learning model using TensorFlow

A CNN deep neural network is frequently utilised for image categorisation, object detection, and other computer vision applications. Figure 2 shows the Python code example executed on Google Collab notebook for implementing a CNN deep learning model using TensorFlow.

We have trained the CNN model using the MNIST data set in this example. In the machine learning community, the MNIST data set is frequently used as a benchmark for image classification tasks, notably for assessing the performance of deep learning models like CNNs. It is a popular data set due to its simplicity and accessibility, and has been used for various applications including handwriting recognition, digit classification, and optical character recognition.

The output of our implemented CNN deep learning model using the MNIST data set is shown in Figure 3. The data set consists of 70,000 images, each of which is a grayscale image of a handwritten digit from 0 to 9. The images are 28x28 pixels and centred within a 32x32-pixel image.

TensorFlow is a Google open source software library. It was designed for tasks requiring complex numerical computations, and is popular because it supports Python and C++ API. It has faster compile times and supports CPUs as well as GPU distributed processing. Overall, TensorFlow's flexibility, scalability, and facilitation of neural network architectures make it well-suited for various industry applications. END

References

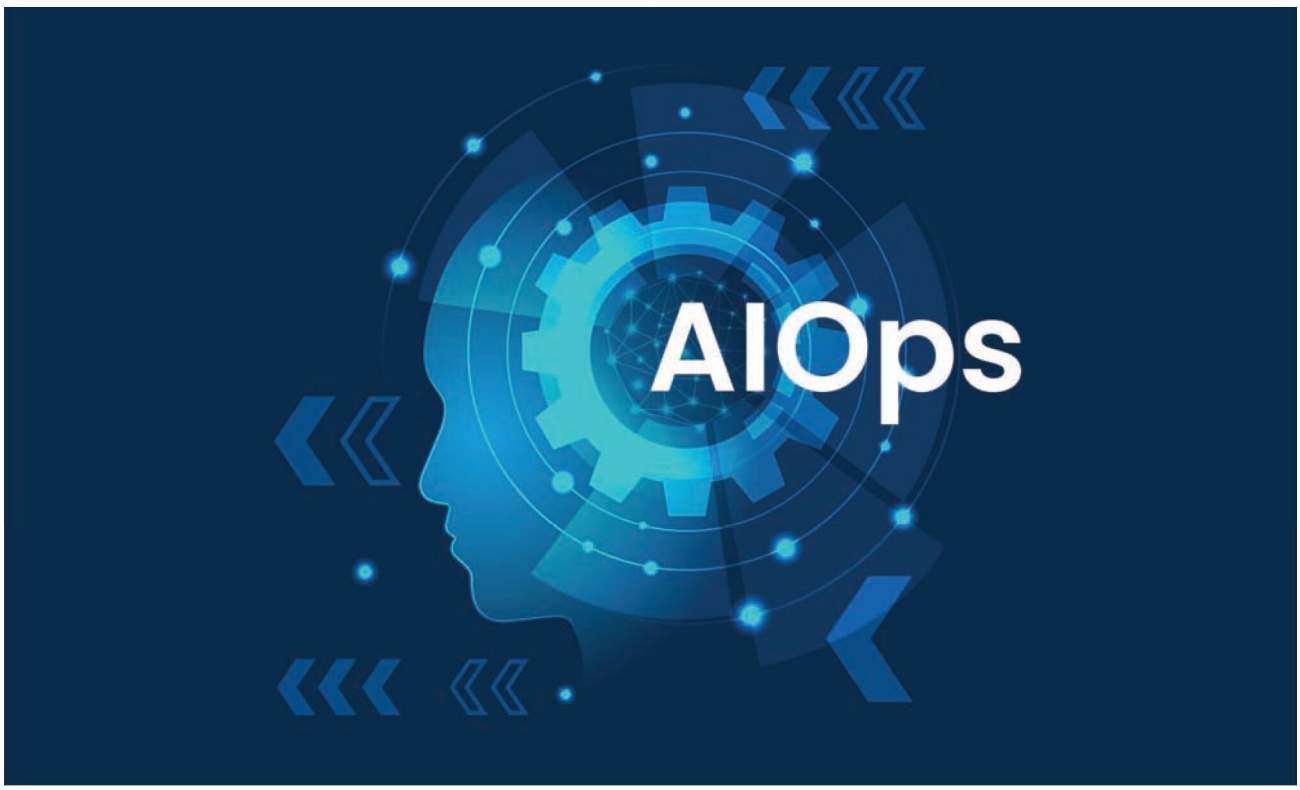
- *https://www.tensorflow.org/api_docs*
- *https://storage.googleapis.com/tensorflow/tf-keras-datasets/mnist.npz*
- *https://www.tensorflow.org/tutorials/images/cnn*

By: Dr Aditya Bhardwaj

The author is a B. Tech, M. Tech, and PhD in CSE. He works as assistant professor in the School of Computer Science Engineering and Technology at Bennett University, Greater Noida. He is experienced in cloud computing and open source technology.

Use AIOps to Make Your Enterprise Agile

Artificial intelligence for IT operations or AIOps is a boon for the modern day enterprise. Read on to find out its benefits, and get to know the popular AIOps platforms.



The adoption of new business models by an enterprise due to digital transformation has increased the complexity of IT operations (ITOps). As a result, existing IT tools can no longer meet enterprise demands. The biggest challenges ITOps face today are:

- Poor user experience on applications across geographies
- Lack of effective event management and no proactive monitoring
- More manual intervention and efforts, and minimum automation
- Inability of component-level drill down across technology platforms for a quick problem and root cause analysis
- Lack of single-pane view of IT and business process metrics

Organisations have now started leveraging artificial intelligence (AI) for IT operations (AIOps) with APM (application performance monitoring) and other data sources to gain insights that improve business outcomes.

AIOps is the use of artificial intelligence (AI) for IT operations to enhance, support and automate the latter. It covers the strategic use of AI, analytics, and machine learning (ML) technologies across IT operations to simplify and streamline processes and optimise the use of IT resources.

It can be considered as a platform consisting of AI and ML engines, Big Data capabilities, and servers covering storage, compliance, infrastructure, provisioning, and backup. The AIOps platform helps in:

- Improving and automating event monitoring
- Ingesting both historical data and real-time streaming data from across the IT environment
- Filtering out the noise so only the most relevant data is analysed

- Better service management
- Modernising IT operations
- Implementing security operations (SecOps), network operations (NetOps) and development operations (DevOps) by using AI to automate IT

Industry adoption of AIOps

According to ESC Research, nearly 30 per cent of the organisations surveyed plan to make significant investments in AIOps over the next 12 to 18 months, and more than 90 per cent expect to spend as much or more on AI and machine learning in 2023.

Gartner predicts that the number of business leaders relying on AIOps platforms for automated insights will increase 10 times by 2024.

As per an IDC report, by 2024, 30 per cent of enterprises will extend attention networks across IT teams, including AIOps.

According to Research and Markets, the global market for AIOps platforms is projected to reach US$ 22.9 billion by 2030, growing at a CAGR of 30.4 per cent between 2022 and 2030.

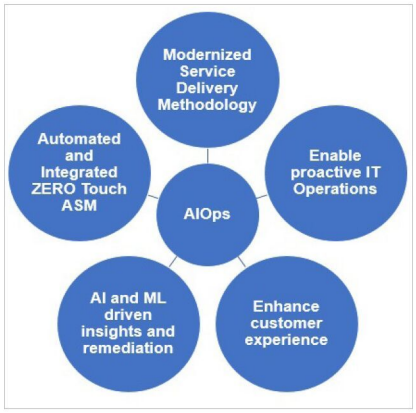


Figure 1: Objectives of AIOps

The key objectives of AIOps are (Figure 1):

- Monitoring
- Event correlation
- Auto ticketing
- Anomaly detection
- Business IQ
- Business transactions monitoring
- Auto remediation
- RCA/Diagnostics

The AIOps framework

AIOps services focus on optimisation, simplification, automation, and elimination to improve the resilience of IT systems, leading to an enhanced customer experience. The AIOps framework enables proactive end-to-end business — IT monitoring and analytics, next-gen event management, and robotic process automation (Figure 2). The aim is to simplify IT operations.

Let's take a look at the components of the framework of an AIOps platform.

Early detection: It is important to continuously monitor all business-critical applications for availability and performance. A significant amount of time and effort is spent on activities such as application monitoring, batch monitoring, etc. Early detection helps in:

- Early warning of potential performance bottlenecks
- Increased application accessibility for a better user experience

Data collection: A key aspect of AIOps is comprehensive analytics that leads to actionable insights. This involves ingestion of data from multiple sources that are vendor-agnostic, storage of the acquired data, real-time analysis at the point of ingestion and historical analysis

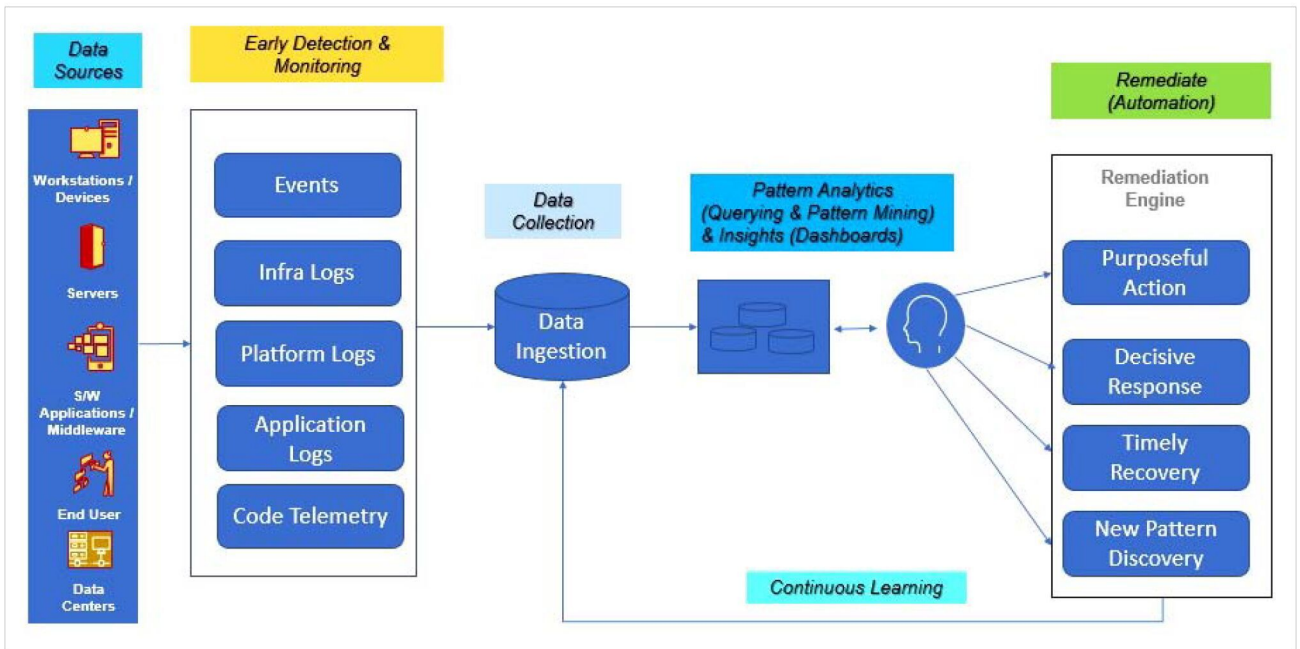


Figure 2: AIOps framework

of stored data, leveraging machine learning and, finally, preventive and remedial actions based on the analysis.

Pattern analytics: This offers real-time analysis and visualisation of automatically collected and correlated data to get insights into IT operations, customer experience and business outcomes. It helps to:

- Analyse rich and extensible data sets to connect the dots between IT operations, user experience and business impact
- Easily collect and correlate user, performance and business data in real-time with no code changes
- Utilise SQL on rich and extensible data sets to run *ad hoc* analysis in real-time and dig deeper into specific performance issues

These insights provide a holistic view of business process and application metrics, which gets generated from different monitoring and automation systems.

Remediation: This helps to reduce the effort put in tasks related to tickets, which can be automatically addressed via orchestrated platforms and self-service solutions. Remediation helps in:

- Reduction of execution time for tasks/processes
- Increased operational efficiency with reduced mean time to resolution

Open source AIOps platforms

Most open source AIOps projects use Python as a programming language for machine learning. Based on the enterprise requirement, various AIOps and open source tools can be combined and used on AIOps platforms. We take a brief look at the top open source AIOps platforms/tools.

SeldonIO: This open source platform deploys enterprise machine learning models on Kubernetes at a massive scale. Seldon handles scaling to thousands of production machine learning models and provides advanced machine learning capabilities out-of-the-box, including advanced metrics, request logging, explainers, outlier detectors, A/B tests, canaries, and more.

Logpai/Loglizer: This is a machine learning-based log analysis toolkit for automated anomaly detection. Loglizer provides a toolkit that implements a number of ML-based log analysis techniques that have multiple supervised and unsupervised models.

Whylabs/Whylogs: Whylogs is an open source statistical logging library that allows data science and ML teams to effortlessly profile ML/AI pipelines and applications, producing log files that can be used for monitoring, alerts, analytics, and error analysis. It's available in Python and Java.

Jixinpu/Aiopstools: This fundamental package for AIOps with Python offers the following:

- Anomaly detection
- Alarm convergence
- Time series forecasting method
- Association analysis for alarms

Log-anomaly-detector: This platform is developed using ML, and is used for log anomaly detection by connecting to streaming sources to predict abnormal log lines. It uses unsupervised machine learning models to achieve this result.

Prometheus: This open source monitoring solution simplifies pulling numerical metrics from a metrics endpoint.

Grafana: This open source metric analytics and visualisation suite is popular among Prometheus users to visualise the metrics.

Elastic Stack: This is a suite of open source products from Elastic designed to help users search, analyse, and visualise data from any type of source, in any format, in real-time. It provides monitoring and logging solutions.

Benefits of AIOps adoption

- Improved application availability and customer satisfaction.
- Minimised application downtime on even the busiest and highest transaction days
- Expensive service disruptions are avoided, and firefighting eliminated
- Continual management of vulnerability risks. AIOps tools help to identify, analyse, prioritise and remediate vulnerability risks
- Intelligent alerts to prevent potential issues
- Increased business responsiveness
- Helps make data-driven decisions
- Helps to meet growing business demands
- Increased efficiency and optimised running costs with the help of automation and AI
- Reduced human intervention and efforts; focus on innovation

The goal of AIOps is automation, which helps in simplifying administrative tasks, thus saving time. An AIOps platform helps to scale up IT operations to support ever-growing business demands, ensuring an enhanced customer experience. It reduces the complexity of IT operations by streamlining systems, configuration management, simplifying operations and improving reliability. In short, it helps to enhance the performance of enterprise operations. END

Acknowledgements

The author would like to thank Santosh Shinde of BTIS, Enterprise Architecture division of HCL Technologies Ltd for giving the required time and support in many ways while this article was being written as part of Architecture Practice efforts.

By: Dr Gopala Krishna Behara

The author is an enterprise architect in the BTIS Enterprise Architecture division of HCL Technologies Ltd. He has a total of 27 years of experience in the IT industry.

How Amazon's Services can Ensure You Don't Miss Your Flight

Large airports and long queues can sometimes put even the most disciplined passenger at the risk of missing a flight. Amazon has a range of AI and ML-based services that can be used at airports to ensure that passengers who reach the airport on time never miss a flight.

Image Source: skift

Jayashree arrived at Dubai airport four hours before her flight was scheduled to depart. She was excited about going home and seeing her loved ones. She checked in her luggage, went through immigration and security, and was soon making her way through the duty-free shops and restaurants. Since she had a few hours to kill, she decided to do some shopping and grab a quick bite. But time flew while she shopped and ate, and she suddenly realised her flight was about to leave in just half an hour. She gathered her stuff and rushed towards the boarding gate.

The airlines had begun the boarding procedure an hour before the scheduled departure, and the boarding was completed within 20 minutes. However, one passenger was still missing. An announcement was made repeatedly for Jayashree to rush to the boarding gate, but there was no response. A staff member from the airline went looking everywhere for her, but could not locate her. He notified his failure to the airlines, the gate was closed and the flight started 15 minutes prior to the scheduled departure.

Jayashree arrived 15 minutes before her flight's scheduled departure time. She saw to her dismay that the gate was already closed, and the aeroplane was starting to pull away. Her heart sank as she realised that she had missed her flight. She was devastated that her plans to see her family, especially her 7-year-old daughter, had been ruined; and now she had to figure out how to go to India as soon as possible.

She approached an airport staff member waiting for her, and tried to explain that she had lost track of time while shopping and dining. "But why did the flight depart 15 minutes ahead of the scheduled departure time?" she asked.

"You were supposed to be at the gate at least 30 minutes before departure," the staff member said. "We can't wait for passengers who are not here on time."

With a heavy heart, Jayashree booked a hotel room for the night and waited to board the next flight to India.

The challenges of boarding a flight

It is not uncommon for passengers to miss their flight even after receiving their boarding pass, as we saw in the above anecdote. There are several reasons why this may happen.

Underestimating the time required: Passengers may not realise how much time they need to go to the gate, especially if they are unfamiliar with the airport or have long distances to cover. They may also underestimate the time required for security checks and other procedures.

Delayed flights: Even if the boarding pass has been issued, flights can be delayed due to various reasons such as weather, technical issues, or air

traffic congestion. Passengers may not realise that their flight has been delayed and miss the boarding time.

Slow boarding: The boarding process can take a long time, especially if passenger's do not follow the airline's instructions. This can lead to delays, missed connections, and other problems.

Security issues: Passengers may have problems with security, such as not being able to bring certain items on board or having to go through additional screening. This can cause delays.

Communication issues: Airlines may not provide clear instructions or communicate effectively with passengers during the boarding process. Last minute gate changes can lead to confusion and delays.

Hassle-free flight boarding experience with AI and ML

AI and ML can help resolve most of the above problems. Let's use the same anecdote to see how.

Jayashree had been eagerly waiting to visit her family in India for months. She arrived at Dubai airport four hours before her flight, checked-in her luggage and completed the immigration and security checks. Since she had several hours before her flight, she decided to do some shopping and grab some food. However, she lost track of time and didn't know that she should reach the boarding gate 30 minutes before the scheduled departure of the flight.

Image courtesy: Oleksandr Pidvalnyi

Fortunately, the airport had an AI-powered boarding system that ensured seamless boarding for all passengers. The system used IoT and machine learning to keep track of passengers' movements in the airport and estimate their time of arrival at the gate.

When Jayashree was predicted late, the system immediately alerted the gate agent and airline staff to locate her. The system used facial recognition and other tracking technologies to pinpoint her location in the airport, and within minutes, a staff member was able to find her and escort her to the gate.

Jayashree was able to board her flight without any hassle and also enjoy her shopping experience — all thanks to the seamless and efficient boarding process facilitated by AI, IoT, and machine learning.

AI, ML, data analytics and IoT services provided by AWS

The hassle-free flight boarding experience described above can happen easily by leveraging AWS cloud services.

There are a lot of IoT devices available to track the movements of a passenger/airline staff using geofencing and BLE (Bluetooth low energy). Beacons, for example, can cover a range of 5m to 2.5km. They emit signals to devices like mobile phones, and are mapped with their location. This information is stored in AWS RDS, which helps to detect the location of the passenger. Motion cameras can also be used.

AWS provides a range of AI, ML, data analytics, and IoT (Internet of Things) services that can be used to build intelligent and data-driven applications. Some of these key services are listed in Table 1.

AI SERVICES	Machine Learning services	Data Analytics services	IOT Services
Amazon Rekognition	Amazon SageMaker	Amazon Athena	AWS IoT Core
Amazon Lex	Amazon Forecast	Amazon EMR	AWS Greengrass
Amazon Comprehend	Amazon Fraud Detector	Amazon Kinesis	AWS IoT Analytics
Amazon Translate	Amazon CodeGuru	Amazon Redshift	AWS IoT Device Defender
Amazon Polly	Amazon Elastic Inference	Amazon QuickSight	AWS IoT Device Management
Amazon Transcribe	AWS Deep Learning AMIs	AWS Glue	Amazon FreeRTOS
Amazon Textract	AWS Deep Learning Containers	AWS Data Pipeline	AWS IoT 1-Click
Amazon Personalize	AWS DeepLens	AWS Lake Formation	AWS IoT Events
Amazon Kendra	AWS DeepRacer	Amazon MSK	AWS IoT SiteWise
Amazon Lookout for Vision	AWS DeepComposer	Amazon Elasticsearch Service	AWS IoT Things Graph
Amazon Lookout for Equipment		AWS Cassandra Service (MCS)	
Amazon Augmented AI (Amazon A2I)		AWS Data Exchange	
Alexa			

Table 1: Key web services offered by Amazon

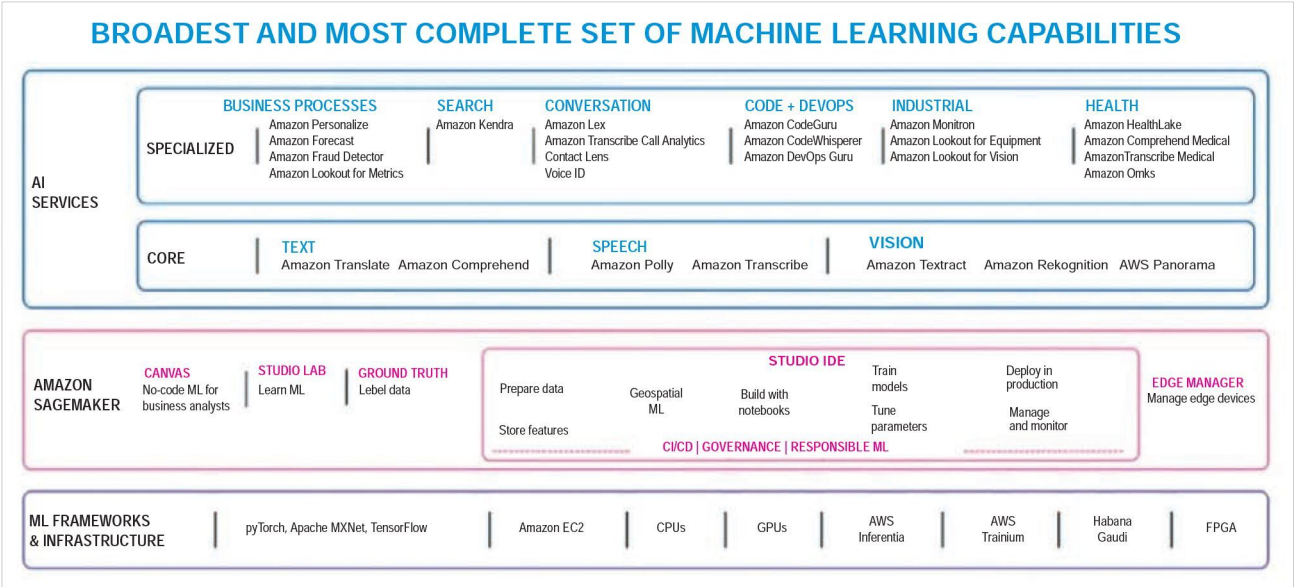


Figure 1: AWS AI/ML stack

For a seamless flight boarding solution, the following services can be used.

Amazon SageMaker: A fully-managed service that provides developers and data scientists with the tools to build, train, and deploy machine learning models at scale.

Amazon Rekognition: A deep learning-based image and video analysis service that can detect objects, faces, and text in images and videos.

Amazon Comprehend: A natural language processing service that can analyse text and extract insights such as sentiment, entities, and key phrases.

Amazon Polly: A text-to-speech service that can convert text into lifelike speech in multiple languages and voices.

Amazon Translate: A neural machine translation service that can translate text between languages with high accuracy.

Amazon Transcribe: An automatic speech recognition service that can transcribe audio and video files into text.

Amazon Forecast: A fully managed forecasting service that uses machine learning to deliver highly accurate forecasts.

Amazon Personalize: A service that provides real-time recommendations for content, products, and services using machine learning algorithms.

Amazon SageMaker

Amazon SageMaker is a powerful machine learning platform with a standard interface that provides a complete set of tools and services to build, train, and deploy machine learning models at scale quickly. SageMaker uses containers to wrap favourite algorithms and frameworks, including built-in algorithms like XGBoost, DeepAR, and FM, as well as frameworks like PyTorch, SKLearn, and TensorFlow.

Some of the key services provided by Amazon SageMaker include:

Data labelling: SageMaker's Ground Truth provides a fully managed data labelling service that makes it easy to label data sets using human annotators or pre-built machine learning models.

Model building: SageMaker provides a range of built-in algorithms and frameworks for building, training, and deploying machine learning models. It supports custom algorithm development using popular frameworks like TensorFlow, MXNet, and PyTorch.

Model training: SageMaker provides a scalable and distributed training environment (using GPUs and multiple instances) that helps train efficient machine learning models quickly.

Model hosting: It provides a fully managed model hosting environment enabling developers to deploy machine learning models as APIs with auto-scaling, monitoring, and debugging capabilities.

Model tuning: Amazon SageMaker provides an automatic model tuning service that enables data scientists to optimise hyperparameters and improve model accuracy without manual intervention.

Real-time inference: It provides a fully managed, highly available real-time inference service that can scale to handle millions of requests per second.

Batch inference: Amazon SageMaker provides a fully managed batch inference service that can process large volumes of data and give predictions in a cost-effective manner.

End-to-end ML workflow: It provides an end-to-end machine learning workflow that includes data preparation, feature engineering, model training, deployment, and monitoring.

Integration with other AWS services: Amazon SageMaker integrates with other AWS services like S3, Lambda, Step Functions, and CloudFormation to provide a seamless machine learning experience.

The role of AWS Rekognition, IoT Core and Greengrass

IoT based beacons and cameras can be installed throughout the airport terminal to detect passenger movements. AWS Greengrass can be used to deploy AWS Lambda functions that capture images from local cameras/sensors and send them to AWS Rekognition for analysis. The camera can capture images or videos at regular intervals (5/10 seconds) and send them to AWS IoT Gateway.

AWS Rekognition's face detection ability can be used to analyse the images or videos, detect the presence of individual passengers, and track their movements. Based on the analysis results, alerts can be generated, and notifications can be pushed if there is a high passenger density or congestion in a specific area, or if a passenger is moving to a wrong gate.

The analysis results can be sent back to AWS Greengrass for further processing or action, such as triggering local alarms or notifications.

The making of an AI/ML-based flight boarding solution

Algorithm selection

Our solution uses the XGBoost algorithm for binary classification, a popular choice for predicting an event's occurrence based on a set of input features. We are trying to predict if a checked-in passenger who is shopping will reach the boarding gate 30 minutes prior to the flight's departure time.

The built-in XGBoost algorithm in SageMaker makes it easy to train and deploy powerful machine learning models. With a little bit of data preparation and some tuning of hyperparameters, building models quickly to make accurate predictions on a variety of tasks is a straightforward process.

SageMaker provisions Jupyter Notebook in the cloud, which is easy to create and use. Here are the steps.

Prepare data: Our data should be in a format that XGBoost can work with. This typically means a CSV file with columns for features and a target column for the label.

Upload data: We need to upload data to an S3 bucket so that SageMaker can access it.

Create a training job: We can create a training job by specifying the location of our training data in S3/Datalake and the hyperparameters we want to use for the XGBoost algorithm. This can be done through the SageMaker console, the SageMaker SDK, or the AWS CLI.

Monitor the training job: Once the training job is started, we can monitor its progress through the SageMaker console or the SDK.

Deploy the model: After the training job is complete, we can deploy the trained model as an endpoint that can be used for inference.

Test the model: We can test the deployed model by sending new data and observing the predictions it produces.

Input data

The input data for our model consists of a set of features related to the passenger's movement within the airport, their shopping behaviour, and their time of arrival at the airport. These features are collected using IoT devices such as cameras, beacons, and sensors placed throughout the airport.

The following are the input features we used in our model.

Time of arrival: The data on the time at which the passenger arrives at the airport is collected from the passengers who have already done a web check-in. The IoT sensors at the entrance of the airport collect this data and pass it to the backend data lake. For non-web checked-in passengers, this is collected from the counter or the kiosk from where the boarding pass is obtained.

Passenger state: This indicates if the passenger has cleared immigration, security, checked-in, etc.

Shopping duration: This is the amount of time passengers spend shopping after getting their boarding pass and immigration/security clearance.

Distance from gate: This is the distance between the shopping area and the boarding gate, and is calculated using the IoT sensors placed throughout the duty-free shopping area.

Gate arrival time: This is the time at which the passenger arrives at the boarding gate. The prediction time will keep on getting calculated and updated in another table, till the passenger reaches the boarding gate. Continuous push message alerts will be sent to the passenger's mobile phone/smart device and also to the shopkeeper in the proximity of the passenger. At the right time, the shopkeeper's help will be sought to alert the passenger to move towards the gate to board the flight, automatically. The TV display will also alert the passenger to move towards the gate. The passenger will also be tagged on social media and asked to move towards the boarding gate. All these things will happen automatically thanks to various AWS AI/ML, IoT and other services.

Scheduled departure time: This is the scheduled departure time of the passenger's flight.

Flight delay: This indicates any delay in the scheduled departure time of the passenger's flight.

Walking pattern: This is the walking pattern captured from the passenger's wearable device or smartphone. If not available, the default value will be used.

Hand luggage: This data will be collected at the time of check-in but may become inconsistent due to shopping add-ons.

Terminal congestion: This is a measure of how busy the terminal is at the time the passenger arrives at the airport, and on the path to the gate. A heat map will also be displayed on the passenger's smart device.

Data preparation

Before training our model, we performed some data preparation steps, as follows.

Cleaning: Removing any invalid or missing data.

Preprocessing: Scaling the numerical features to ensure they have the same range, and normalising the data to improve the model's performance.

Feature engineering: Creating new features based on the existing data to improve the model's accuracy.

Model training and deployment

Once we had prepared our data, we used Amazon SageMaker's built-in XGBoost algorithm to train our binary classification model. We then deployed the trained model using SageMaker's model deployment service, which automatically scales the model to handle high volumes of traffic.

Integration with IoT devices

We then integrated the deployed machine learning model with IoT devices such as cameras and sensors to monitor passengers' movements and predict their arrival times at the gate.

Alerting system: To build an alerting system, we used Amazon Simple Notification Service (SNS) to notify gate agents and airline staff when passengers are predicted to be late.

Tracking system: To build a tracking system, we used facial recognition and other tracking technologies to locate passengers who are predicted to be late and escort them to the gate.

Overall, this implementation of predictive analysis using SageMaker can help airlines optimise their boarding process by predicting which passengers may be at risk of missing their flight and taking proactive measures to ensure they reach the gate on time.

Sample SageMaker code

SageMaker's built-in XGBoost algorithm was used to predict if a checked-in passenger will reach the boarding gate on time. Data was split into training and validation sets, pre-processed with Pandas, and saved to S3. The model was trained and deployed to make predictions on new data. The model's accuracy was evaluated with a validation set, and metrics like confusion matrix and classification report were printed as shown in Figure 3.

```
import boto3
import sagemaker

# Set up the SageMaker session and role next, Define the S3 bucket and prefix for storing the data and model
sagemaker_session = sagemaker.Session()
role = sagemaker.get_execution_role()
#
bucket = 'your_s3_bucket_name'
prefix = 'sagemaker/xgboost_model'

# Load the input data from S3
data_location = 's3://{}/{}'.format(bucket, prefix)
data = sagemaker.inputs.TrainingInput(data_location, content_type='text/csv')

# Set up the hyperparameters for the XGBoost algorithm
hyperparameters = {
    'max_depth': '5',
    'eta': '0.2',
    'gamma': '4',
    'min_child_weight': '6',
    'subsample': '0.7',
    'objective': 'reg:squarederror',
    'num_round': '100'}

# Configure the estimator with the XGBoost algorithm and the hyperparameters
estimator = sagemaker.estimator.Estimator(
    image_uri=sagemaker.amazon.amazon_estimator.get_image_uri(
        sagemaker_session.boto_region_name, 'xgboost', '0.90-1'),
    role=role,
    instance_count=1,
    instance_type='ml.m5.large',
    output_path='s3://{}/{}/output'.format(bucket, prefix),
    sagemaker_session=sagemaker_session,
    hyperparameters=hyperparameters)

# Train the model with the input data
estimator.fit({'train': data})

# Deploy the trained model to an endpoint
predictor = estimator.deploy(initial_instance_count=1, instance_type='ml.t2.medium')

# Use the deployed model to make predictions and Delete the endpoint
test_data = '0,1,2,3,4,5,6,7,8,9'  # Replace with your test data
prediction = predictor.predict(test_data)

predictor.delete_endpoint()
```

Figure 2: Sample SageMaker code

	Passenger Name	Time of Arrival	Passenger State	Shopping Duration	Distance from Gate	Gate Arrival Time	Flight Departure Time	Flight Delay	Walking Pattern	Hand Luggage	Terminal Congestion
2	Jayashree	07:20	Late	15	250	07:30	08:00	00:30	Slow	2	Moderate
3	Precila	08:05	On Time	10	150	08:15	09:00	00:00	Moderate	1	Low
4	Abdul	07:40	Late	20	300	07:50	08:30	00:15	Fast	3	High

Figure 3: Classification report

As we saw in this article, by combining AWS IoT, AI, ML, and other services, airports can provide a smoother and more enjoyable passenger onboarding experience. This not only benefits passengers but also enhances airport operations, making them more efficient and cost-effective. END

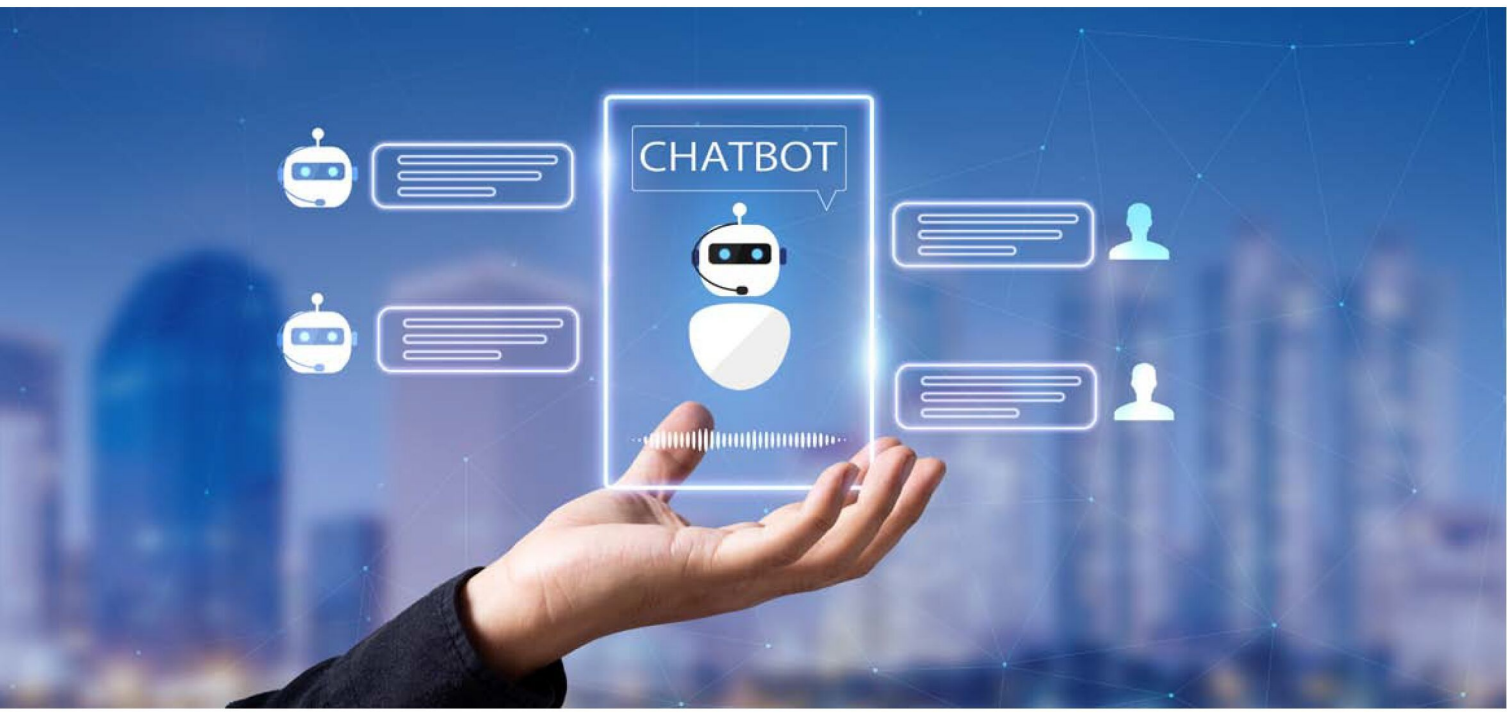
Acknowledgements

I would like to thank Balakrishnan Srinivasan (CTO - hybrid cloud application modernization services, IBM India) for encouraging me to write this; Gopalakrishna U. (associate partner, IBM India) for providing me an opportunity to gain real-time experience on this subject; and Harsh Mehta (delivery manager, IBM India) who took the effort to help me release this article for publishing.

By: Venkateswaran N.

The author is a lead architect, containerization and modernization, AWS practice, IBM India.

Making a Difference with Generative AI



Everyone seems to be using ChatGPT for something or the other. Here, our experts suggest some serious, game-changing applications that the technology can be put to—and the pitfalls to avoid.

Social media is abuzz about people's experiments with ChatGPT. Amongst other things, it seems to be capable of debugging and writing code (even small apps), drafting essays, poetry and emails, having a meaningful conversation with users, planning your vacation, telling you what to pack for a business trip, preparing your shopping list, extracting tasks from a conversation or meeting minutes, summarising a long text into a brief overview, writing a new episode of Star Wars, and much more!

And that is not all. Developers can also utilise the power of OpenAI's AI models to build interactive chatbots and advanced virtual assistants, using the application programming interface (API). They can use the GPT-3 API, or join the waitlist for the GPT-4.

Many companies are also using OpenAI's generative AI models to enhance their own applications and platforms. OpenAI offers multiple models, with different capabilities and price points. The prices are per 1,000 tokens, so customers can pay for what they use.

DuoLingo uses GPT-4 to deepen conversations, while Be My Eyes uses it to enhance visual accessibility, and Stripe uses it to combat fraud. Morgan-Stanley is using GPT-4 to organise its knowledge base, while the government of Iceland is using it to preserve its language.

It is also believed that tools like ChatGPT and Dall-E (which generates images from textual prompts) will help advance the metaverse, as it enables people with no art or design background to design spaces, engage in meaningful conversations in the virtual world, and more.

"Generative AI is already being used for many art and creative domains, such as Firefly from Adobe and Picasso

from Nvidia. Similarly, there are also language-specific applications around generative AI, such as composing emails, creating a summary of documents, and detecting to-dos from call transcripts. The generative AI techniques used for images and text could also be used for other kinds of data, such as chemical compound data or application log data," says Sachindra Joshi, IBM Distinguished Engineer, Conversational Platforms, IBM Research India.

He cites the example of molecular synthesis. By capturing the language of molecules in a foundation model and using it to "generate" new ideas for drugs and other chemicals of interest, IBM Research created a large-scale and efficient molecular language model transformer that is trained on over a billion molecular text strings. This model performs better than all state-of-the-art techniques on molecular property prediction and captures short-range and long-range spatial relationships through learned attention. IBM has also partnered with NASA to build a domain-specific foundation model, trained on earth science literature, to help scientists utilise up-to-date mission data and derive insights easily from a vast corpus of research that would be otherwise challenging for anyone to thoroughly read and internalise.

"At IBM, we are also applying these advancements to automate and simplify the language of computing, i.e., code. Project CodeNet, our massive dataset encompassing many of the most popular coding languages from past and present, can be leveraged into a model that would be foundational to automating and modernising countless business applications. In October 2022, IBM Research and Red Hat released Project Wisdom, an effort designed to make it easier for anyone to write Ansible Playbooks with AI-generated recommendations—think pair programming with an AI in the "navigator" seat. Fuelled by foundation models born from IBM's AI for Code efforts, Project Wisdom has the potential to dramatically boost developer productivity, extending the power of AI assistance to new domains," he says.

Indeed, the potential uses of this tech are quite vast—with impacts ranging from casual to serious. We asked each of our experts to suggest a couple of applications, which they think can be seriously useful to businesses and/or society at large, and have documented these ideas below.

Enhancing accessibility

"ChatGPT has transformed how people access information and interact with technology. Its vast knowledge base and natural language processing capabilities have impacted individuals worldwide, making it a critical tool for those seeking advice, information, or even casual conversation," says Sanjeev Azad, Vice President - Technology, GlobalLogic.

Here is his selection of serious applications that ChatGPT can be used in:

1. ***Personalised customer support:*** ChatGPT can revolutionise the customer support industry by providing 24/7 customer support via chatbots. With the ability to understand human-provided inputs and natural language responses, ChatGPT can provide customers with personalised and immediate solutions to their problems without human intervention. This can significantly reduce response times and increase customer satisfaction.
2. ***Market research:*** ChatGPT can be used for market research

Cybersecurity threats posed by ChatGPT

Steve Grobman, CTO at McAfee, explains some of the cybersecurity related concerns to us. "When it comes to ChatGPT, one of the main considerations for risk is that the bot is lowering the bar of who can create malicious threats, and improving efficiency of tasks that traditionally require a human. For example, well-crafted unique phishing messages can be created at a scale, and a wide range of malware implementations can be built by even relatively unskilled individuals. ChatGPT has attempted to prevent malicious use cases. However, there are already internet posts on how to circumvent these restrictions. This includes using ChatGPT to build components that are benign on their own but can be stitched together to create malware," he says. "Any new method to defend against attacks needs the ability to understand how the attacks will be created. ChatGPT helps with this, as research can test the boundaries of what attacks ChatGPT can create. What is less clear is how directly ChatGPT can auto-generate elements of the defence. While there may be some efficiencies and unique insights that ChatGPT provides, many other tools, techniques and technology will be required to defend against ChatGPT curated attacks."

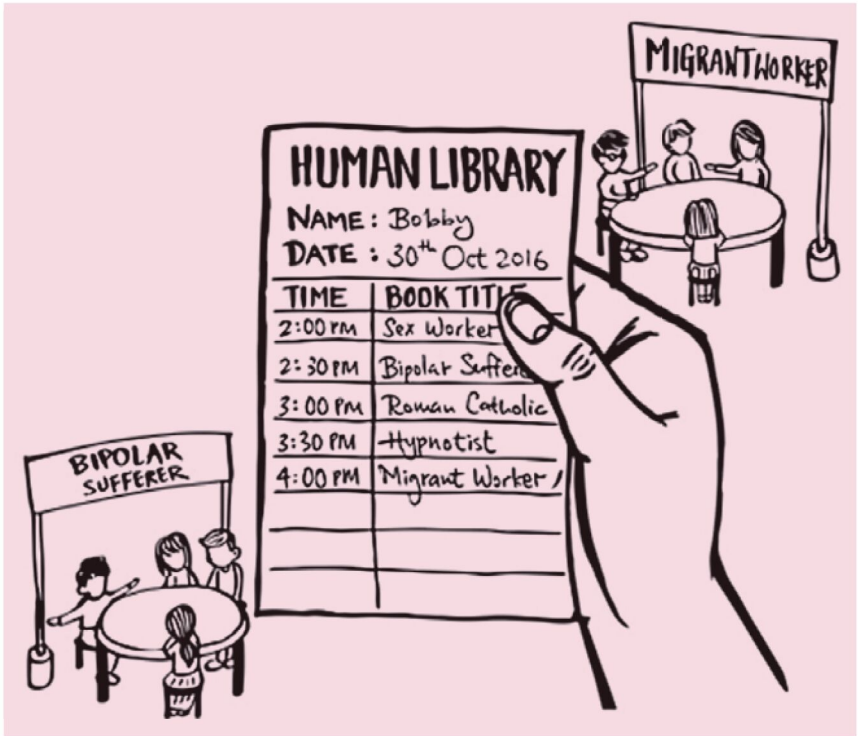
Within the threat landscape specifically, Grobman says we might witness ChatGPT's impact enhancing the effectiveness and efficiency of cyberattacks in the future. For example, it enables spear phishing to operate at the scale of traditional bulk phishing. Attackers can now use ChatGPT to craft automated messages in bulk that are well-written and targeted to individual victims, making them more successful. "Today's state-of-the art AI-authored content is challenging to differentiate from human-authored content. For example, McAfee recently conducted a survey where two-thirds of the 5,000 respondents could not differentiate between machine-authored and human-authored content," he says.

He explains that ChatGPT also lowers the barrier-to-entry, making technology that traditionally required highly-skilled individuals and substantial funding, available to anyone with access to the internet! This means that less skilled attackers now have the means to generate malicious code in bulk. For example, they can ask the program to write code that will generate text messages to hundreds of individuals, much like a non-criminal marketing team might. However, instead of taking the recipient to a safe site, it directs them to a site with a malicious payload. The code in and of itself is not malicious, but it can be used to deliver dangerous content.

He signs off, saying, "As with any new or emerging technology or application, there are pros and cons. ChatGPT will be leveraged by both good and bad actors, and the cybersecurity community must remain vigilant in the ways these can be exploited."

by analysing large volumes of customer data and extracting insights from it. With its extensive natural language processing (NLP) and natural language understanding (NLU) capabilities, ChatGPT can analyse data collected from feedback, reviews, and social media posts to identify trends, sentiments, and preferences. This can help businesses make data-driven decisions and stay ahead of their competitors.

3. ***Accessibility:*** This can be one of the potential areas where ChatGPT can bring a positive impact by improving ease of access for people with disabilities by providing natural language interfaces for technology. For example, people with visual impairment can use chatbots powered by ChatGPT to access information and services without relying on visual interfaces. Furthermore, by providing accurate and instant translation, ChatGPT can help improve communication across languages and cultures. This can aid in the removal of barriers and promotion of cross-cultural understanding.



The 'Human Library' project that is running successfully in several technologically advanced nations, shows that AI cannot replace humans... yet (Courtesy: Human Library SG)

From healthcare to learning, and more

"ChatGPT has changed the lives of people all over the world with its vast knowledge base and natural language processing capabilities, and has become an indispensable tool for those looking for information, advice, or even just a friendly chat," says Anurag Sahay, CTO and Managing Director - AI and Data Science, Nagarro.

Among many other use cases, its application in the following will be game-changing, according to him:

1. ***Healthcare:*** ChatGPT can be used as a tool for providing preliminary medical information and triaging. It can assess people's symptoms and narratives sympathetically, and use that information to come up with a potential diagnosis and recommend further tests, streamlining the healthcare process and reducing the workload of medical professionals. However, it is important to note that ChatGPT is not a substitute for professional medical advice and should only be used as the first step.
2. ***Personalised learning:*** ChatGPT can be used in education to further help students with their homework, clarify their concepts, provide explanations, generate practice problems, and adapt to each student's learning style and pace. It is an excellent intervention for educational improvement, as it can provide personalised assistance to students and help them learn better.
3. ***Real-time support for crisis response and information dissemination:*** During public health emergencies or crises, people are looking for real-time information and guidance to appropriate resources. ChatGPT can provide the first layer of intervention, guiding people to the latest information and alleviating the workload on emergency responders.

"Overall, ChatGPT has the potential to be a game-changer in these three areas and can significantly improve the quality of life for many people. However, it is important to use it responsibly and recognise its limitations as a tool for preliminary information and guidance, and should not be used as a substitute for professional advice," he says.

Augmenting enterprise AI with generative AI

"Organisations are re-imagining their core processes with generative AI. They are using it to realise rapid business value, like improving accuracy, near real-time insights into customer

concerns or issues, and driving better efficiency. At IBM, we see business value in augmenting existing enterprise AI deployments with generative AI to improve performance and accelerate time to value," says Joshi.

There are four categories, where generative AI can deliver, according to him:

1. ***Summarisation of text documents***—for instance, call centre interactions, financial reports, analyst articles, emails, news, and media trends
2. ***Semantic search***—from reviews, knowledge base, and product descriptions
3. ***Content creation***—like technical documentation, user stories, test cases, data, generating images, personalised UI, personas, and marketing copy
4. ***Code creation***—like code co-pilot, pipelines, Docker files, terraform scripts, converting user stories to Gherkin format, diagrams as code, architectural artifacts, threat models, and code for applications

"While the use cases around generative AI are exciting, the work involved in building generative AI solutions must be developed carefully and with critical attention to building trust, and devoid of hallucinations. Hence, business leaders must ensure that they put in place strong AI ethics and governance mechanisms to mitigate against the risks involved," he says, explaining that foundation models must protect data and privacy of business users, be explainable, auditable, and operate properly within the sometimes-sensitive parameters of their industry.

What it cannot do, and better not do!

It is evident that generative AI can do a lot. And, of these, ChatGPT has gained a lot of traction within a phenomenally short time. There are some who use it instead of Google. We only hope they are aware that ChatGPT is trained on data only up to September 2021 (as per the company's blog, even GPT-4 is trained with data only up to that). So, if you rely on info it provides, you are missing out on later developments! In fact, when we asked ChatGPT about GPT-4, it categorically said, "As of my knowledge cutoff in September 2021, there was no such thing as GPT-4."

(I sure am glad it does not have the latest information, or it might be writing this article instead of me!)

ChatGPT does not have access to the internet, so it cannot handle any question that requires it to look up Web resources. Bing Chat claims to augment GPT-4 capabilities with that of Bing Search, but the solution is still half-baked, and in trials with a limited user base. If they succeed in fine-tuning it, it might overcome a major limitation in ChatGPT.

Within a few months, ChatGPT has also raised a startling number of concerns, ranging from plagiarism to privacy and security. Plus, there is this very scary thought of how quality of education will spiral, if students make ChatGPT do their assignments and quizzes. Some websites like Stack Overflow have temporarily banned users from posting ChatGPT generated responses on forums. This is because the responses given by ChatGPT may not always be right (the company itself warns users about this), and it does not cite sources for information presented in its responses.

"There may be risks associated with using AI-generated content, such as spreading misinformation or manipulating public opinion through chatbots or deep fakes," says Azad.

ChatGPT is programmed not to respond to certain types of prompts, including those that involve hate speech or discrimination, invade one's privacy or violate someone's rights, which involve finance or investment advice, seek to promote misinformation or conspiracy theories (though it might sometimes not be able to clearly distinguish between such information and fiction), and so on.

"ChatGPT is generating a lot of buzz around the world, and it has been a game changer for many. However, it must be balanced against the potential concerns and risks, necessitating a significant increase in our efforts in responsible AI. One of the privacy risks is the information provided to ChatGPT through user prompts. When we ask the tool to answer questions or perform tasks, we may unintentionally reveal sensitive information. While there are no problems with using ChatGPT for publicly available data, caution must be exercised when uploading private information to the platform. We must take all necessary precautions to safeguard personally identifiable information (PII) and thus anonymise our personal data," warns Sahay.

"In addition to privacy, another concern is the inappropriateness of content. While we marvel at the versatility of ChatGPT and its myriad

The Human Library Project

The Human Library was first started in Denmark in 2000. Here, real people are the books lent to readers! You can basically select a "human book" (experienced people who volunteer for this task). You are then given a slot where you can meet them in a safe environment—to listen to their experiences, discuss things and learn from them. This could be individual or as a small group. This project is successfully going on in several technologically-advanced countries, including Singapore.

If AI can answer all questions, why do we need this? It just proves that nothing can replace a human's experience, emotions, and knowledge.

(In fact, it was human experts who explained ChatGPT to us in this article, not ChatGPT itself!)

of applications, we must exercise caution while sharing our data with these emerging generative AI tools. Moderation is crucial in ensuring that AI-generated content is not only accurate and unbiased but also compliant with intellectual property laws. It is imperative that we recognise the significance of these issues and take steps to address them," he adds.

Will ChatGPT steal our jobs?

One of the major concerns with regards to ChatGPT is that it is going to take away a lot of jobs, ranging from customer service representatives to developers. However, many experts say this might not be the case.

"ChatGPT is without a doubt a technological marvel that can provide incredible assistance to its users. Its ability to analyse large datasets and generate responses that resemble human intelligence is truly remarkable. However, it is important to remember that ChatGPT is not a sentient being with independent thought, emotions, or consciousness. While it can provide insights and recommendations based on patterns and relationships discovered in massive amounts of data, it lacks personal biases and its own agenda. Its responses should be interpreted as suggestions or prompts rather than authoritative statements. ChatGPT, despite its impressive capabilities, cannot replace humans' distinct qualities and skills. It is critical to understand that the distinction between human and artificial intelligence must be based on their distinct capabilities and limitations. Creativity, empathy, and critical thinking are all essential components of human intelligence that cannot be replicated by a machine," asserts Sahay.

As in the case of industrial automation, it will only be a tool to enhance productivity of human workers, removing mundane and repetitive tasks from their plate, freeing up their bandwidth for tasks that require a human touch. This interaction between humans and AI will only help the AI improve—it is a symbiotic relationship.

"It seems that we are moving towards a more AI-assisted world where humans will take the outputs from AI, validate them, or get better outputs from AI by interacting with them in an iterative manner and then use it further. This human-in-the-loop approach would change many of the things that we do today in the world and would make us significantly more productive," says Joshi.

If you are still sceptical about whether generative AI will steal your job, consider shifting to agriculture, mining or manufacturing. According to *The Potentially Large Effects of Artificial Intelligence on Economic Growth*, a report recently released by Goldman Sachs, jobs in these three sectors are the least exposed to generative AI, while jobs that involve programming and writing skills are more closely related to GPT's capabilities! END

Speakers

Anurag Sahay, CTO and MD - AI and Data Science, Nagarro

Sachindra Joshi, IBM Distinguished Engineer, Conversational Platforms, IBM Research India

Sanjeev Azad, Vice President - Technology, GlobalLogic

Steve Grobman, CTO, McAfee

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.

The article was originally published in the May 2023 issue of Electronics For You.

A Quick Look at Deep Learning

A subset of machine learning, deep learning has been around since the middle of the last century, but started evolving rapidly since the early 21st century. Generative AI and ChatGPT have brought the focus back on it. We take a look at what deep learning is about and how it's being used.

Data management is one of the hardest and costliest tasks every organisation is faced with. The fact that organisations appoint a chief data officer (CDO) shows how much importance they give to managing data — its treatment, classification, analytics, security and compliance requirements, etc. This is where deep learning (DL) shines as a perfect data management strategy when employed with the right set of tools.

While DL has been around for a while, the recent popularity of ChatGPT brings it back to being a hot topic in the world of AI. Deep learning is a subset of machine learning (ML), and is based on how the human brain is structured and works. If you are new to the subject and looking to learn the differences between AI, ML and DL, check out my writing mentioned in the 'References' at the end of this article. DL uses artificial neural networks to learn from large

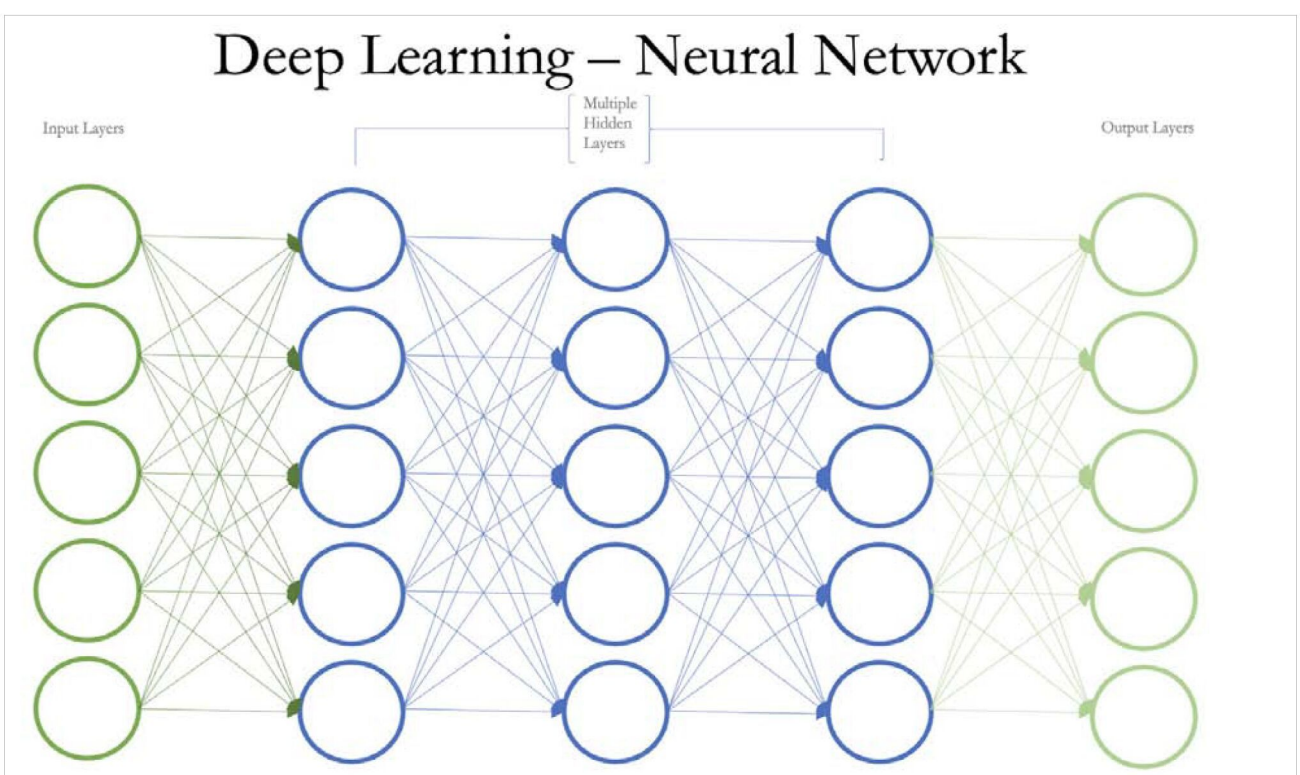


Figure 1: Deep learning - neural network

amounts of data. It gets the name 'deep' because it leverages multiple layers of interconnected neurons to process and learn from the data. DL algorithms are capable of learning complex patterns in images, audio and text. Deep learning uses large amounts of unlabelled data to learn the underlying structure and patterns in data. It is an unsupervised learning module that does not require labelled data to start learning from.

Deep learning has been around since the 1940s. However, it gained popularity in the 1980s with the development of backpropagation, a technique for training neural networks. In the 2000s, the evolution of advanced techniques with respect to dropout and rectified linear units (ReLU) helped address core challenges. In 2012, AlexNet, a convolutional neural network, focused on image recognition, won awards and regained popularity for mainstream uses in natural language processing (NLP), speech recognition and autonomous vehicles.

Some common use cases of deep learning are listed below.

- **Image recognition:** Classification of images leveraging DL algorithms is being done across healthcare institutions for the diagnosis of diseases.
- **Natural language processing (NLP):** NLP, when integrated with DL algorithms, opens up use cases like chatbots, machine translation and voice assistants.
- **Speech recognition:** DL algorithms help transcribe spoken language for use cases like speech-to-text transcription and voice assistants.
- **Fraud detection:** Deep learning algorithms help to keep a lookout for fraudulent transactions in banking and finance.
- **Autonomous vehicles:** DL algorithms help to navigate the surroundings in autonomous vehicles, enabling real-time decisions to be made while these cars are on the road.

Now, let's take a deeper look at what deep learning is all about! As the 'deep' in deep learning represents the depth of layers in neural networks, a neural network that consists of more than three layers is considered a deep learning algorithm. As shown in Figure 1, most deep learning networks are feed-forward models. But they can be trained to back-propagate as well. The latter is especially useful to provide a feedback on errors so that the algorithm can be corrected appropriately.

Various studies and experiments have been conducted to validate the accuracy of deep learning neural networks. A network with at least three hidden layers has produced more than 98 per cent accuracy in data classification. This figure goes up with proper adoption of backpropagation. This is really cool, given the model does not need any preparation or prerequisites to begin with.

Although deep learning is considered to be a subset of machine learning, it does have its own unique characteristics in the way how each algorithm learns and how much data each type of algorithm uses. The fact that DL automates elimination of human intervention in mining large unstructured data sets with no prerequisites makes it a very scalable and extensible practical solution for industrial applications.

Deep learning may seem like a low maintenance application, but the high computing power it needs to process large amounts of data, as well as the black box nature of hidden layers, poses significant challenges in its adoption for many. Also, the classification and meaning of the data it generates is a reflection of what's fed to it. Skilled data scientists are needed to understand the classifications, as well as refine and tune the model to look for specifics.

While DL algorithms are not perfect and much is yet to be researched and refined, just like in the other disciplines of AI, its technique has been recognised as powerful and good for many industrial use cases. The ability to learn without human intervention, and label large data sets of image, video and text, makes deep learning a game changing strategy for large enterprises where, typically, more than 80 per cent of data is unstructured. This labelling is an important step in making machine learning targeted and meaningful. END

References

- Neural Networks and Deep Learning; *http://neuralnetworksanddeeplearning.com/*
- Introduction to Deep Learning; *https://www.geeksforgeeks.org/introduction-deep-learning/*
- Deep Learning.AI: Start or Advance Your Career in AI
- Do more with less. Generative AI: The Next Big Thing; *https://www.deeplearning.ai/*; *https://www.opensourceforu.com/2023/05/generative-ai-the-next-big-thing/*
- AI, ML and DL: What's the Difference? *https://www.opensourceforu.com/2022/08/ai-ml-and-dl-whats-the-difference/*

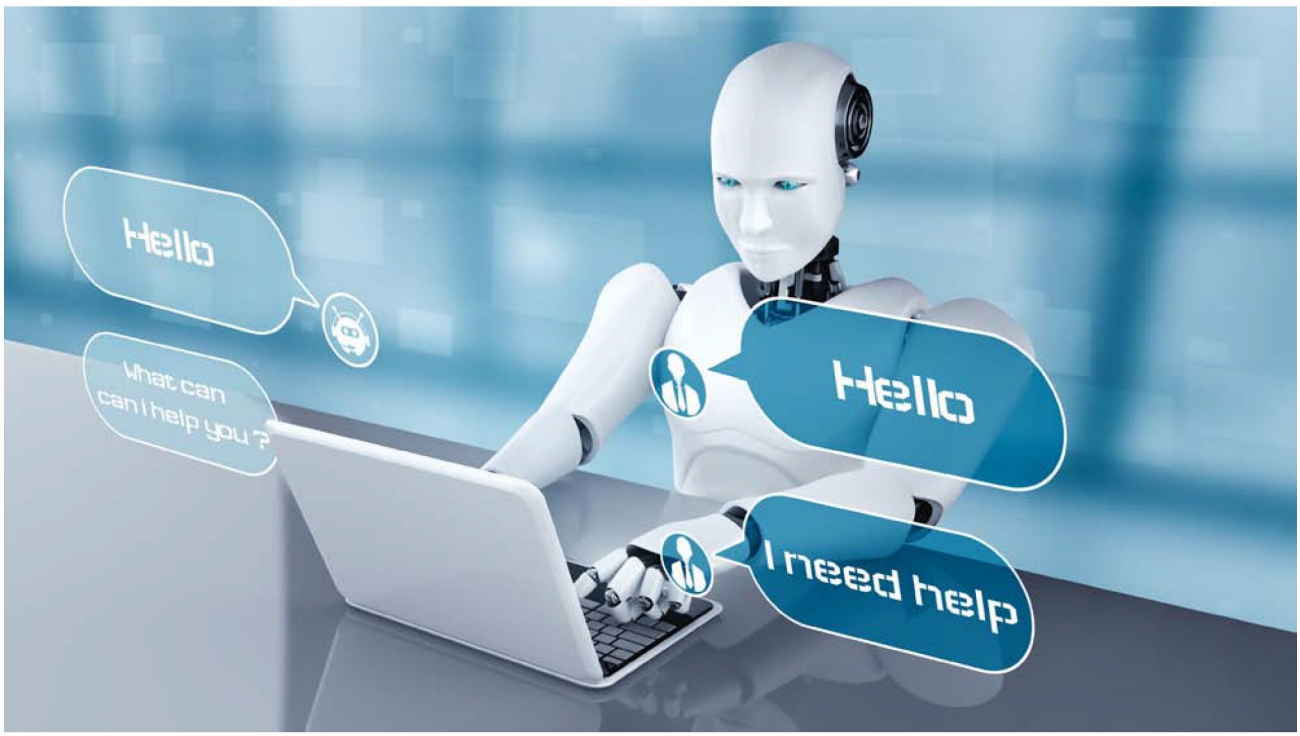
By: Bala Kalavala

The author is a technical architect, evangelist, thought leader, and sought-after keynote speaker. He currently works as a distinguished member of the technical staff and head of the Enterprise Architecture practice as chief architect in a global technology consulting firm.

Disclaimer : This article expresses the views of the author and not of the organisation he works in.

A Deep Dive into Generative AI and ChatGPT-3

ChatGPT has taken the world by storm. But there is a lot more happening in the world of AI. We take a look at that, and also peek into the strengths and a few shortcomings of ChatGPT-3.



Generative AI refers to the use of artificial intelligence algorithms to generate new data, text, images, or other content that is similar to existing data or content. It involves the use of machine learning techniques to create models that can generate new data based on patterns found in existing data.

OpenAI developed a Generative AI architecture called generative pretrained transformer (GPT), and ChatGPT-3 is an example of a generative AI model that is used for natural language processing tasks, such as chatbots, language translation, and text summarisation. It is a deep learning model that is capable of generating human-like text responses based on the input it receives.

The model is trained on a massive data set of text from the internet, which includes a wide range of topics and writing styles. This allows it to generate responses that are contextually relevant and stylistically appropriate for a given conversation. One of the key benefits of generative AI models like ChatGPT-3 is their ability to adapt and learn over time. As they are exposed to more data and feedback from users, they can refine their responses and become even more accurate and effective.

However, there are also concerns about the potential for generative AI models to generate biased or misleading content, particularly when they are used to create news articles or other forms of content that could impact public opinion.

Generative AI has the potential to revolutionise many industries and applications, including chatbots, content creation, and even scientific research. As technology continues to advance and data becomes more widely available, it is likely that we will see even more powerful and sophisticated generative AI models like ChatGPT-3 emerge in the coming years.

History and development of ChatGPT solutions

ChatGPT is a large language model developed by OpenAI that utilises the GPT architecture. It was officially released in June 2020 and has since become one of the most powerful natural language processing models in the world.

GPT-1 was the first model in the GPT series and was introduced by OpenAI in 2018. It was trained on a massive data set of text from the internet and was capable of generating coherent text based on a given prompt. However, its performance was limited by the relatively small size of its training data set and the lack of advanced natural language processing techniques.

In 2019, OpenAI released GPT-2, which was a significant improvement over GPT-1 in terms of performance and capabilities. GPT-2 was trained on a much larger data set of text, which included web pages, books, and even Wikipedia. It also utilised advanced natural language processing techniques, such as the Transformer architecture, which enabled it to generate more coherent and contextually relevant text.

However, due to concerns over the potential misuse of GPT-2 for malicious purposes, such as generating fake news or propaganda, OpenAI initially decided to withhold the full version of the model from the public. Instead, they released only a smaller, less powerful version of the model, known as GPT-2 117M.

In June 2020, OpenAI released ChatGPT, which is based on the same transformer architecture as GPT-2 but is specifically designed for conversational applications. ChatGPT is trained on a massive data set of text from the internet, including social media, online forums, and blogs. This enables it to generate human-like responses to a wide variety of prompts and questions.

One of the key innovations of ChatGPT is its ability to generate not just text, but also images, audio, and video. This is made possible by the use of a multimodal transformer architecture, which combines natural language processing with computer vision and audio processing techniques.

In addition to its technical capabilities, ChatGPT also represents a significant milestone in the history of artificial intelligence and natural language processing.

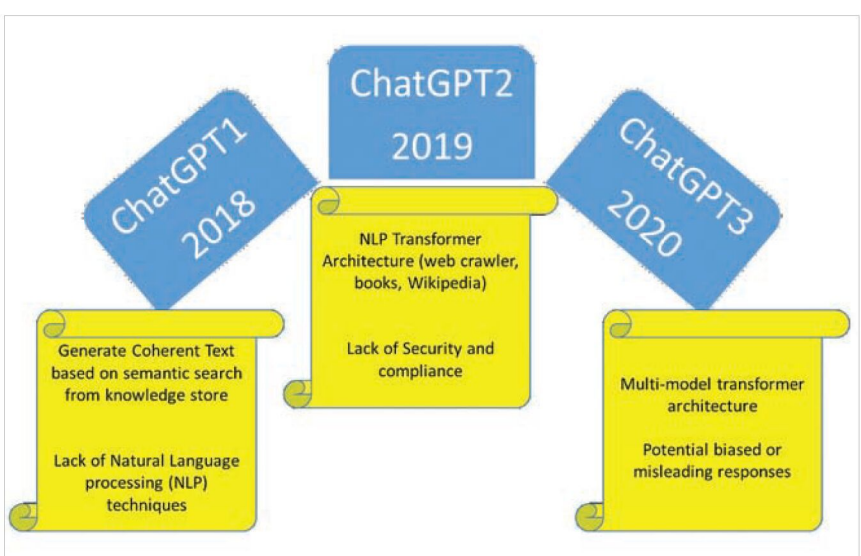


Figure 1: History and evolution of ChatGPT models

It has demonstrated the potential for machines to generate human-like text and engage in meaningful conversations with humans.

However, as with any powerful technology, there are also concerns about the potential misuse of ChatGPT. Its ability to generate convincing fake news, propaganda, or other types of malicious content could be used to manipulate public opinion or spread disinformation. The evolution of ChatGPT and its predecessors, GPT-1 and GPT-2, represents a significant milestone in the development of artificial intelligence

Reinforcement learning with human feedback

Reinforcement learning with human feedback (RLHF) is a powerful technique in the field of artificial intelligence (AI) that combines the strengths of both human and machine learning. In this approach, the machine learns from human feedback to refine its decision-making capabilities and improve its overall performance. RLHF has numerous applications across a wide range of domains, including robotics, gaming, finance, and healthcare.

At its core, RLHF is a type of machine learning algorithm that enables an agent to learn from its own experiences through trial and error. The agent is rewarded or penalised based on its actions in a given environment, with the goal of maximising its cumulative reward over time. The agent uses this feedback to adjust its behaviour and make better decisions in the future.

However, in many real-world applications, it can be difficult to define the rewards and penalties that the agent should receive. In these cases, RLHF allows humans to provide feedback on the agent's behaviour, helping to guide its learning and improve its performance. This feedback can take many forms, including explicit

rewards or penalties, natural language instructions, or even physiological signals such as facial expressions or brain waves.

One key advantage of RLHF is its ability to leverage the unique strengths of both humans and machines. Humans can provide rich, nuanced feedback that is difficult for machines to generate on their own, while machines can process vast amounts of data and make decisions at speeds far beyond what humans are capable of. By combining these two approaches, RLHF can accelerate the learning process and achieve higher levels of performance than either approach alone.

There are several technical challenges associated with RLHF, including how to integrate human feedback into the learning process, how to handle noisy or inconsistent feedback, and how to ensure that the agent does not become overly dependent on human guidance. One common approach is to use a hybrid reinforcement learning model that incorporates both human feedback and traditional RL techniques, such as Q-learning or policy gradient methods.

To implement RLHF, several tools and frameworks are available, such as TensorFlow, Keras, PyTorch, and OpenAI Gym. These frameworks provide a rich set of APIs and libraries for building RLHF models, training and evaluating them, and deploying them in real-world applications.

RLHF has several applications in different domains. In robotics, it is used to train robots to perform complex tasks such as grasping and manipulation of objects, navigation, and control. In gaming, it is used to develop more intelligent and responsive game agents that can learn from player behaviour and adapt to changing game environments. In finance, RLHF is used for portfolio optimisation, fraud detection, and risk management. In healthcare, RLHF is used for diagnosis and treatment of diseases, prediction of patient outcomes, and drug discovery.

RLHF is a powerful technique that combines the strengths of human and machine learning to achieve higher levels of performance in a wide range of applications. With the continued development of new tools and frameworks, RLHF is poised to become an increasingly important approach to AI in the years ahead.

Architecture of ChatGPT-3

ChatGPT-3 has a complex architecture that involves several layers of artificial neural networks. It is based on the Transformer architecture, which was introduced in a 2017 paper by Vaswani *et al*. This architecture is a type of neural network that is designed to process sequential data, such as natural language text. The main components of the ChatGPT-3 architecture include:

Input embedding: The input text is converted into a sequence of vectors that can be processed by the neural network. These vectors are called embedding.

Transformer encoder: The embeddings are fed into a stack of transformer encoder layers, which process the input text to create a sequence of hidden representations.

Transformer decoder: The decoder takes the hidden representations generated by the encoder and uses them to generate an output sequence of vectors.

Output embedding: The output sequence is then converted back into text using output embedding.

The number of encoder and decoder layers in the ChatGPT-3 architecture varies depending on the size of the model. The largest version of the model, with 175 billion parameters, has 96 encoder and 96 decoder layers.

Overall, the architecture of ChatGPT-3 is highly complex and requires a significant amount of computational resources to train and run. However, this complexity allows the model to generate highly realistic and coherent text, making it a powerful tool for natural language processing tasks.

Generative AI and RLHF

Generative AI, reinforcement learning with human feedback (RLHF), and generative adversarial networks (GANs) are all related to the field of AI. Generative AI is a subset of AI that focuses on creating new data, such as images, videos, and text, using machine learning algorithms. These algorithms can create original content by learning patterns and structures from existing data.

RLHF is a subfield of reinforcement learning that uses human feedback to train AI models. RLHF is a more interactive form of machine learning, where the AI model is given feedback from human experts to improve its performance.

GANs are a type of Generative AI that uses two neural networks to create new data. The first network generates new data, while the second network evaluates the generated data to ensure it is realistic.

ChatGPT-3 is a state-of-the-art generative AI model that uses a large neural network to generate text based on the input it receives. It can create responses to questions, write essays, and even generate code. The model was trained on a massive amount of data, and has been praised for its ability to generate coherent and useful text.

RLHF and GANs can be used to improve the performance of ChatGPT-3. RLHF can provide feedback to the model based on its generated text, allowing it to improve its responses over time. GANs can generate more realistic and diverse text, further enhancing the model's capabilities.

Generative AI, RLHF, GANs, and ChatGPT-3 can be used together to create more advanced and sophisticated AI models.

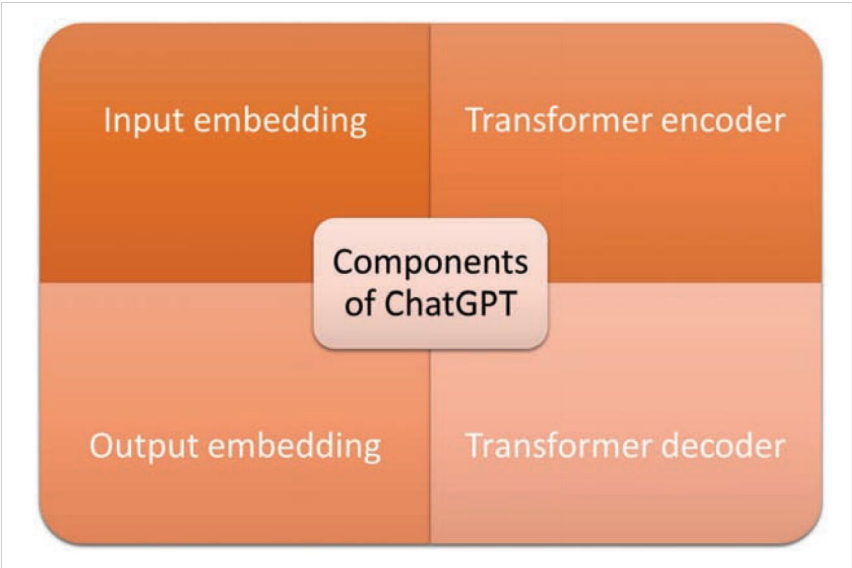


Figure 2: Components of ChatGPT architecture

Risks in Generative AI and ChatGPT3

Disruption risk: AI is evolving each day and now has enough potential to disrupt existing business models and diverse markets like no technology has ever done before. Jobs as varied as trucking and customer care to high-end ones like financial trading analysts, or medical jobs like radiologists, etc, are at risk. As per a March 2023 report from Goldman Sachs, as many as 300 million jobs will be eliminated by ChatGPT-like AI including 19 per cent of the existing job market. So, in the next four years, the job market will have a different face with absolutely new skills requirements.

Cyber security risk: There was a massive 38 per cent increase in cyber attacks in 2022 and this figure may even increase to 44-48 per cent by the end of 2024. AI can increase this risk of cyber security, especially when sending phishing emails. Constant evolution of deepfake and voice clone technologies is increasing these risks each day.

Reputation risk: After the launch of ChatGPT, Google and other top IT giants entered the fray to make their own AI products like Google BardAI. However, sometimes, AI doesn't give the expected results, which puts the reputation of IT companies at stake. As per recent reports by Forrester, 75 per cent of customers face issues in real-time interaction with chatbots and 30 per cent of businesses have migrated to some other technology rather than an AI-driven solution, as it is still at a very young stage and prone to errors.

Risk of legal consequences: Governments of many countries are trying to make proper laws to tackle the diverse issues of AI. In 2023, the US-based National Institute of Standards and Technology (NIST) released an 'AI Risk Management' framework to help corporate and other leaders handle risks that originate from AI. In addition, EU has proposed an AI act to ban use of software like biometrics recognition, deepfakes, and more. Many more legal regulations are expected in the coming few months to safeguard normal lives and the functioning of governments.

Operational risk: The most debatable risk of ChatGPT and Generative AI is the operational risk. Recently, tech giant Samsung banned its employees from using ChatGPT after some trade secrets were leaked.

ChatGPT is a good example of advanced AI but it is still in its testing phase, and many shortcomings are being reported by a large number of people.

Software development using ChatGPT-3

ChatGPT is an extremely powerful tool that has the potential to revolutionise almost anything. To leverage its power, we need to integrate it into normal workflows and one of the best ways to do this is software development.

With proper understanding, ChatGPT can be used to improve the software development life cycle. Developers can use its power to improve coding skills and develop error-free projects, thus cutting down the time of delivery of the final product to customers.

In order to make effective use of ChatGPT, developers can make use of libraries like OpenAI's API, Hugging Face's Transformers or spaCY.

ChatGPT can revolutionise software development in the following ways.

Code completion and optimisation: Code completion is one of the most fantastic applications of ChatGPT in software development. It can suggest code or develop complete code just by typing the requirement and language name. Developers can use it for:

a. *AutoComplete*: With this, ChatGPT completes the line of code with function definition, with integration of arguments and return types.
b. *Predictive code*: It can predict what a developer is expected to write. Taking the base idea, the entire code is generated by the system.
c. *Translation:* Developers can convert the code from one language to another with just a simple click.
d. *Optimisation:* If there is any error in the code, ChatGPT finds it, gives suggestions, and automatically fixes the code.

- **Example prompt**: "Given the code snippet, generate the next line of code that completes the functionality."

Continued to page...71

Building a Cross-Platform Mobile Application with Flutter

Building a mobile application is a necessity for every business and choosing the right tech stack for building the app is always a challenge. With so many options available in the market, why choose Flutter? Let's find out.

Launched in May 2017, Flutter is a free, open source, and cross-platform UI SDK created by Google to develop an application for Android and iOS, as well as web applications.

One of the first reasons to consider using Flutter is that you will end up with one code base to maintain, one place to debug, and one place to update your application. This is a huge advantage compared to creating your apps natively, say, building your iOS apps in Swift and your Android apps in Java or Kotlin. To update any feature in your application, you need to make changes at numerous places within it, which is both difficult and error-prone. If you use Flutter, you only need to know one language — Dart, which is a powerful language and easy to work with. Once you have learnt and understood how to use it, you can create your iOS, Android, and web apps. You have to learn only one language instead of many.

If you have done any kind of programming before, then you will realise that Dart is actually very similar to a lot of modern object-oriented programming languages. It has been used at Google to build powerful tools such as Google AdWords and Google Fiber. I bet its usage is only going to get bigger in the future.

Anatomy of a Flutter app

Everything in the Flutter app is a widget. Just like Lego blocks, you put a widget upon another widget to build the application. To develop an app using Flutter, the first thing to do is to create a scaffold, i.e., a blank screen for our app. Inside this scaffold, we need to add an app bar at the top and a container below that. These are pre-built widgets. The app bar looks like the toolbar/app bar and acts like one. The container is nothing but a box that will contain the contents of the application. This will look somewhat like the Android UI design — scaffold as your parent layout, app bar as a toolbar, and container just

like the fragment/nested layout.

Let's build it further such that the container has a column. Like any other widget, the column can be nested too. It stacks the items/widgets vertically. Let's say this column has two items: a row at the top and some text at the bottom. What is a row, you may ask? This is nothing but a widget — like I said before, everything is a widget. Row arranges the widgets horizontally. We can go deeper into the widget tree by adding some text and an icon in the row. What you need to remember is that when you want widgets to be positioned vertically (one on top of the other), use the column to lay them out, and when you want widgets to be positioned horizontally, use a row. To add text, use a text widget, and if you want to add an icon, use an icon widget. For images, use the image widget. Pretty simple, isn't it? Your widget tree will look something like what's shown in Figure 1 by the time you build the app.

Our widget tree is just a whole bunch of widgets that are nested within each other. Using these widgets, you can create a beautiful yet interactive UI. Now let's set up/download the tools to get started with app development.

Prerequisites for Flutter app development

Development environment setup: Linux, Mac or Windows based computer

IDE: Android Studio/Visual Studio Code

Disk space: Minimum 10GB free space. Although Flutter requires only 1.64GB, you will need space to install Android Studio

Tools: Git

> **Note:** For this article I will be covering only Android application development using a Windows system and Android Studio.

Before you begin, make sure you have a version of Java 11 installed and that the JAVA_HOME environment

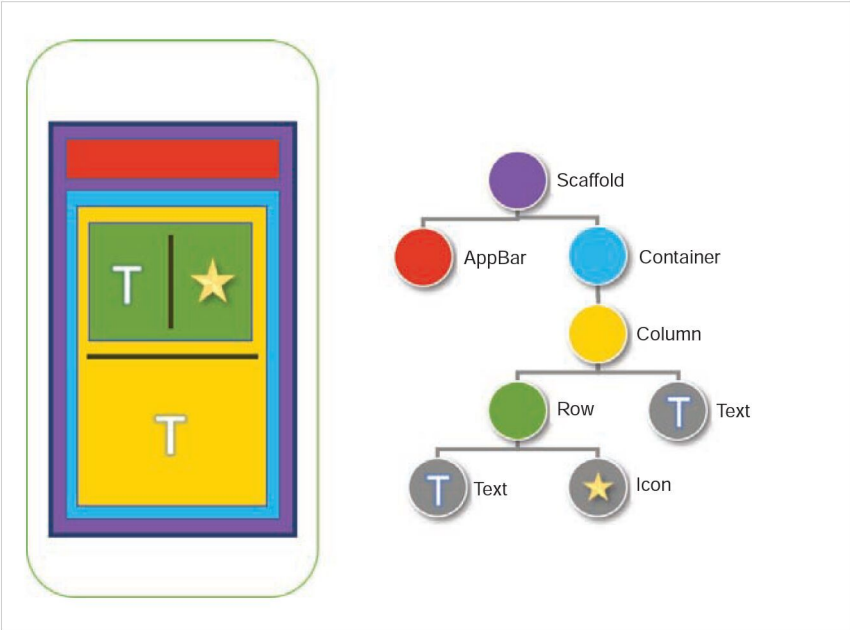


Figure 1: Anatomy of a Flutter app

variable is set to the JDK's folder. If you have installed Android Studio version 2.2 and higher, this should already be the case as it comes with a JDK.

There are three steps for completing the installation process:

- Install the Flutter SDK
- Install Android Studio
- Install Android Emulator

Installing the Flutter SDK

Download the Flutter SDK for Windows (*https://docs.flutter.dev/get-started/install/windows*) and follow the installation instructions. Once the installation is over, update the environment variable as mentioned in the installation guide. Let's verify that we have installed it correctly by running the following command:

```
Flutter –version
C:\Users\verma_ru>flutter --version
Flutter 3.7.6 . channel stable •
https://github.com/flutter/flutter.git
Framework revision 12cb4eb7a0 (7 days
ago) • 2023-03-01 10:29:26 -0800
Engine revision ada363ee93
Tools • Dart 2.19.3 DevTools 2.20.1
```

This will give you the installed version of Flutter and Dart. Now, run another command to check the environment details:

```
Flutter doctor
```

This will display a status report of the installation and the other software that we may need to install.

```
C:\Users\verma_ru>flutter doctor
Doctor summary (to see all details, run
flutter doctor -v):
[√] Flutter (Channel stable, 3.7.6,
on Microsoft Windows [Version
10.0.17763.557], locale en-IN)
[√] Windows Version (Installed version
of Windows is version 10 or higher)
[√] Android toolchain develop for
Android devices (Android SDK version
33.0.1)
[√] Chrome - develop for the web
[√] Visual Studio
develop for Windows (Visual Studio
Professional 2019 16.11.11)
[√] Android Studio (version 2020.3)
[√] Android Studio (version 2021.3)
[√] Connected device (2 available)
[√] HTTP Host Availability
• No issues found!
```

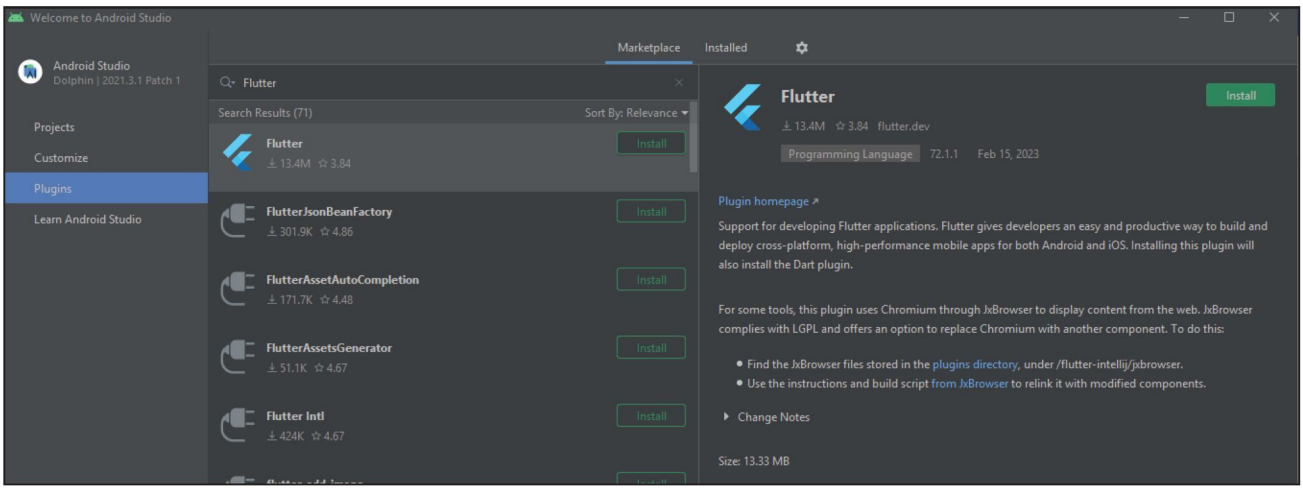


Figure 2: Flutter installation in Android Studio

Installing Android Studio

If you don't have Android Studio installed, download and follow the instructions mentioned in the Flutter installation link. Once you have installed Android Studio, be sure to go through the Android Studio setup wizard. This installs the latest Android SDK, Android SDK platform tools, and Android SDK build tools which are going to be required by Flutter when it's building the Android app.

Next, open the Android Studio. It should display a *Welcome* screen. We will configure Flutter and Dart in Android Studio by browsing for the Flutter and Dart repositories. Click on the *Plugin* menu, search for *Flutter*, and click on *Install*. It's also going to install the Dart plugin together with this (Figure 2).

Once this is done, restart Android Studio. You will notice a new menu item on the Welcome screen – 'New Flutter Project'. If you see this, the installation has been successful.

Installing Android Emulator

To run your application, we must install an Android Emulator. You can choose to run the application on a physical device as well. But first, we must create a Flutter app.

Click on 'New Flutter Project' on the *Welcome* screen to start, and click on the *Flutter* menu. Make sure that the Flutter SDK path is listed here. Now click on the *Next* button. Refer to Figure 3.

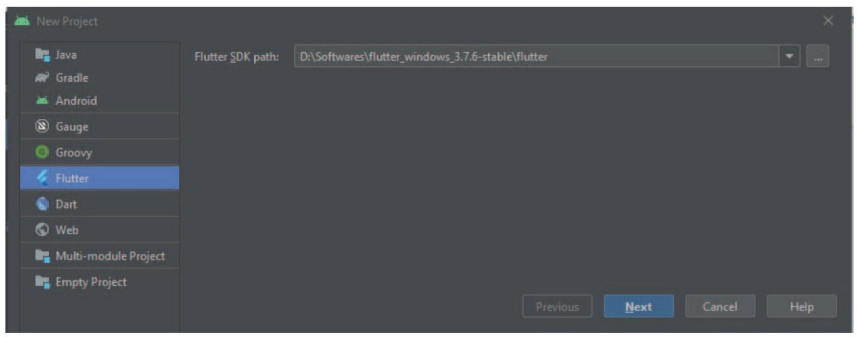


Figure 3: New Flutter Project 1

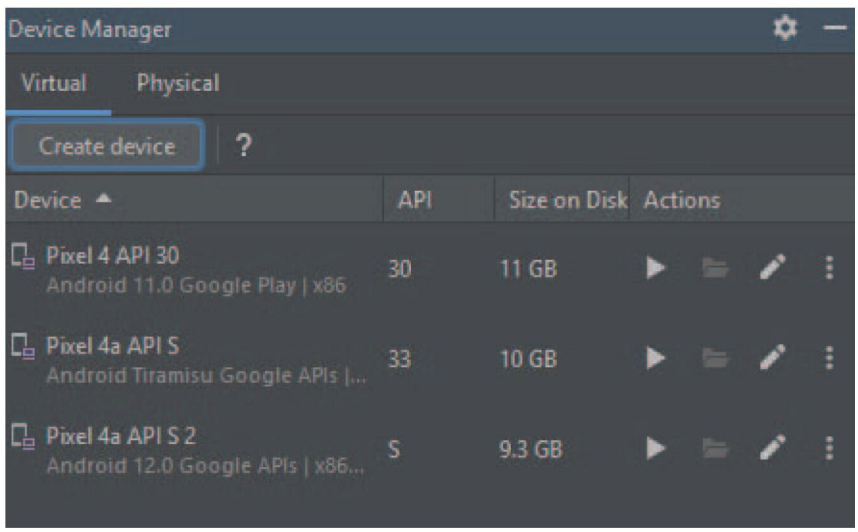


Figure 4: Android Device Manager

Next, enter the project name as HelloFlutter and select the project location. Select project type as *Application*. Additionally, select the Android and iOS development language of your choice. You do not need to learn or know the

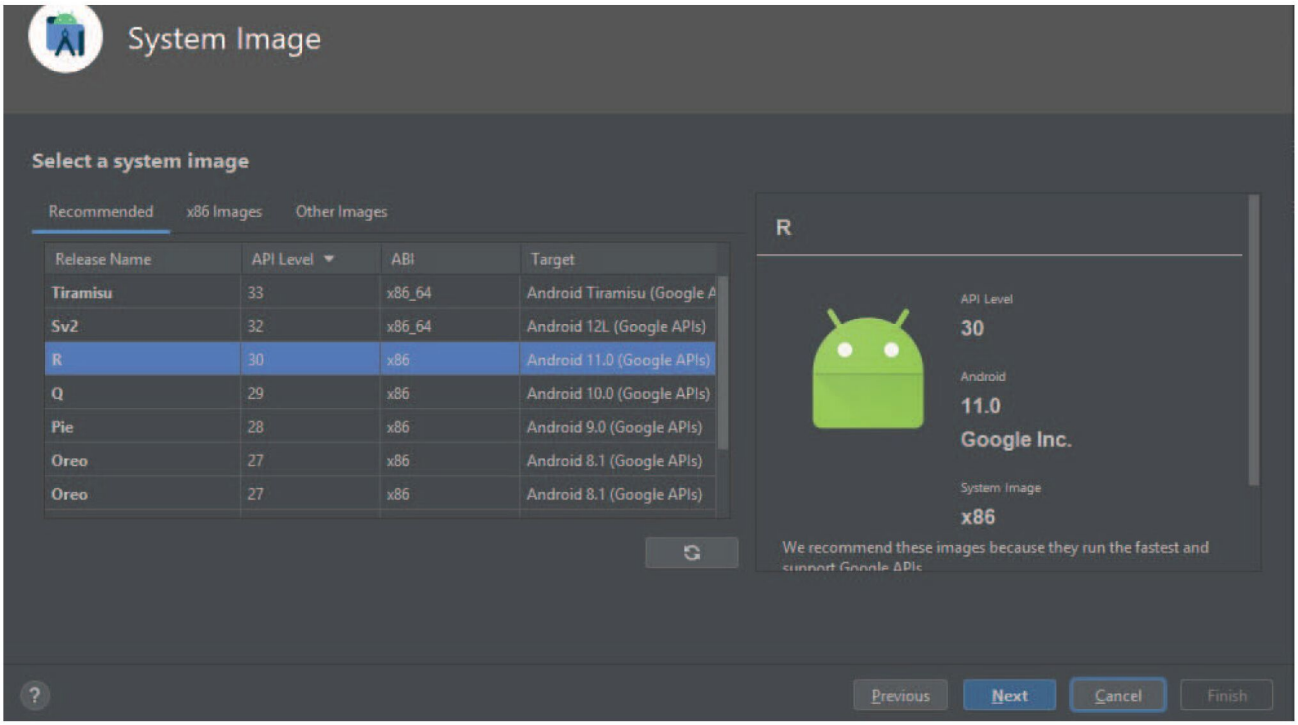


Figure 5: Choosing emulator OS version

language at this moment. Select the platforms for which you are building the application. Once done, click on the *Finish* button to build your first Flutter app. In case the IDE is displaying a popup for creating a new directory, click on the *Create* button.

Now we will set up the Android Virtual Device (Emulator) to run our application. To get the Android Emulator, click on the *Device Manager* (*Tools -> Device Manager*).

You can find the Device Manager in the toolbar too.

This will open a device manager window as shown in Figure 4, where you can create a new virtual device or start an existing one.

Let's click on the *Create Device* button to create a new one. I would recommend using the Pixel or Nexus emulator from the list.

Once you've selected the device, click on *Next*. Now, you get to specify which operating system you want to run on that device. Here I recommend going for Android R. If you haven't installed that OS yet, do so. This might take anywhere between 5 to 10 minutes. Once done, click on the *Next* button.

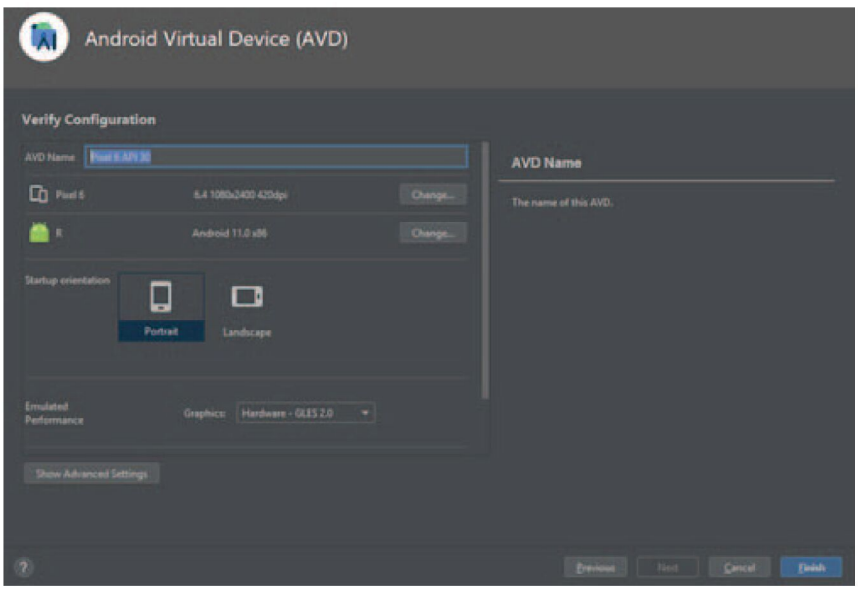


Figure 6: Changing emulator configuration

Now, we get to the option to change the virtual device's name. The most important thing here is to choose hardware for the graphics (hardware – GLES 2.0). This way, we'll be able to use the computer's graphics card for faster rendering, which will vastly speed up the Android Emulator. Now we're all done, and we can click on *Finish*.

You will see this newly created emulator in the Device Manager window listed under Virtual Devices now.

In order to launch it, all you have to do is click on the *Play* button. Alternatively, you can click on the

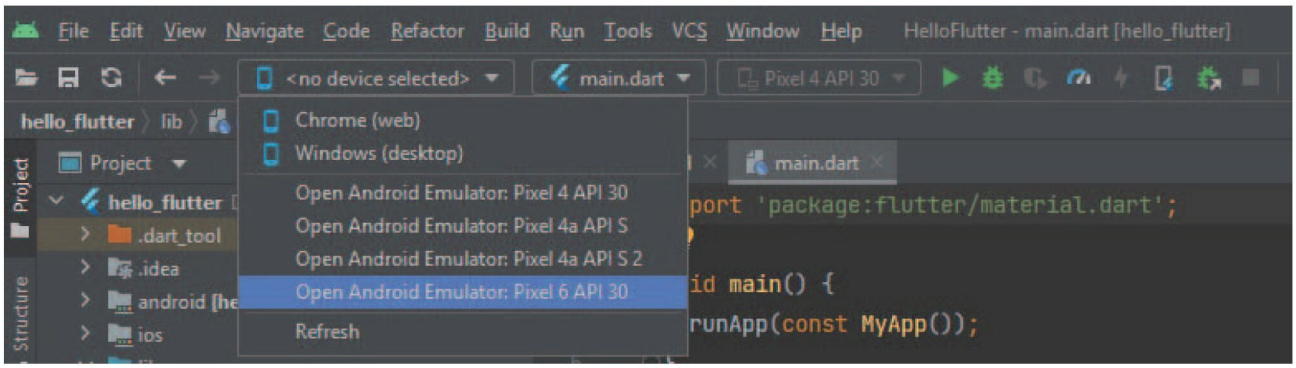


Figure 7: Flutter device selection

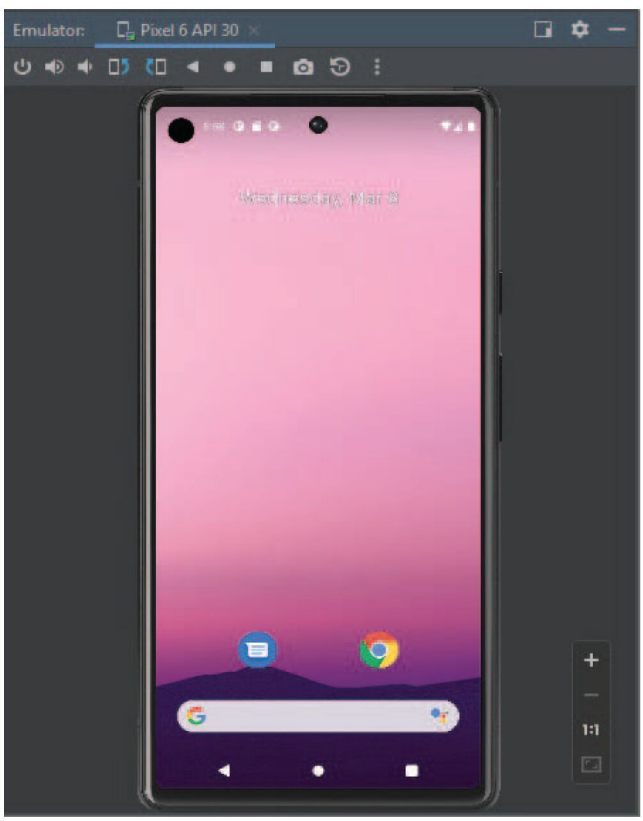


Figure 8: Pixel 6 emulator

Figure 9: Hello Flutter app

Device Selection drop-down in the toolbar to launch the device.

Once it's done launching, we have a brand new Pixel device as seen in Figure 8.

Now let's run the Flutter application created by the Android Studio by clicking on the *Play* button available in the toolbar. But before doing that, make sure you have selected the correct virtual device.

Congratulations!! You have built your very first Flutter app and all it does is keep count of how many times you push this button.

This also demonstrates that we've actually got our Android Studio, Flutter SDK, and Android Emulator installed, set up, and running.

Now that we have everything set up and running, let's build our Hello Flutter app. For that, you can create a new Flutter project, or you can update the earlier created application. I'm using the existing app for this article. Let's open the *main.dart* file and delete everything except the first few lines. Your file should look like what's given below:

```
import 'package: flutter/material.dart';

void main() {
runApp (const MyApp());
}
```

So now we get a little error here, and you see it in Dart Analysis as

well, where it says 'error: The name 'MyApp' isn't a class.'. Well, that's because MyApp was the Flutter team's app, and it doesn't exist anymore because I just deleted it.

Let's create a blank MaterialApp. Material design is a design style or a concept created by Google. Now your code should look like this:

```
void main() {
  runApp(const MaterialApp());
}
```

Let's go ahead and build the widget tree. The most important thing is to set the home inside the MaterialApp. Let's set it to a text widget that says 'Hello Flutter'. If you run the application at this moment, you'll notice that it displays 'Hello Flutter' on top of the screen. Let's align it to the centre of the screen with the help of the centre widget.

The centre widget will be our home now, and it will take the text widget as a child.

```
void main() {
  runApp(const MaterialApp(
    home: Center(
      child: Text("Hello Flutter"),
    ),
  ));
}
```

Let's run the application and see the result, as shown in Figure 9.

That's great. Isn't it? We have built our Hello Flutter app from scratch. I recommend reading more about the different types of widgets. You can create a beautiful-looking app with only one codebase using Flutter and Dart. Happy learning!! END

References

- *https://docs.flutter.dev/get-started/install*
- *https://docs.flutter.dev/development/ui/widgets-intro*
- *https://docs.flutter.dev/development/ui/material*
- *https://docs.flutter.dev/development/ui/layout*

By: Ruby Verma

The author follows the motto 'Less talk more code', and likes to learn and explore new things. She has more than 12 years of experience in software development.

Continued from page...65

Software documentation: This is quite a long and tedious process. But ChatGPT can write simple, clear, concise and audience-specific documentation with proper guidance

- **Example prompt**: "Given a set of requirements, generate a technical specification document that outlines the software solution and its components."

***Bug detection and automated testing*:** ChatGPT can improve code quality considerably, and also carry out strong automated testing.

a. *Bug detection:* ChatGPT can identify bugs by simply analysing the code base and flagging potential issues at various lines in the code.

b. *Automated testing:* Testing is time-consuming and a never-ending task. With ChatGPT, developers can get help for generating test case designs and test cases for all possible scenarios.

- **Example prompts**: "Given the expected output and input parameters, detect any differences in the actual output and provide possible reasons for the error."
- "Given the code block, generate an automated test that checks for expected output and provides feedback for any unexpected behaviour."

Natural language processing: ChatGPT is known for natural language processing. It analyses all test inputs by users and takes appropriate action. It has the potential to improve NLP accuracy, generate accurate responses, and carry out text classification and language translation.

- **Example prompts**: "Given a product review, determine the overall sentiment of the review as positive, negative, or neutral."
- "Generate a list of possible responses to a customer inquiry based on previous interactions." END

References

- *https://www.forrester.com/blogs/watch-out-for-turingbots-a-new-generation-of-software-development/*
- *https://www.scalablepath.com/data-science/chatgpt-architecture-explained*
- *https://www.thoughtspot.com/data-trends/ai/what-is-transformer-architecture-chatgpt*
- *https://flashpoint.io/blog/potential-risks-chatgpt-ai/*
- *https://www.infoworld.com/article/3689172/chatgpt-and-software-development.html*

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.

R Series: 'dplyr' Package

In this twenty-second article in the R, Statistics and Machine Learning series, we will continue on the exploration of handling text using the 'dplyr' package. This package is a grammar for data manipulation, and provides simple and fast functions to handle data frame-like objects.

We will use R version 4.2.2 installed on Parabola GNU/Linux-libre (x86-64) for the code snippets.

```
$ R --version
R version 4.2.2 (2022-10-31) -- "Innocent and Trusting"
Copyright (C) 2022 The R Foundation for Statistical Computing
Platform: x86_64-pc-linux-gnu (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under the terms of the
GNU General Public License versions 2 or 3.
For more information about these matters see
https://www.gnu.org/licenses/.
```

You can install and load the 'dplyr' package using the following commands:

```
> install.packages("dplyr")
Installing package into '/home/shakthi/R/x86_64-pc-linux-gnu-
library/4.1'
(as 'lib' is unspecified)
--- Please select a CRAN mirror for use in this session ---
...
 * copying figures
 * building package indices
 * installing vignettes
 * testing if installed package can be loaded from temporary
location
 * checking absolute paths in shared objects and dynamic
libraries
 * testing if installed package can be loaded from final
location
 * testing if installed package keeps a record of temporary
installation path
 * DONE (dplyr)

> library(dplyr)
Attaching package: 'dplyr'
```

Consider the 'Bank Marketing Data Set' available from the UCI Machine Learning Repository available at *https://archive.ics.uci.edu/ml/datasets/Bank+Marketing*. The data set is from a Portuguese banking institution and is available freely for public research use. There are four data sets available, and we will use the *read.csv()* function to import the data from a *bank.csv* file into a data frame, as shown below:

```
> bank <- read.csv(file="bank.csv", sep=";")

> bank[1:3,]
  age        job marital education default balance housing
loan  contact day
1  30 unemployed married   primary      no    1787      no
no cellular  19
2  33   services married secondary      no    4789     yes
yes cellular  11
3  35 management  single  tertiary      no    1350     yes
no cellular  16
  month duration campaign pdays previous poutcome  y
1   oct       79        1    -1        0  unknown no
2   may      220        1   339        4  failure no
3   apr      185        1   330        1  failure no
```

select

You can use the *select()* function to choose specific columns from the data frame. In the following example, we select the 'age', 'job' and 'description' fields from the data set:

```
> bank %>% select(age, job, education)
    age           job education
1    30    unemployed   primary
2    33      services secondary
3    35    management  tertiary
4    30    management  tertiary
5    59   blue-collar secondary
6    35    management  tertiary.
```

filter

The *filter()* function is used to produce a subset of the data that matches the input conditions. The bank entries with 'blue-collar' jobs alone can be listed as follows:

```
> bank %>% filter(job == "blue-collar")
   age         job  marital education default balance
housing loan   contact
1  59 blue-collar  married secondary   no  0  yes no  unknown
2  31 blue-collar married secondary no  360  yes yes cellular
3  25 blue-collar  single primary  no  -221  yes no   unknown
4  55 blue-collar  married  primary  no  627 yes no unknown
5  32 blue-collar  married secondary no 2089 yes no cellular
...
```

arrange

The data from selected columns can be sorted using the *arrange()* function. In the following example, the bank data is first sorted by 'age' and then by the 'balance' column:

```
> bank %>% arrange(age, balance)
    age  job  marital education default balance housing loan
1    19  student   single   unknown  no     0      no   no
2    19  student   single   primary  no     103      no   no
3    19  student   single secondary  no     302      no   no
4    19  student   single   unknown  no    1169      no   no
5    20  student   single secondary  no     291      no   no
6    20  student   single secondary  no     502      no   no
7    20  student   single secondary  no    1191      no   no
8    21  student   single secondary  no       6      no   no
```

relocate

The *relocate()* function is used to change the column positions in the output. You can use the '.before' and '.after' arguments to specify the location of the columns. For example:

```
> bank %>% relocate(age, .after = "job")
    job age  marital education default balance housing loan
1   unemployed  30  married   primary   no    1787   no   no
2   services  33  married secondary     no    4789  yes  yes
3   management  35   single  tertiary   no    1350  yes   no
4   management  30  married  tertiary no   1476     yes  yes
5   blue-collar  59  married secondary  no    0     yes   no
6   management  35   single  tertiary   no   747      no   no
...
```

count

The unique values for a column can be computed using the *count()* function, as shown below:

```
> bank %>% count(age)
   age   n
1   19   4
2   20   3
3   21   7
4   22   9
5   23  20
6   24  24
7   25  44
8   26  77
9   27  94
```

tally

The *tally()* method is a low-level function that can also be used to count unique values for a column. The total number of bank entries in the CSV file is 4521, as indicated below:

```
> bank %>% tally()
     n
1 4521
```

distinct

The unique rows in a data frame can be obtained using the *distinct()* function. The three possible values for marital status are shown below:

```
> bank %>% distinct(marital)
   marital
1  married
2   single
3 divorced
```

mutate

The *mutate()* function is used to create new columns from existing data. The balance field is in Euros, and a new USD column is computed based on the Euro-USD conversion rate as follows:

```
> bank %>% select(age, education, balance) %>% mutate(usd =
balance * 1.08)
     age education balance      usd
1     30   primary    1787  1929.96
2     33 secondary    4789  5172.12
3     35  tertiary    1350  1458.00
4     30  tertiary    1476  1594.08
```

```
5     59 secondary       0     0.00
6     35  tertiary     747   806.76
   ...
```

pull

The *pull()* function accepts a numeric argument for the column number and returns its values. The last column in the data set is indexed at '-1' and corresponds to whether the client had subscribed to a term deposit.

```
> bank %>% pull(-1)
   [1] "no"  "no"  "no"  "no"  "no"  "no"  "no"  "no"  "no"
"no"  "no"  "no"
  [13] "no"  "yes" "no"  "no"  "no"  "no"  "no"  "no"  "no"
"no"  "no"  "no"
  [25] "no"  "no"  "no"  "no"  "no"  "no"  "yes" "no"  "no"
"yes" "yes" "no"
  [37] "yes" "yes" "yes" "no"  "no"  "no"  "no"  "no"  "no"
"no"  "no"  "no"
```

group_by

The data set can be categorised using the *group_by()* function, as demonstrated below:

```
> bank %>% group_by(age)
# A tibble: 4,521 × 17
# Groups:   age [67]
     age job       marital education default balance housing
loan  contact   day
   <int> <chr>     <chr>   <chr>     <chr>     <int> <chr>
<chr> <chr>   <int>
 1    30 unemploy… married primary   no         1787 no
no    cellul…    19
 2    33 services  married secondary no         4789 yes
yes   cellul…    11
 3    35 manageme… single  tertiary  no         1350 yes
no    cellul…    16
 4    30 manageme… married tertiary  no         1476 yes
yes   unknown     3
 5    59 blue-col… married secondary no            0 yes
no    unknown     5
 6    35 manageme… single  tertiary  no          747 no
no    cellul…    23
 7    36 self-emp… married tertiary  no          307 yes
no    cellul…    14
 8    39 technici… married secondary no          147 yes
no    cellul…     6
 9    41 entrepre… married tertiary  no          221 yes
no    unknown    14
10    43 services  married primary   no          -88 yes
yes   cellul…    17
# i 4,511 more rows
...
```

summarise

You can create a new data frame by also grouping variables using the *summarise()* function. The *summarize()* name can also be used instead.

```
> bank %>% summarise(age, balance)
     age balance
1     30    1787
2     33    4789
3     35    1350
4     30    1476
5     59       0
```

glimpse

The entire data frame can be transposed using the *glimpse()* function, which shows every column as a row. It tries to show as much of the data as possible in the output.

```
> bank %>% glimpse()
Rows: 4,521
Columns: 17
$ age       <int> 30, 33, 35, 30, 59, 35, 36, 39, 41, 43, 39,
43, 36, 20, 31, …
$ job       <chr> "unemployed", "services", "management",
"management", "blue-…
$ marital   <chr> "married", "married", "single", "married",
"married", "singl…
$ education <chr> "primary", "secondary", "tertiary",
"tertiary", "secondary",…
$ default   <chr> "no", "no", "no", "no", "no", "no", "no",
"no", "no", "no", …
$ balance   <int> 1787, 4789, 1350, 1476, 0, 747, 307, 147,
221, -88, 9374, 26…
```

slice

You can select the first three rows by specifying them with the *slice()* function. It allows you to select, filter and show duplicate rows as well. For example:

```
> bank %>% slice(1:3)
  age        job marital education default balance housing
loan  contact day
1  30 unemployed married   primary      no    1787      no
no cellular  19
2  33   services married secondary      no    4789     yes
yes cellular  11
3  35 management  single  tertiary      no    1350     yes
no cellular  16
  month duration campaign pdays previous poutcome  y
1   oct       79        1    -1        0  unknown no
2   may      220        1   339        4  failure no
3   apr      185        1   330        1  failure no
```

Continued to page...78

Building a Private Cloud Using OpenStack

OpenStack is a cloud computing platform that offers a suite of software tools for creating and maintaining both public and private clouds. It gives users the freedom to set up and control an unlimited number of virtual computers, networks, and storage devices. This short tutorial will show us how to set up a private cloud using OpenStack and launch a virtual machine over the deployed environment.

OpenStack architecture contains several components like controller nodes, compute nodes, neutrons, block storage, and object storage. Here's a brief description of these components (Figure 1).

Controller node: This node is responsible for providing most of the OpenStack services such as identity, image store, dashboard, etc.

Compute node (Nova): This is responsible for managing and automating compute instances (VMs) in the OpenStack environment.

Networking (Neutron): This enables networking connectivity and IP addresses for users.

Block storage node (Cinder): This node is used to provide persistent storage for the instances running at OpenStack.

Object storage (Swift): This is used for ensuring redundant, scalable, and fault-tolerant storage in the server cluster.

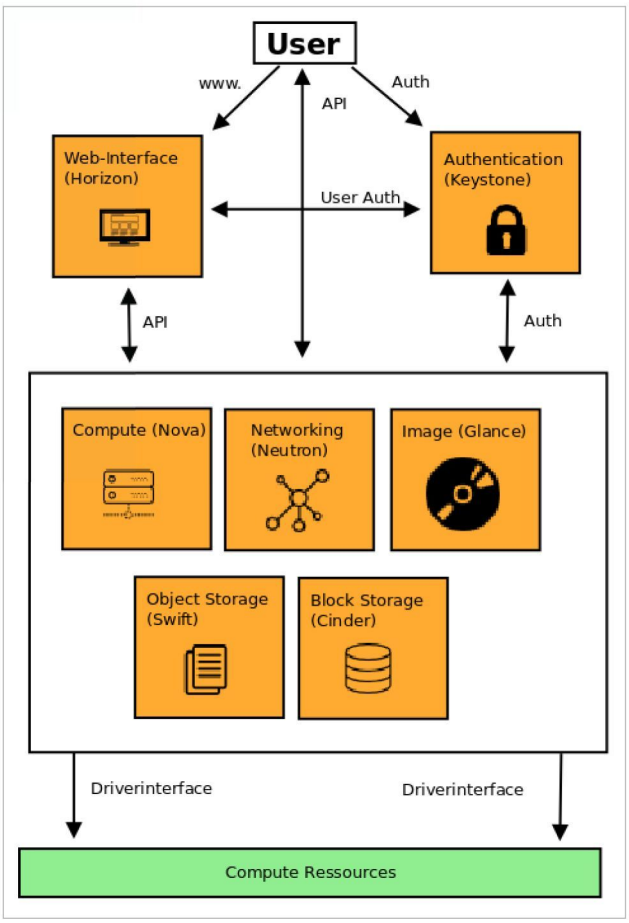


Figure 1: Key components in the OpenStack architecture

Minimum hardware requirements to set up OpenStack

The minimum hardware requirements for installing OpenStack may vary depending on the specific components and services you want to deploy, as well as the scale of your intended deployment. However, a general guideline for the minimum hardware requirements for an OpenStack deployment is given in Table 1.

Steps to install OpenStack on Ubuntu server

The seven key steps required for successful installation of OpenStack on an Ubuntu server are briefly listed below.

Update the system: Make sure your Ubuntu Linux system is up-to-date before installing OpenStack. To update your system's repositories, you can use the following command from your root account:

```
$ sudo apt-get update && sudo apt-get upgrade
```

Install the prerequisites packages: Now, you need to install Python and other required dependencies for the OpenStack setup.

```
$ sudo apt install -y python3-dev python3-pip libffi-dev
libssl-dev
```

Install the OpenStack packages: Give the following command to install these packages:

Table 1: Minimum hardware requirements to set up OpenStack

Component	CPU (cores)	RAM	Storage hard disk	Number of NICs
Controller node	**Dual-core processor**	8GB	100GB	2 NICs
Compute node	**Dual-core processor**	8GB	100GB	2 NICs
Block storage node	**Dual-core processor**	4GB	100GB	1 NIC
Object storage node	**Dual-core processor**	4GB	100GB	1 NIC

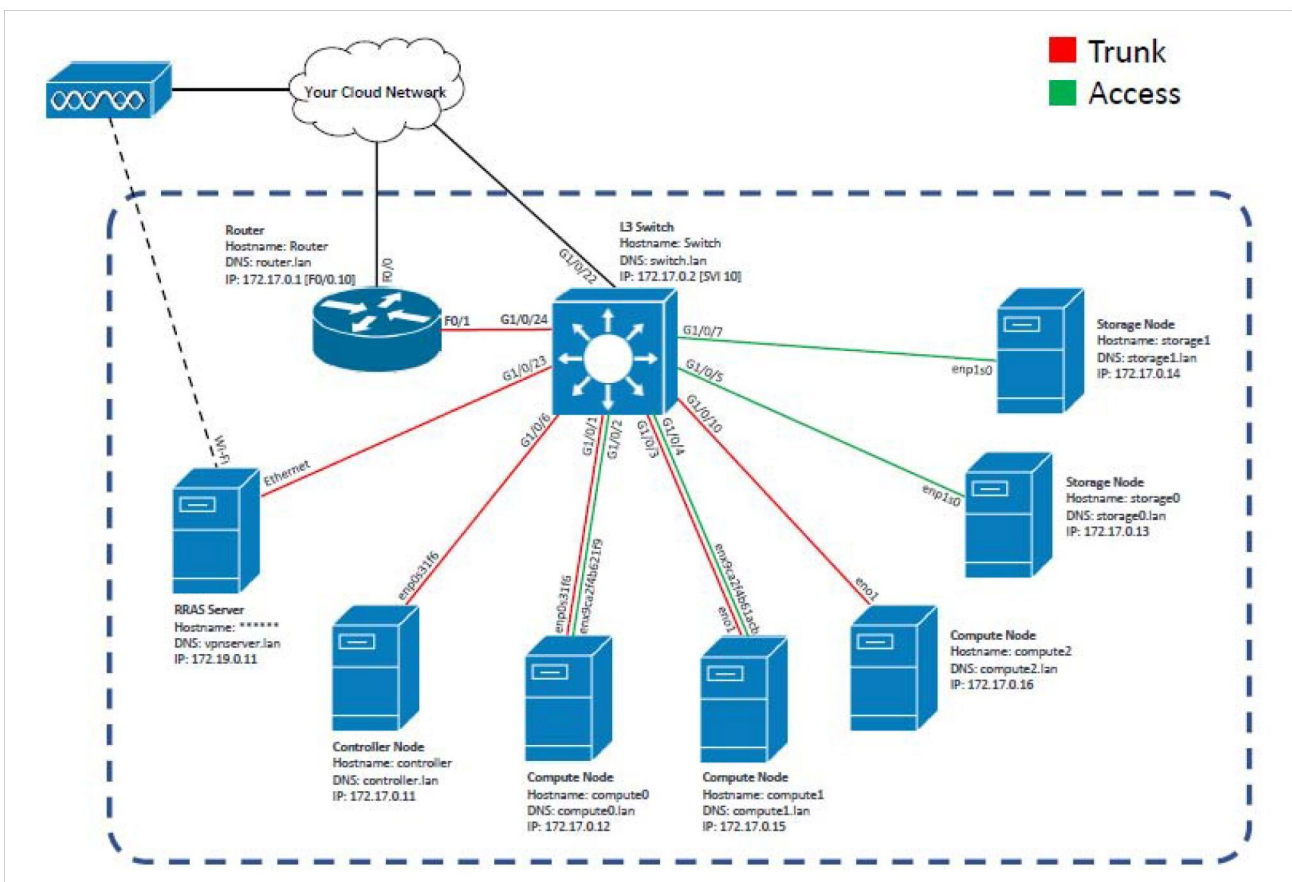


Figure 2: Networking configuration of deployed OpenStack architecture

```
$ sudo apt install -y software-properties-common
$ sudo add-apt-repository cloud-archive:wallaby
$ sudo apt update
$ sudo apt install -y openstack-dashboard
```

Configure networking: You may need to configure your networking settings based on your OpenStack deployment requirements. This involves setting up network bridges, configuring network interfaces, or other network-related settings. The following command is executed for configuration:

```
$ sudo dpkg-reconfigure openstack-dashboard
```

The networking configuration of our deployed OpenStack architecture is shown in Figure 2.

Configure authentication: OpenStack uses authentication mechanisms such as Keystone for managing users and roles. You'll need to configure authentication settings based on your deployment requirements, including setting up users, roles, and authentication backends.

Start OpenStack services: Give the following command to start OpenStack services:

```
$ sudo systemctl enable apache2
$ sudo systemctl start apache2
$ sudo systemctl enable memcached
$ sudo systemctl start memcached
$ sudo systemctl enable uwsgi
$ sudo systemctl start uwsgi
$ sudo systemctl enable horizon
$ sudo systemctl start horizon
```

Access OpenStack dashboard: You can now access the

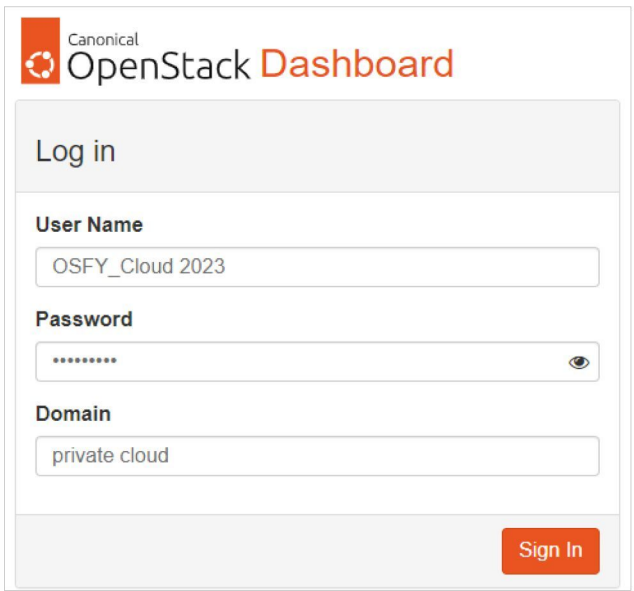


Figure 3: OpenStack dashboard login page

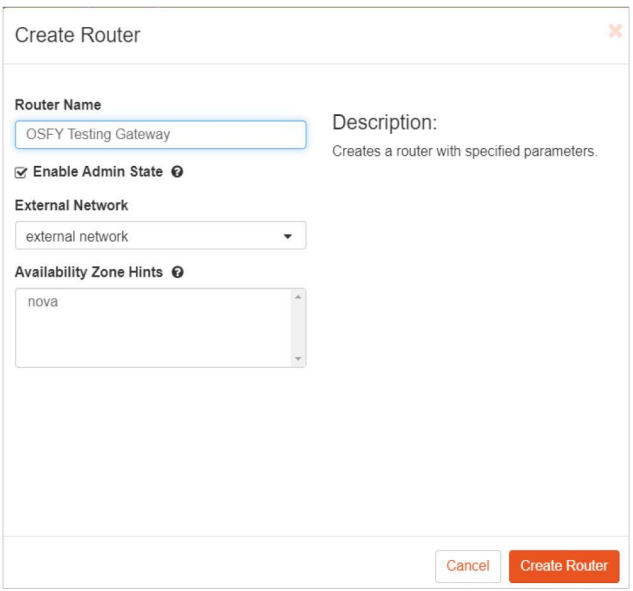


Figure 4: Create a router and an external network at OpenStack

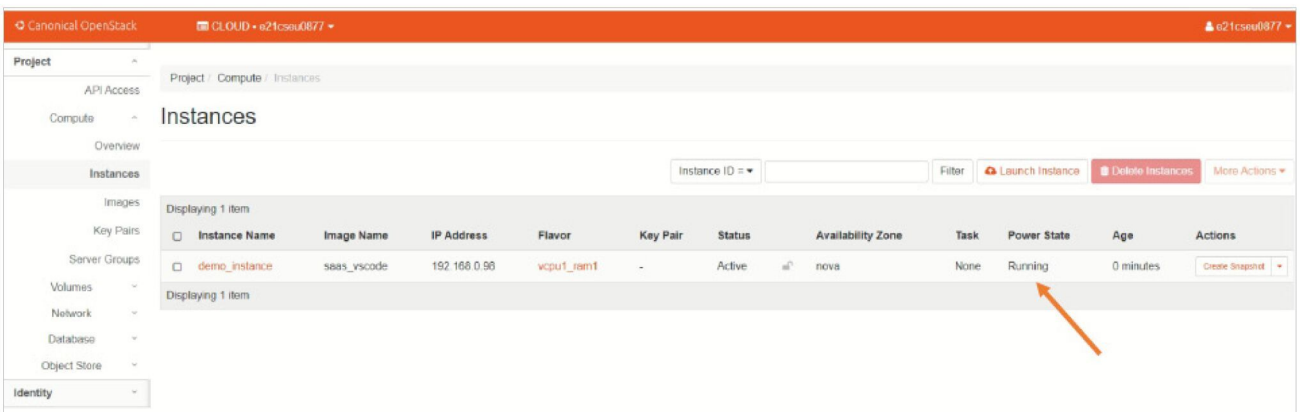


Figure 5: Final launch of virtual machine instance at the OpenStack

OpenStack dashboard by navigating to *http://<your-server-ip>/dashboard* in your web browser. This will direct you to the OpenStack login page, as shown in Figure 3.

Creating a virtual machine on the deployed OpenStack setup

Create router with the required parameters: After logging into the OpenStack dashboard, click on *network>routers>create a router*. As shown in Figure 4, provide the router name (OSFY Testing Gateway) and select the external network.

Configure the network: In this step, configure the network, subnet, and IP address for the deployed VM instance. In the subnet tab, provide a subnet name, network address, and gateway IP.

Launch instance: Finally, from the *compute->instance* option, launch the instance. After some time, the deployed instance will start running, as shown in Figure 5.

OpenStack is a popular choice for building private clouds due to its open source nature, flexibility, and modular architecture. It is widely used by organisations and cloud service providers to build and manage private, public, and hybrid clouds, as it offers a flexible and extensible solution for building scalable and interoperable cloud computing environments. END

References

- *https://www.openstack.org/software/project-navigator/openstack-components#openstack-services*
- *https://cloud.denbi.de/wiki/Concept/openstack/*
- *https://www.openstack.org/marketplace/hosted-private-clouds/*

By: Dr Aditya Bhardwaj

The author is a B. Tech, M. Tech, and PhD in CSE. He works as assistant professor in the School of Computer Science Engineering and Technology at Bennett University, Greater Noida. He is experienced in cloud computing and open source technology.

Continued from page...74

desc

The *desc()* function is used to sort a column in the descending order. The bank balances can be displayed from the highest to the lowest values, as shown below:

```
> bank %>% arrange(desc(balance))
   age    job  marital education default balance housing
loan

1     60  retired  married   primary   no   71188    no   no
2     42  entrepreneur  married  tertiary no  42045 no   no
3     43  technician   single  tertiary  no  27733  yes   no
4     36   management  married  tertiary no  27359  yes   no
5     57   technician  married  tertiary no  27069   no  yes
6     31   housemaid   single   primary  no  26965      no
no
    ...
```

nth

A specific row from the data set can be retrieved using the *nth()* function. The 10th entry in the bank data frame is shown below:

```
> bank %>% nth(10)
  age      job marital education default balance housing loan
contact day
1  43 services married   primary      no     -88     yes  yes
cellular  17
  month duration campaign pdays previous poutcome  y
1   apr      313        1   147        2  failure no
```

You are encouraged to read the dplyr package manual to learn more functions, arguments and usage. END

References

- [Moro et al., 2014] S. Moro, P. Cortez and P. Rita; A Data-Driven Approach to Predict the Success of Bank Telemarketing; Decision Support Systems, Elsevier, 62:22-31; June 2014
- Package dplyr; *https://cran.r-project.org/web/packages/dplyr/index.html*

By: Shakthi Kannan

The author is a free software enthusiast.

Ansible AWX:
The GUI Configuration Management Automation Tool You Will Love to Use

Ansible AWX is an open source community project that provides a web based user interface and API to manage an organisation's Ansible playbooks, inventories, vaults, and credentials. It is an open source version of Ansible Tower. Ansible AWX makes Ansible simpler for IT teams that are not comfortable with command lines by providing a GUI version of Ansible. This article is a tutorial on how to install and configure it.

Ansible AWX services are deployed inside containers; hence Docker and Docker Compose must be installed in order to run multiple container images. Version 18.x onwards AWX is supported by red hat team via awx operator and not using docker-compose. This installation procedure requires a Kubernetes cluster/ minikube and the setup is pretty easy, as described below.

- Docker should be installed in the server machine where AWX is installed.
- Python 3 should be installed on the AWX server and all target machines.
- We are using awx-ee:21.11.0 version.
- Our target machines/hosts are localhost and container.

Installing AWX on Ubuntu using Kubernetes/ minikube cluster

For installing AWX in minikube, we have followed the procedure given at *https://github.com/ansible/awx-operator* with some additional steps.

1. Update and upgrade your Debian system before you install Ansible AWX using the following command:

```
sudo apt update && sudo apt -y full-upgrade
```

2. To create a minikube cluster, first install the latest minikube stable release on x86-64 Linux using the command given below:

```
curl -LO https://storage.googleapis.com/minikube/releases/
latest/minikube-linux-amd64.
sudo install minikube-linux-amd64 /usr/local/bin/minikube
```

3. From a terminal, run the following command with sudo privilege (but not logged in as root) to start Kubernetes/ minikube with the required CPU number and RAM size:

```
minikube start --cpus=4 --memory=6g --addons=ingress
```

4. Once minikube is deployed, we can check if the node(s) and kube-apiserver communication are working as expected or not by executing the command:

```
minikube kubectl -- get nodes
```

```
apiVersion: kustomize.config.k8s.io/v1beta1
kind: Kustomization
resources:
  # Find the latest tag here: https://github.com/ansible/awx-operator/releases
  - github.com/ansible/awx-operator/config/default?ref=1.1.4
  # Add this extra line:
  - awx-demo.yaml

# Set the image tags to match the git version from above
images:
  - name: quay.io/ansible/awx-ee:21.11.0
    newTag: 1.1.4

# Specify a custom namespace in which to install AWX
namespace: awx
```

Figure 1: *kustomization.yaml* file

5. By executing the following command we can verify whether Kubernetes has started some pods or not:

```
minikube kubectl -- get pods -A
```

We do not need to install kubectl separately since it is already wrapped inside a minikube.

6. Once Kubernetes starts making pods, let's create an alias for easier usage using the following command:

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

7. When our Kubernetes cluster starts running, we can deploy AWX Operator in our cluster using Kustomize, which is a Kubernetes configuration transformation tool. To install Kustomize by downloading precompiled binaries use the command given below:

```
curl -s "https://raw.githubusercontent.com/kubernetes-sigs/kustomize/master/hack/install_kustomize.sh"  | bash
```

8. Confirm the installation of Kustomize by checking the version using the following command:

```
kustomize version
```

9. Next, create a file called *kustomization.yaml*, which has the content shown in Figure 1.

In place of *newTag* and *ref* we can pass the latest version of AWX Operator, which can be found at *https://github.com/ansible/awx-operator/releases*. Here, we are using AWX Operator version 1.1.4. We can also save the latest version from AWX Operator releases as *RELEASE_TAG* variable, and pass that variable instead of passing the hardcoded latest AWX Operator version.

The AWX Operator is used to manage one or more AWX instances in any name space within the cluster.

10. Now, we have to install the manifests by running the following command:

```
kustomize build . | kubectl apply -f -
```

11. Wait for a few minutes and check if the AWX operator is deployed or not by using the command:

```
kubectl get pods -n awx
```

If the status shows *Running*, it means our operator has been deployed successfully.

12. Since we would rather not keep repeating *-n awx*, we should set the current name space for *kubectl* using the command given below:

```
kubectl config set-context --current --namespace=awx
```

13. Now create another file named *awx-demo.yaml* in the

```
apiVersion: awx.ansible.com/v1beta1
kind: AWX
metadata:
  name: awx-demo
spec:
  service_type: nodeport
  # default nodeport_port is 30080
  nodeport_port: <nodeport_port>
```

Figure 2: *awx-demo.yaml* file

same folder with the content shown in Figure 2. The name we mention in metadata will be the name of the resulting AWX deployment. The port number to run AWX can be mentioned in the file, or the system will assign the default port number.

14. This *awx-demo.yaml* file is to be added in the list of resources in the *kustomization.yaml* file, as shown in Figure 3.
15. Finally, we have to run Kustomize again to create the AWX instance in our cluster by executing the following command:

```
kustomize build . | kubectl apply -f -
```

16. After a few minutes, the new AWX instance will be deployed in a cluster and we can also monitor its installation logs using the command:

```
kubectl logs -f deployments/awx-operator-controller-manager -c
awx-manager
```

17. After a few seconds, we should be able to see that the operator has begun to create new resources by using the command given below:

```
kubectl get pods -l "app.kubernetes.io/managed-by=awx-operator"
kubectl get svc -l  "app.kubernetes.io/managed-by=awx-operator"
```

18. When AWX is deployed in a cluster, we can access the AWX instance by running the command given below:

```
minikube service -n awx awx-demo-service --url
```

19. The user name is *admin* by default and we can get the password using the following command:

```
kubectl get secret awx-demo-admin-password -o jsonpath="{.
data.password}" | base64 --decode ; echo
```

This completes the basic installation. Go to the URL and log in using *admin* as user name and password.

Running Ansible Playbook using AWX

- We can get the URL of the Ansible AWX dashboard by using the following command:

```
minikube service -n awx awx-demo-service --url
```

- The Ansible AWX web portal is now accessible on *http://hostip_or_hostname:30080* (by default, the port number is 30080 if we have not set the port number of AWX in the *awx-demo.yaml* file).
- Launch your browser to access the dashboard and you will get a screen as shown in Figure 4.
- Use *admin* as user name and get the password by running the following command:

```
kubectl get secret awx-demo-admin-
password -o jsonpath="{.data.
password}" | base64     --decode ; echo
Sample password:
LkyWUKDwKdnhiEcvFe0zRQ9jOJCz7eM
```

- Log in using the user name and password, and enter into the AWX Administration Dashboard, which is shown in Figure 5. Now we can start adding inventory, credentials, hosts, projects, templates and Ansible roles, and automate our infrastructure and application deployment.

Next, select the organisation tab on the left side of the screen, click on the *Add* button to add a new organisation, add the name of the organisation and click on *Save* (Figure 6).

Inventory setup

- Now go to *Inventory* and click on the *Add* button to create inventory.

```
...
resources:
  - github.com/ansible/awx-operator/config/default?ref=<tag>
  # Add this extra line:
  - awx-demo.yaml
...
```

Figure 3: *kustomization.yaml* file content

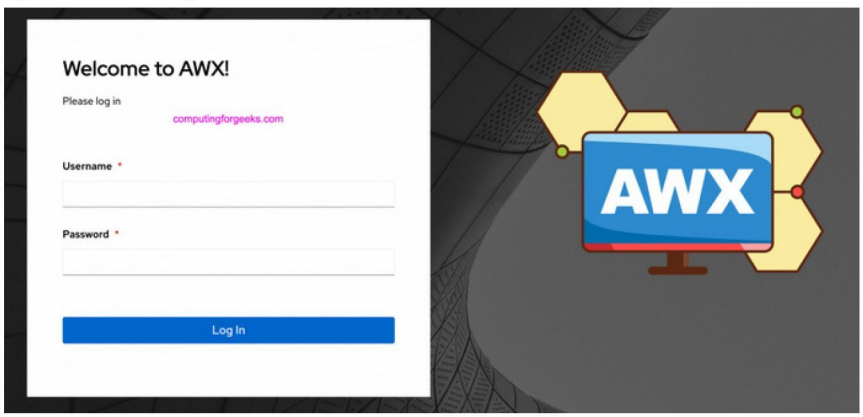


Figure 4: AWX login page

- Enter inventory name and if you want to create this inventory for a specific organisation, you can select the created organisation, or else choose *Default* and then save it.
- In variable, we pass those parameters that we want to apply to all hosts connected to that inventory. Here we want to connect with hosts/target machines using ssh. So we have to pass *ansible_connection: ssh* in variable.
- If you get an error related to the Python interpreter while executing the template, add *ansible_python_interpreter: '{{ ansible_playbook_python }}'* in variables of that inventory, as shown in Figure 7.

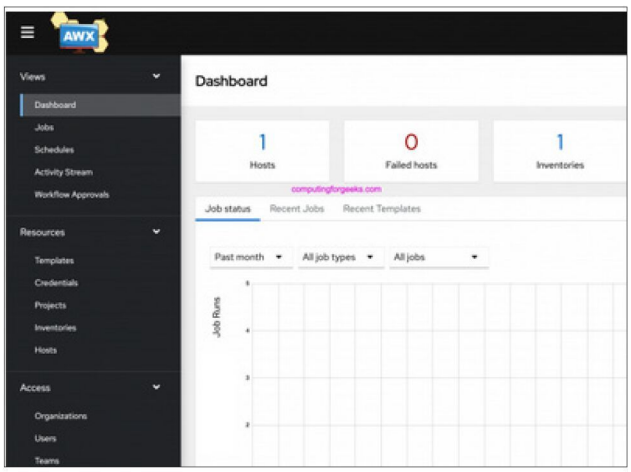


Figure 5: Dashboard

Configuration of hosts

- To add a host, which you want to access using AWX, click on *Hosts* and then select the *Add* button. Add the IP address or host name of the host/target machine, choose the inventory to which you want to add this host, and then add details of that host and save it as shown in Figure 8.
  We have to pass variables in hosts, as shown below:

```
ansible_host: 172.16.145.49                    # ip address
of host
ansible_user: suchi                                    #
enter username of host you are accessing
ansible_become: true                                  #
Ansible_become used for privilege escalation.
ansible_ssh_pass: Suchi@123                 # password of
ansible_user of host machine
ansible_sudo_pass: Suchi@123
```

If the *ansible_user* is root, then we don't need to pass *ansible_sudo_pass*. If you want to connect with the target machine/hosts through ssh without password, it is not required to pass *ansible_ssh_pass* variables in the host configuration.

Adding credentials

- If you want to access the target machine/hosts using the password, we can directly select *demo credentials* and nothing needs to be configured.

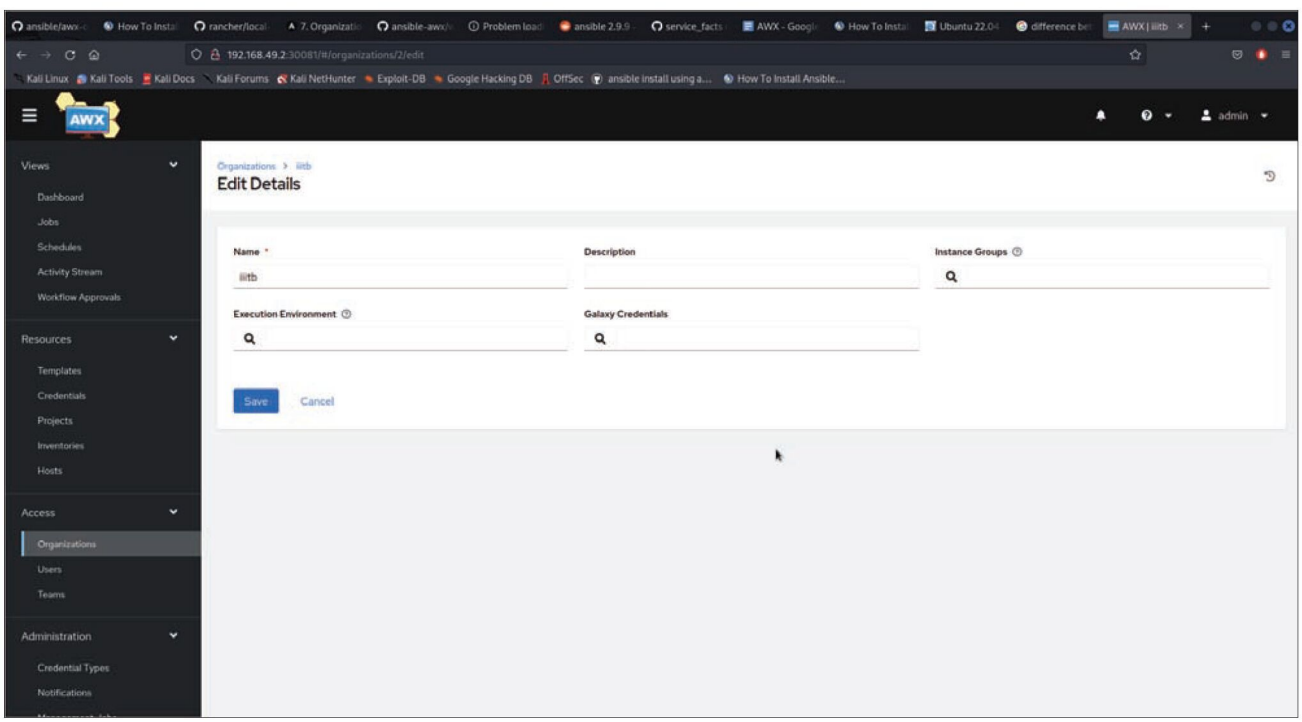


Figure 6: Screenshot to add organisation

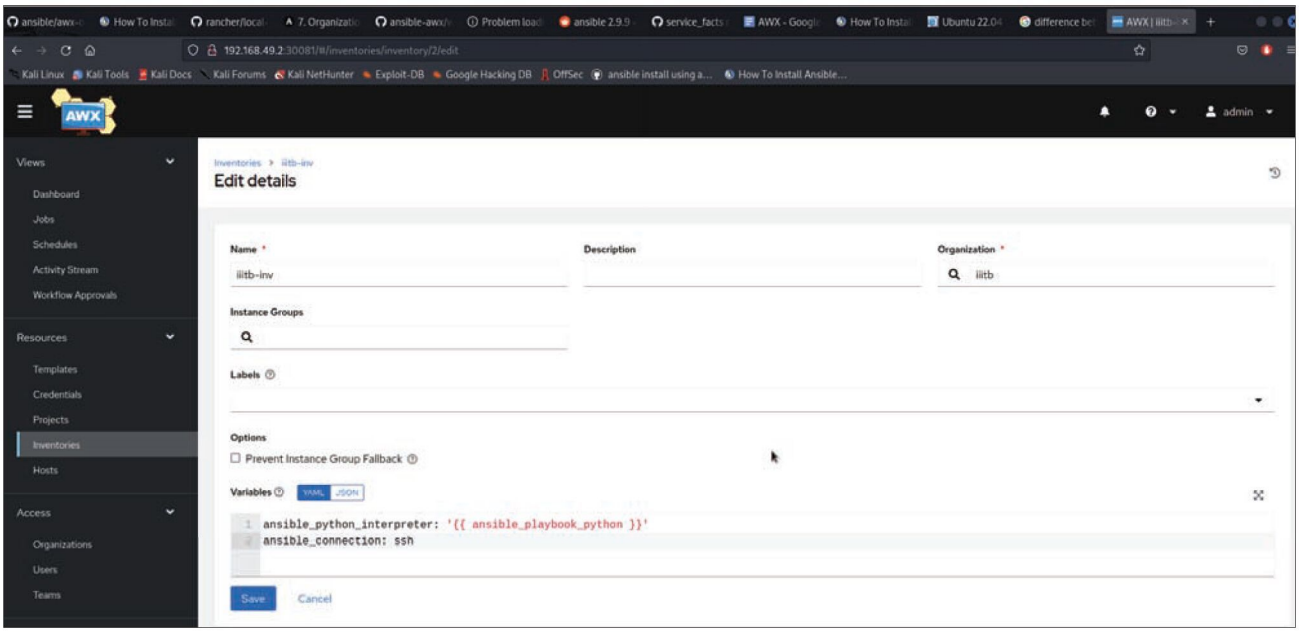


Figure 7: Screenshot to add inventory

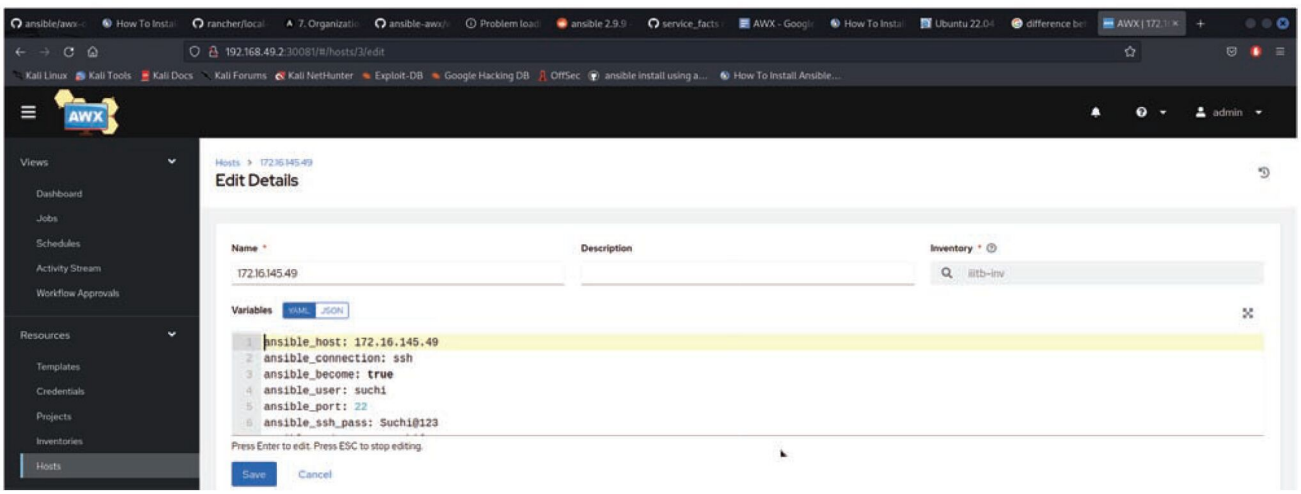


Figure 8: Screenshot to add hosts

- But if you want to access the target machine/hosts using ssh without password, we have to use the ssh key sharing mechanism.
- Go inside minikube where AWX is installed by using the command:

```
docker exec -it minikube_container_id /bin/bash
```

- Generate the public and private keys of minikube machine using the command:

```
ssh-keygen
```

- Now, copy the public key of minikube in all target machines/hosts using the command *ssh-copy-id user@ip* from the minikube terminal. It will copy the public key of minikube in targets.
- Here, *user* is the user of the host you are accessing and its IP that you have mentioned in this host configuration.
- Now, you have to add the private key of minikube in the AWX credential. This private key is available in *cat ~/.ssh/id_rsa*.

Next, go to the credential tab in AWX, click on the *Add* button, give credential name, and copy paste the private key.

In user name, add the login user of AWX who will execute templates. In our case, we are using *admin*. Select the name of the organisation with which the credential is associated.

Select *sudo* in *Privilege Escalation* and *root* in *Privilege Escalation* user name to give root permission to execute. Then save this, as shown in Figure 9.

Creation of projects

- Now select the *Project* tab and click the *Add* button. Basically, the project is the collection of playbooks you want to execute in your host machine using AWX.
- Give your project a name and select your organisation. Here, we are fetching playbooks from GitHub; so, select *git* in source control type and enter the URL of the GitHub playbook repository where all the playbooks are present.
- It is recommended to use GitHub to fetch playbooks instead of fetching from the machine, as there will be problems with maintaining version control in the local machine.
- Enable update revision on launch so that whenever we launch our project, i.e., run our template, it will always fetch the latest version of playbook. So even in case there are changes in playbook in GitHub, only the updated version will always be fetched.
- Then save this, as shown in Figure 10.
- Select the GitHub account from where we are fetching playbooks in AWX, which is shown in Figure 11. This URL has to be used in the source control URL in the project shown in Figure 10.

Creation of templates

- Now select the *Template* tab, click on *Add* button, enter template name and select the project, inventory and

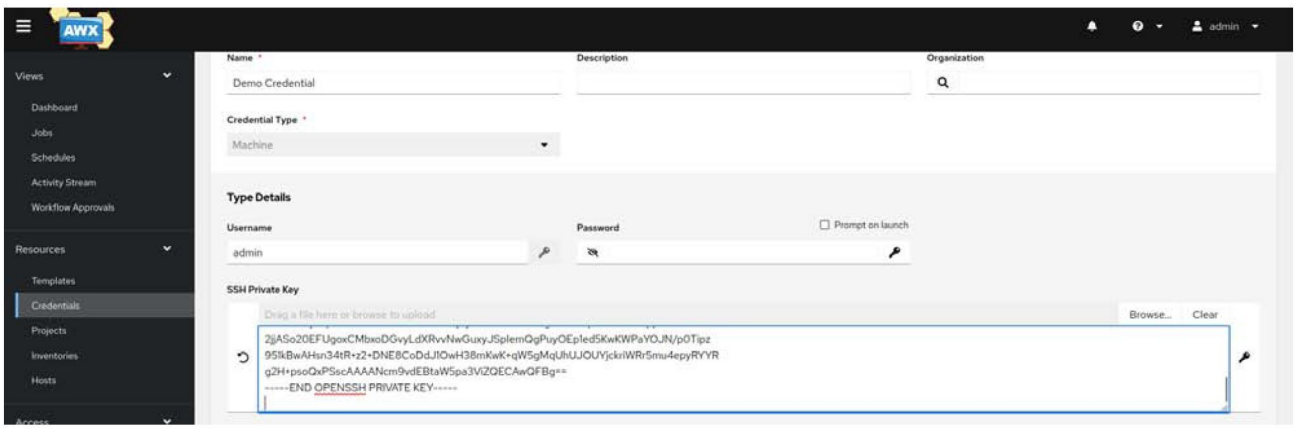


Figure 9: Screenshot to add credentials

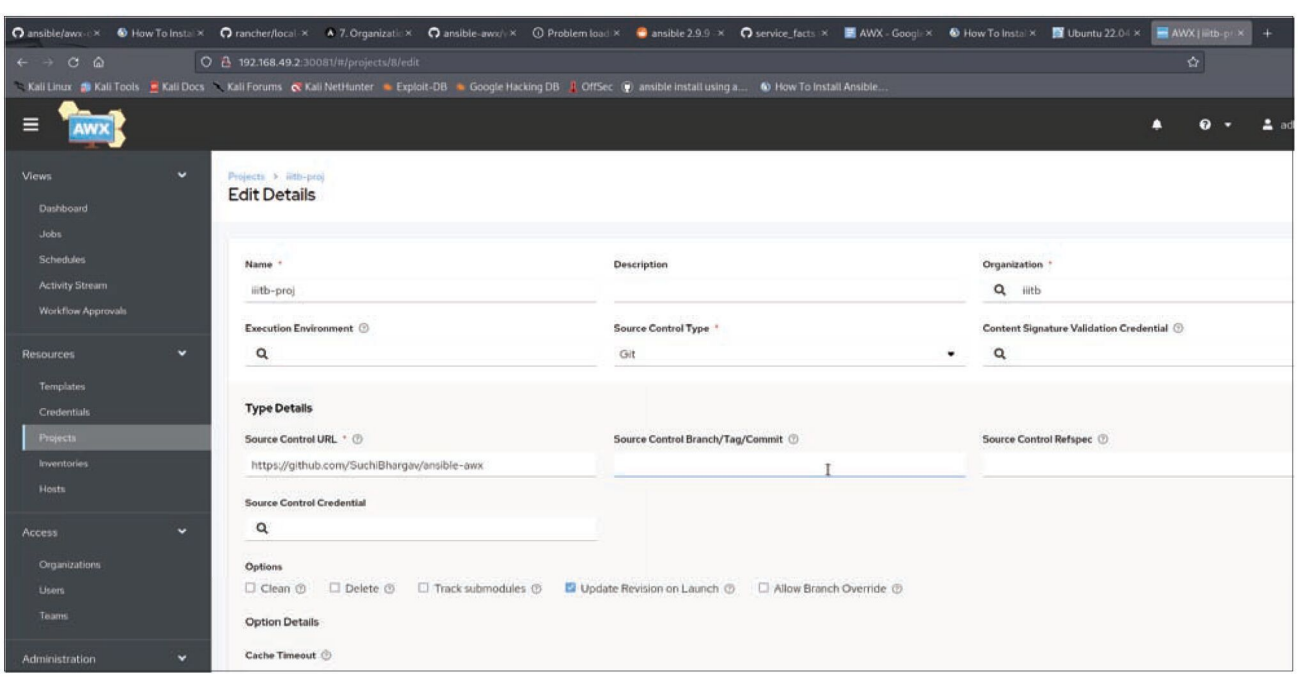


Figure 10: Screenshot to add project

playbook present in your GitHub repo. Also enable the *Privilege Escalation* option at the bottom of the template and save it (Figure 12).

- Templates are basically jobs where we define what playbook we want to run and what inventory source the hosts that we want to run against are in.

Our template is now ready to launch. Just click on the rocket icon or *Launch* button to run the playbook.

Some points to be noted when using Ansible AWX for accessing hosts

- If we want to access any container and deploy any package or application inside it using Ansible AWX, we must create a Docker network and connect both minikube and the container in the same network, so that they can communicate with each other.
- Note that minikube can directly access and be deployed inside localhost without the need to connect both in the same network.
- Commands to be used to create a network, and connect minikube and container in the same network, are shown below:

```
docker network create -d bridge network_name
docker network connect  network_name  container_id
docker network connect  network_name  container_id_of_
minikube
```

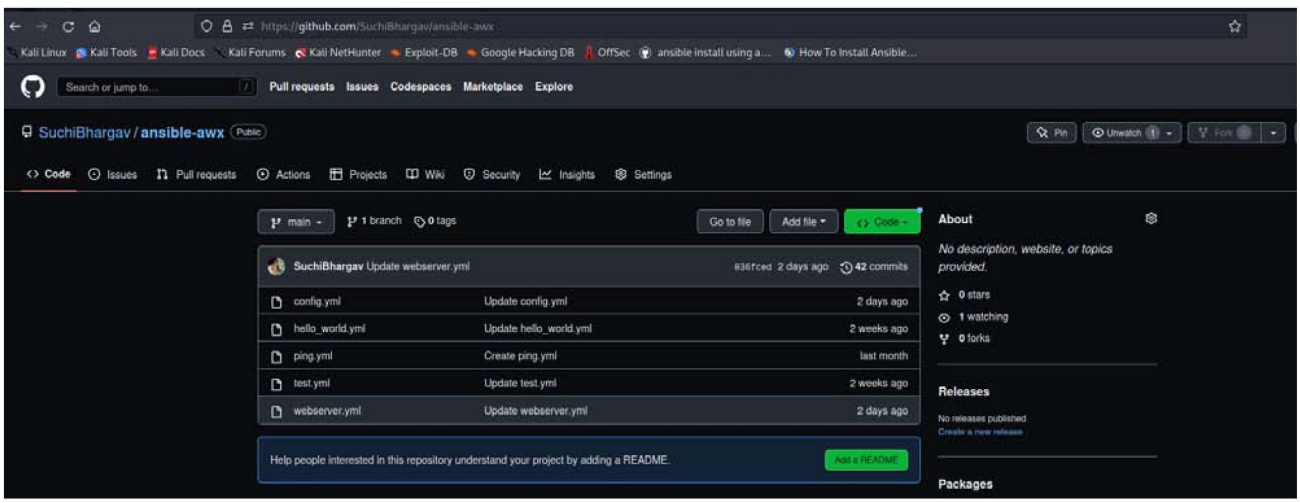


Figure 11: GitHub repository where all the playbooks are present

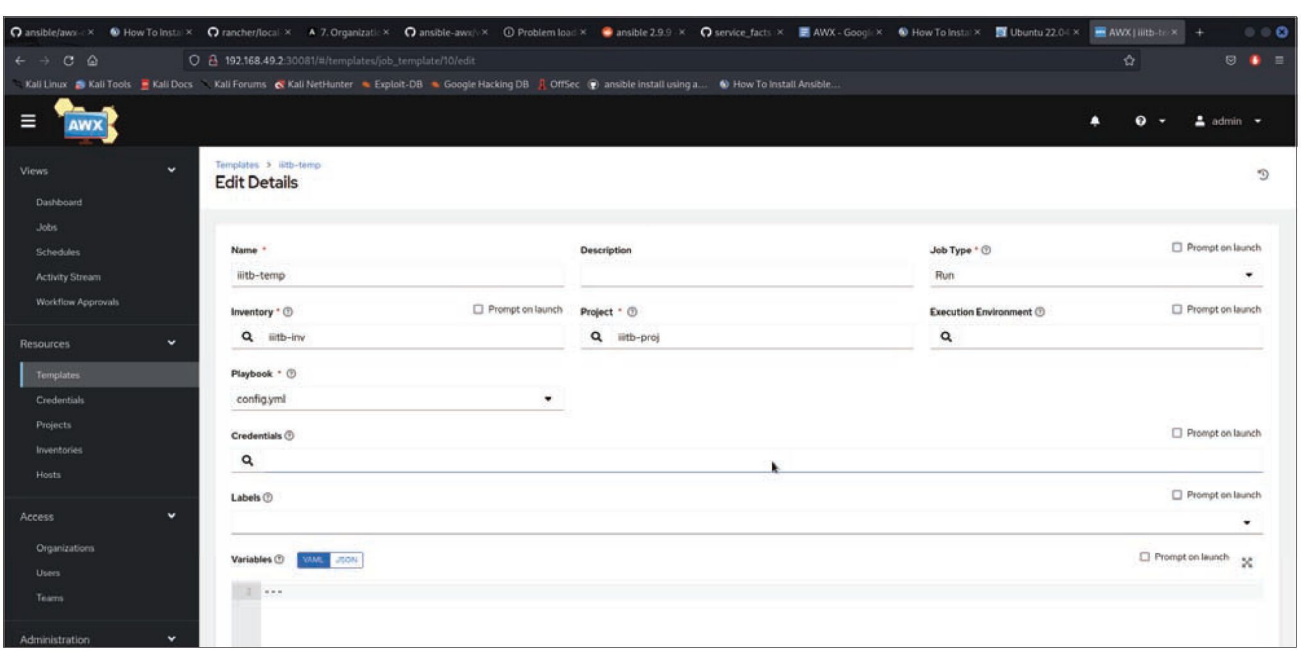


Figure 12: Screenshot to add templates

Figure 13: Screenshot of Playbook 1

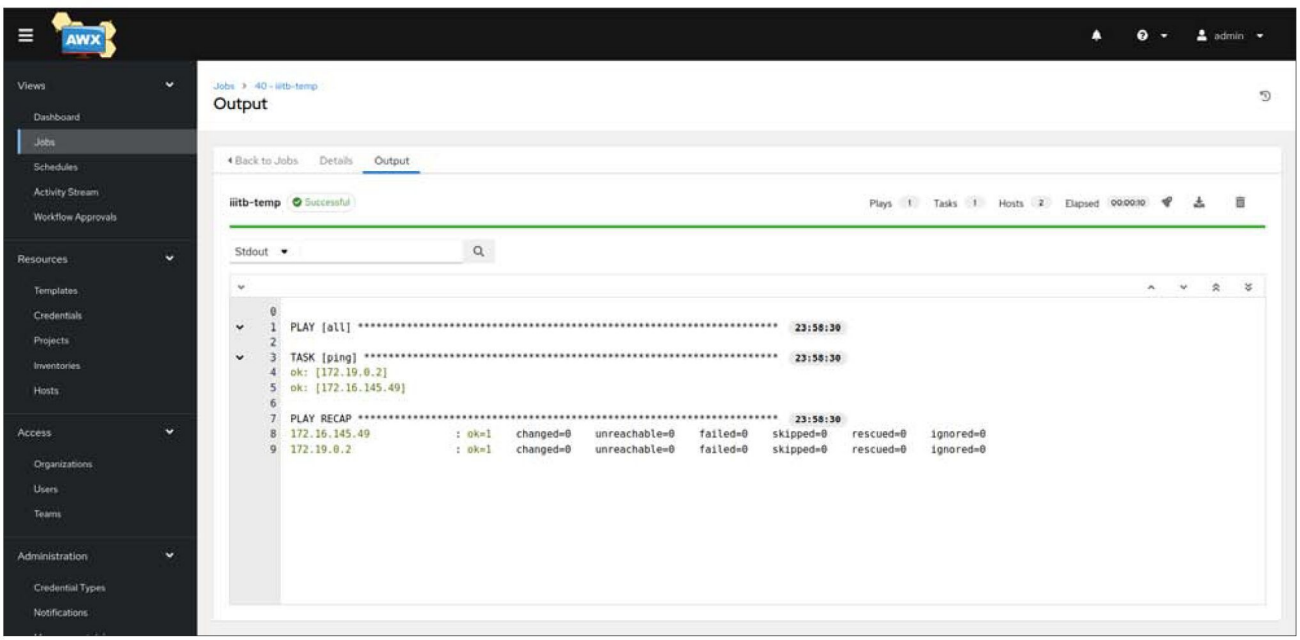


Figure 14: Output of Playbook 1

- After execution of these commands, minikube and container can communicate with each other through the network.
- Then, we have to start ssh service (*service ssh start*) in both localhost and container so that minikube can communicate with both using ssh.
- We can also cross-check if ssh is working or not by going inside the minikube AWX container using the following command:

```
docker exec -it container_id /bin/bash
```

- Now connect to the hosts through ssh by executing the command *ssh root@ip* or *ssh user@ip*. Here *ip* is the IP address of the host with which AWX wants to communicate to check if it is able to ssh into that host or not. *user* is the user name of the host you want to communicate with.

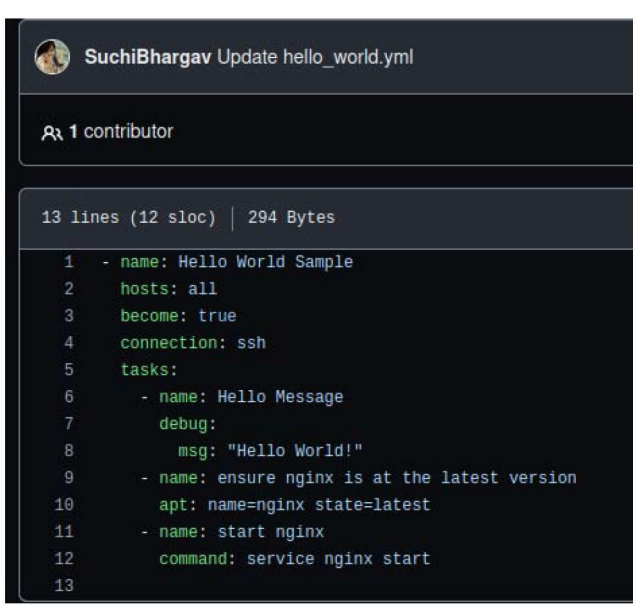


Figure 15: Screenshot of Playbook 2

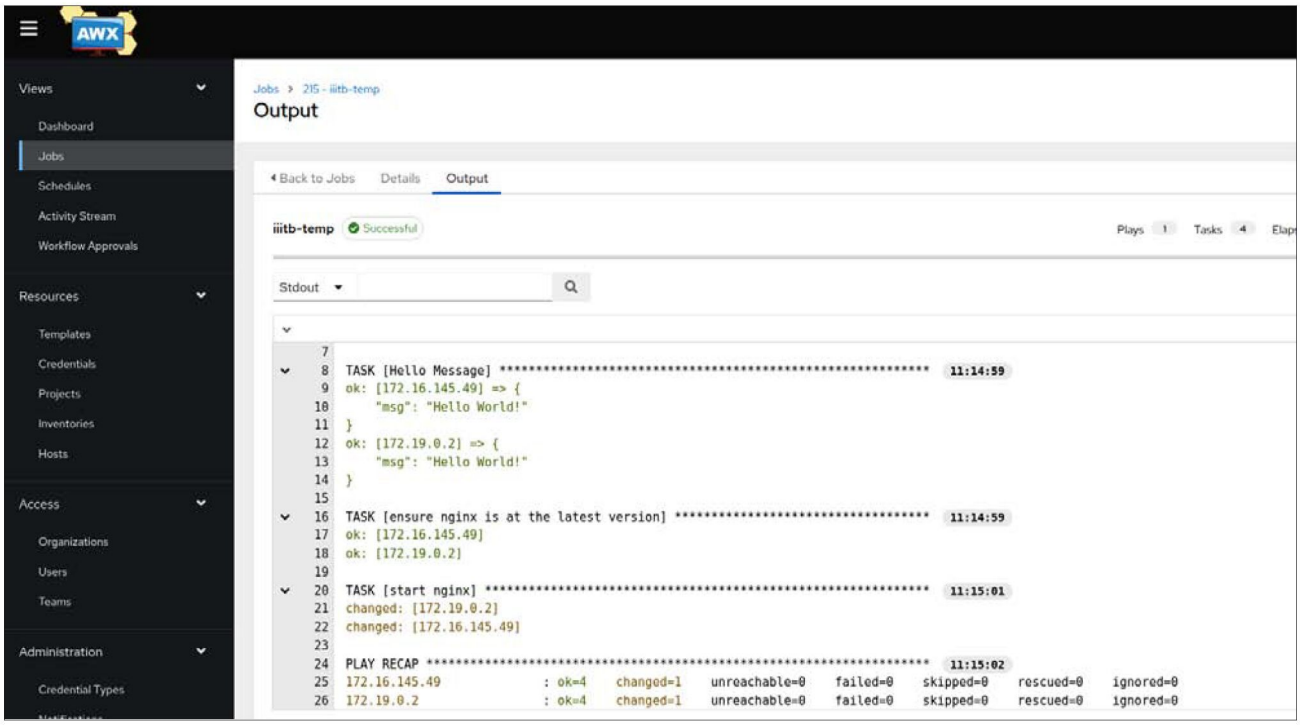


Figure 16: Screenshot after Playbook 2 execution

- In case you are accessing the root user of the host/target machine using ssh by AWX, you have to configure the */etc/ssh/sshd_config* file in the host and modify the parameters as shown below.

```
permitrootlogin yes
PubkeyAuthentication yes
PasswordAuthentication yes
```

Now restart ssh service using the command:

```
service ssh restart
```

You may get some errors during execution. Some of these are listed below.

- While doing ssh, if you get the error *permission denied* even after entering the correct password or get the error *Host key verification failed*, use the command *ssh-keygen -R ip*. Here, *ip* is the IP address of the client you want to ssh and execute the command *ssh root@ip*.
- When executing playbook, if you are getting the error *The module fails to execute correctly, you probably need to set the interpreter.\nSee stdout/stderr for the exact error,* then you have to check whether Python is installed and the command is available in */usr/bin*.

We have completed AWX installation, configured all the parameters, and enabled the network connection between hosts and targets. Now let us execute some example playbooks to configure infrastructure in the target systems.

Executing playbooks

Playbook 1: *Ping.yml*

This playbook will ping all the hosts that are attached with inventory. Figure 13 shows the screenshot of the playbook.

After the template is launched to execute it, the output is shown in Figure 14.

Playbook 2: *hello_world.yml*

This playbook (Figure 15) will print 'hello world' and then install NGINX server inside localhost and container. The execution of this playbook can be seen in Figure 16.

By using the command *service –status-all*, we can verify whether NGINX has started or not after execution of the above playbook.

We have seen only two playbook examples in this article but you can execute any playbook in this user-friendly graphical user interface called Ansible AWX. END

By: Suchi and Dr B. Thangaraju

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

Using *ffmpeg* for Intruder Protection in Linux

None of us want others intruding into our Linux machines. *ffmpeg* is a tool that can detect when someone is trying to get into our system. Let's see how we can use it to secure ourselves from such unwanted invasions.

Data is a major resource today and securing all our devices has become crucial. We would like to protect ourselves from anyone logging into our laptop or computer.

If we are using hardware shared by multiple people who have multiple user IDs, there is a chance that another user may want to log into our system for our files, etc. So let us understand how we can detect such intrusions in our Linux machines, by setting up a program that takes a photo when a wrong password is entered in our system.

The first step is to install the *ffmpeg* package. This can be done using the following command:

```
sudo apt-get install ffmpeg
```

Next, create a shell script, as shown below. First, open any editor. I am using gedit here; you can use it too with the following command:

```
sudo gedit /usr/local/bin/passwordpicture
```

When you run this command, the editor will open and you can paste the following shell script in it, as shown in Figure 1:

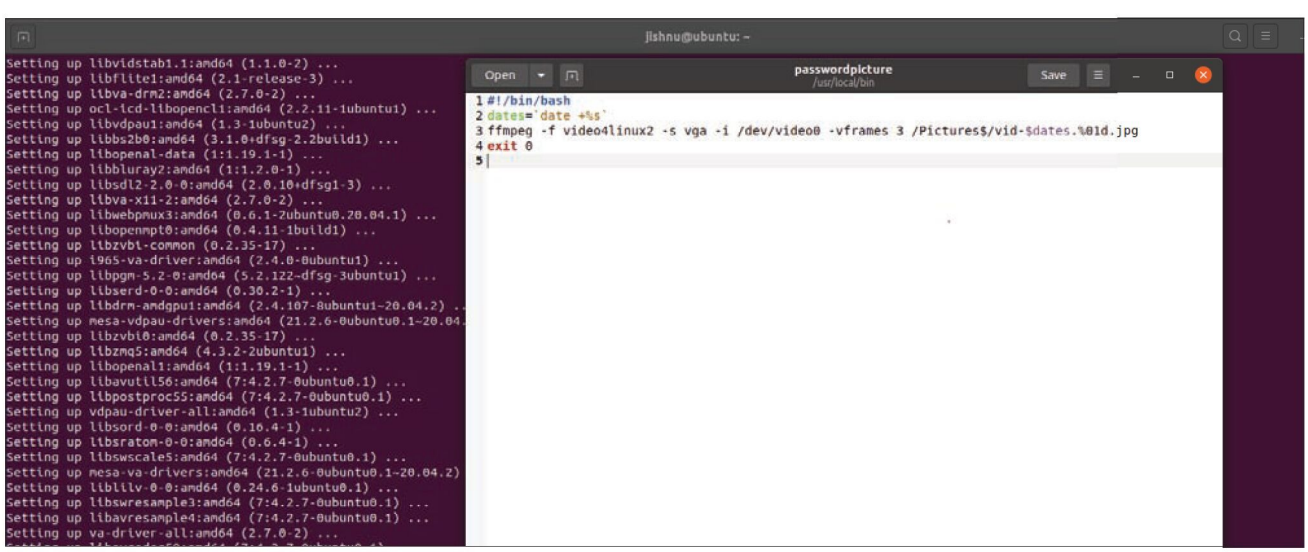


Figure 1: Image clicking scripts

```
#!/bin/bash
dates=`date +%s`
ffmpeg -f video4linux2 -s vga -i /
dev/video0 -vframes 3 /Pictures$/vid-
$dates.%01d.jpg
exit 0
```

If your video source is different, you can replace */dev/video0* with the path of your camera. I have stored the clicked photos in the */Pictures* folder; if you want to store your pictures in some other location, do change the path for that as well.

Now run the following command to make the written script executable:

```
chmod +x /usr/local/bin/
passwordpicture
```

This is basically the script to click the photograph. Now, let's write the condition to take the photo only if the password entered into the system is wrong.

Open the following file, using this command:

```
sudo gedit /etc/pam.d/common-auth
```

Change the following line:

```
auth    [success=1 default=ignore]
pam_unix.so nullok_secure
```

…to:

```
auth    [success=2 default=ignore]
pam_unix.so nullok_secure
```

In order to not take cognisance when the password is correct, add the following just below this line:

```
auth    [default=ignore]
pam_exec.so seteuid /usr/local/bin/
passwordpicture
```

This will click a picture if a wrong password is entered while logging into the system.

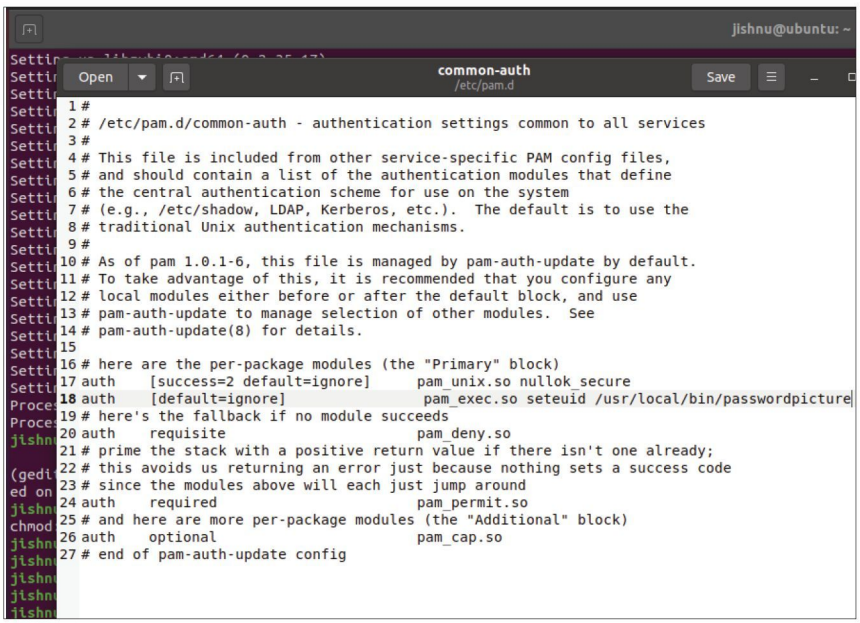


Figure 2: Full script of login

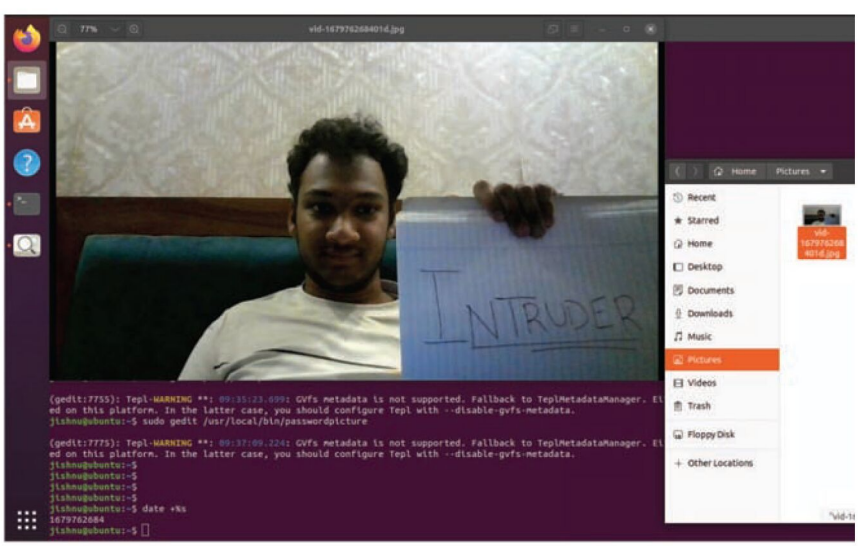


Figure 3: Picture stored in folder

Finally, the overall script should look as shown in Figure 2.

Now save and close it.

If you now log out and try to log in with a wrong password, a picture will be clicked, and it will be stored in the */Pictures* folder, as shown in Figure 3.

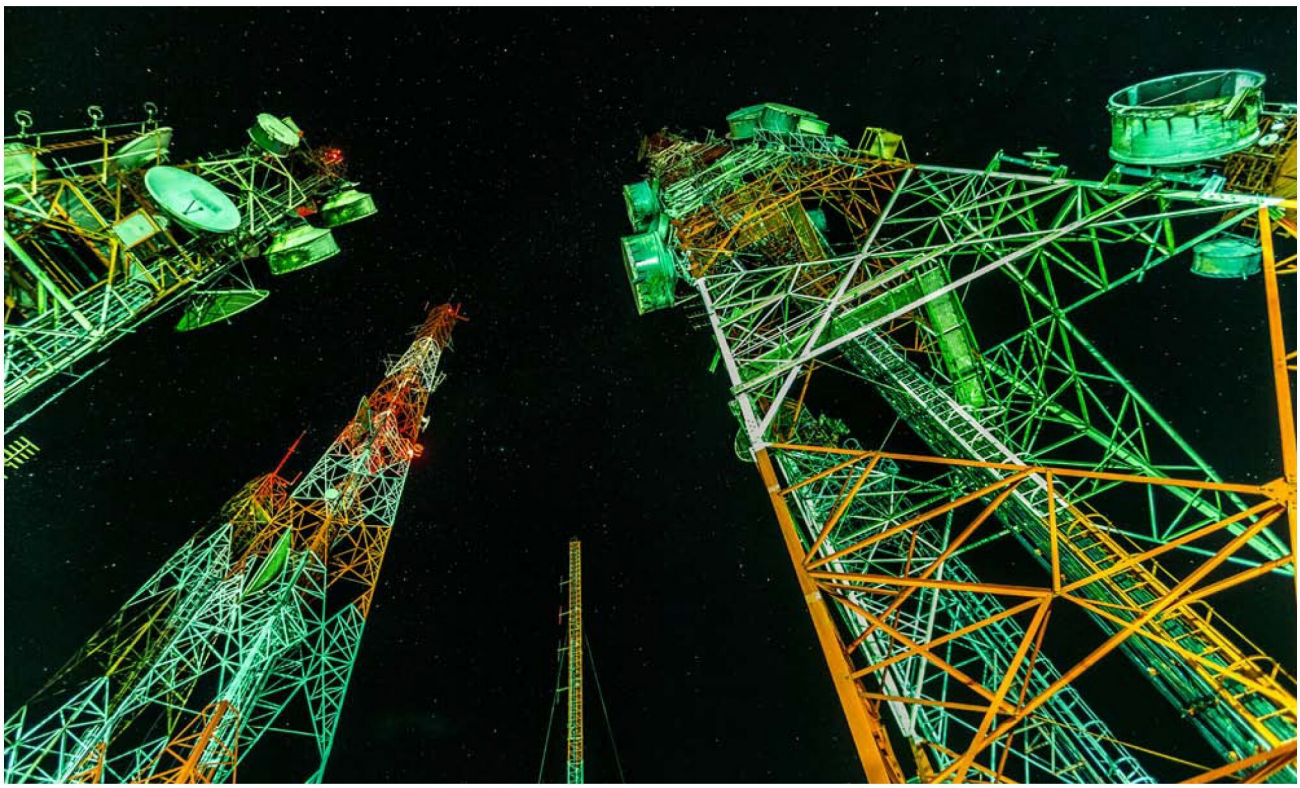
I hope you found this interesting and are going to try it out. And if you explore the tool some more, you will get many more such ideas!! This concept could be implemented while building a larger security system for offline computers. END

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.

The Role of Network Function Virtualization in Telecom Infrastructure

This sixth article in the series on integration of network function virtualization (NFV) with the DevOps pipeline talks about the advantages of NFV and the best way to manage NFV infrastructure.



Service providers are always looking for solutions that provide the latest telecom applications and comply with the demands of their users. These solutions should not be a bottleneck while scaling or call for hefty financial investment. The hardware deployed is specialised to do a task specific to the service — for example, firewall, monitoring, routing, and more. However, capex and opex investments will see a rise if this proprietary hardware reaches the end of life quickly due to rapid innovations.

Network function virtualization

Network function virtualization (NFV) brings service providers out of this bottleneck. It is a cost-effective implementation of telecommunications infrastructure. The functioning proprietary hardware is replaced with an IT solution with the use of virtualization. NFV connects a wide range of industry-grade servers and hardware, which are driven by a common set of virtual network functions (VNFs). It aims to transform the way network communications connect to network equipment such as switches and storage (the latter is remotely located in either data centres or end user premises).

The book titled 'Network Function Virtualization' by Ken Gray and Tom Nadeau defines NFV as: "NFV

describes and defines how network services are designed, constructed, and deployed using virtualized software components and how these are decoupled from the hardware upon which they execute."

In Figure 1, we can see how traditional network devices such as routers, VPNs, load balancers, and others can be first converted to software-based functions and then placed on standard switches and servers. These functions run as virtual applications over the decoupled hardware. This NFV-based approach does not require specialised hardware.

NFV and SDN

SDN (software defined network) is a component and enabler of NFV. It paves the way for virtualization and orchestration. With the advent of SDN came the separation of the control and data plane, which served NFV by providing programmable connectivity between VNFs. These connections are further managed by the orchestrator of the VNFs, which acts as an SDN controller. NFV can be implemented without SDN too, but the two are combined for greater performance and scalability, and to facilitate operations and maintenance procedures.

Advantages of NFV

NFV brings a lot to the table. If implemented widely, it can enable great flexibility and scalability in telecom infrastructure.

Scalability: By leveraging the power of NFV, we can make a serious dent in capex and opex expenditures. The operator need not look into network scalability demands but only concentrate on consumers and cater to their service usage.

Flexibility: With software-based applications, it is easy to seamlessly integrate new services without any prerequisites. NFV assures higher adaptability and easy installation.

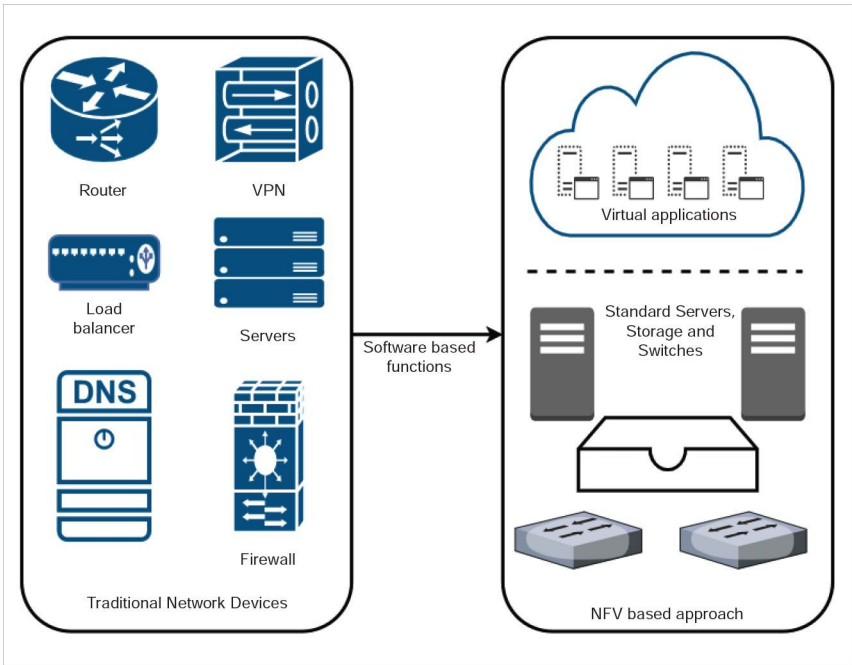


Figure 1: NFV-based approach compared to the traditional approach in telecom infrastructure

Time: The installation time too can be narrowed down with automation and tools.

Cost: The expense of buying new network devices, their cooling equipment, the space to keep them, and the power to operate them comes down steeply with NFV. Much of the software is accessible under an open source licence.

Security: With NFV, operators are able to run software-based virtual firewalls between the networks. They have greater control over the traffic and services between the networks.

NFV infrastructure management

OpenStack, the popular open source cloud computing software project, is a good solution for NFV infrastructure management. It provides the APIs to service computing, storage, and networking. This IaaS platform provides an interface to control the NFV infrastructure, which is deployed between the virtual machines and other networking hardware such as switches, routers, and load balancers. The life cycle of virtual machines is also carried out on this platform — they can be deployed, managed, scaled, and destroyed on it.

NFV infrastructure is all about virtualization and OpenStack does provide that with its various components, as shown in Figure 2. Many telecom companies recognise the potential of OpenStack as it offers a choice to manage and orchestrate all these virtual functions. Ninety-six per cent of cloud service providers (CSPs) implementing NFV strategies say OpenStack is essential to their success *(https://www.openstack.org/use-cases/telecoms-and-nfv/)*. The open, modular and interoperable framework of the OpenStack project offers telecoms and enterprises the ability to design the NFV system of their choosing, without using unnecessary components.

But that's not all when it comes to NFV technology. VMs are fine but the technology is a

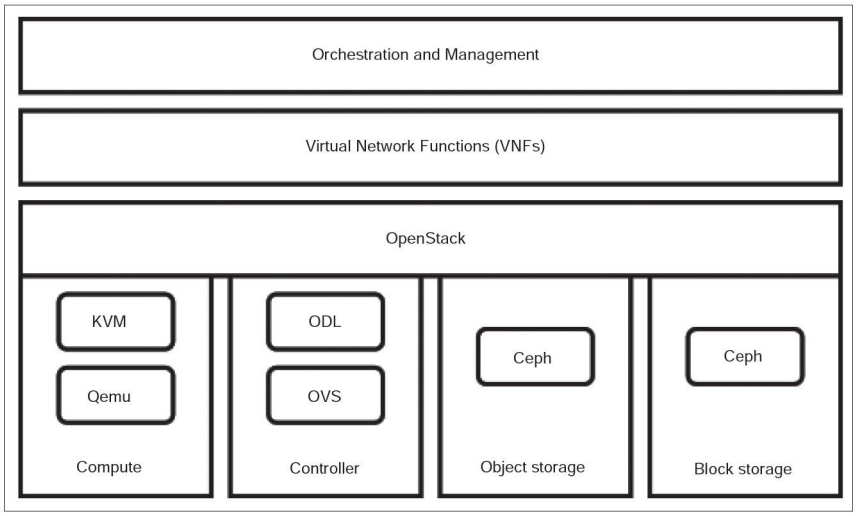


Figure 2: NFV infrastructure management with OpenStack

little too old. Operating system virtualization is a budding virtualization technology and is much more lightweight than that of VMs. Then there are containers that enable microservices, where many physical network functions migrate to virtual network functions (VNFs). As of today, these VNFs are transforming and migrating to cloud native networking functions (CNFs), which are implementing network functionalities that coexist through one or more microservices. CNFs are boosting the telecom industry with cutting-edge network architecture integrated with DevOps processes, aiming for greater service flexibility and a faster delivery mechanism.

Where is Kubernetes in all of this?

We know Kubernetes as an orchestration tool, which automates the entire deployment and scaling procedure of container-based applications. But in relation to NFV, it also covers virtual infrastructure management. In this role, it schedules container-based workloads and jobs while managing the networking between them. Kubernetes can manage virtual network functions by monitoring the health of each container and restarting these when required. Operators can also scale the containers up or down based on the incoming traffic.

The grand scheme of NFV is compelling in cloud native computing. Today, Kubernetes is bringing quite a change in NFV infrastructure, but the evolution from VMs to container-based environments is still a challenge. While the operator might deny it today, surely this is the need of the hour going by the automation, scaling, and monitoring functionalities Kubernetes brings to the table. Without any doubt, this is the path to the future of NFV infrastructure management.

NFV infrastructure is difficult to manage in an environment where developers and operators clash. This can lead to slow movement of NFV applications on the infrastructure. Apart from this, the setting up of such infrastructure is a challenge. DevOps methodologies are a buzz in the market and are known for their automated tools chain. They also help connect developers with the production environment of the Kubernetes cluster and OpenStack cloud services. END

References

- Margaret Chiosi, Don Clarke, Peter Willis, Andy Reid, James Feger, Michael Bugenhagen, Waqar Khan, Michael Fargano, Chunfeng Cui, Hui Deng, et al; Network Functions virtualisation: An introduction, benefits, enablers, challenges and call for action; In SDN and OpenFlow World Congress, volume 48, pages 1–16, 2012
- Ken Gray and Thomas D. Nadeau; Network Function Virtualization; Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1st edition, 2016; ISBN:0-12-802119-5
- Hassan Hawilo, Abdallah Shami, Maysam Mirahmadi, and Rasool Asal; NFV: state of the art, challenges, and implementation in next generation mobile networks (vEPC); IEEE Network, 28(6):18–26, 2014; DOI:10.1109/MNET.2014.6963800.
- OpenStack for Telecom & NFV; *https://www.openstack.org/use-cases/telecoms-and-nfv/; Last accessed: March 1, 2022*
- Russell Bryant, Kathy Cacciatore, Stephen Gordon, Eric Ji, Armando Migliaccio, Iben Rodriguez, et al; Accelerating NFV Delivery with OpenStack; *https://object-storage-ca-ymq-1.vexxhost.net/swift/v1/6e4619c416ff4bd19e1c087f27a43eea/www-assets-prod/marketing/OpenStack-NFV-A4.pdf; Last accessed: March 1, 2022*
- Fabio V; How Kubernetes is Helping NFV Towards Cloud Native; *https://gonorthforge.com/how-kubernetes-is-helping-nfv-towards-cloud-native/; Last accessed: March 15, 2022*

By: Shubham Aggarwal, Nithya Ganesan, and B. Thangaraju

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

Building Amazon Machine Images with HashiCorp Packer

HashiCorp Packer is an open source tool that helps to create identical machine images for multiple platforms from a single source template. This article shows how to use this tool to build Amazon machine images or AMIs.

Image Source: http://www.freepik.com/

Everyone knows that a virtual machine or a VM is not a physical machine. Instead, it is created using software that runs on a physical machine to emulate the functionality of another physical computer. The machine on which the VM runs is called a *host* and the machine it emulates is called a *target*. In other words, we can just launch a VM on a host to get the functionality of a target.

Many organisations have moved their functionality to VMs running on cloud infrastructure since it is hassle-free and works out cheaper. For example, being a popular cloud provider, AWS offers a facility to launch VMs on a vast number of their host machines. This service is called Elastic Compute Cloud or EC2 for short. You can build your entire computing infrastructure just by launching the required VMs on the AWS EC2 service.

To help you in building the VMs, AWS offers a few ready-to-be-launched target machine images or AMIs (AWS machine images). An AMI is a deployment unit that includes an operating system, utilities, and other resources. AWS offers AMIs to build Ubuntu VMs, Windows VMs, etc, to name a few. These are all general-purpose images.

In most cases, you may want to launch a custom VM for your special needs. Such a VM may be based on a specific version of an operating system, with a specific provisioning.

There are two ways of achieving this.

Mutable VM: In this approach, you launch the VM from a suitable AMI, install other utilities, and provision the ports and other resources. AWS offers a nice web interface to accomplish this. However, the problem with this approach is that you have to repeat the process for

all the instances that you want to launch, which is both time taking and error-prone.

Immutable VM: In this approach, you build your own custom AMI with all the special needs and then launch the instance. Once the custom AMI is ready, you can launch any number of instances quickly without any intervention. If you need a VM with another set of requirements, you repeat the process of building another custom AMI and launch the instances. AWS offers special tooling such as AWS CLI for building custom VMs.

```
krishna $ packer
Usage: packer [--version] [--help] <command> [<args>]

Available commands are:
    build           build image(s) from template
    console         creates a console for testing variable interpolation
    fix             fixes templates from old versions of packer
    fmt             Rewrites HCL2 config files to canonical format
    hcl2_upgrade    transform a JSON template into an HCL2 configuration
    init            Install missing plugins or upgrade plugins
    inspect         see components of a template
    plugins         Interact with Packer plugins and catalog
    validate        check that a template is valid
    version         Prints the Packer version

krishna $ packer version
Packer v1.8.6
```

Figure 1: Installing Packer

So far so good, as long as you want all your computing infrastructure only on AWS. However, AWS is not the only player in this market. Other cloud providers like Google Cloud, Azure Cloud, etc, also offer images to launch VMs on their cloud. Your organisation may want to deploy different apps on different clouds.

In such a case, your DevOps team needs to use different tools to build machine images for different clouds. Though it is not impossible to live in such an environment, it definitely calls for a uniform way and toolset to build identical images for different cloud platforms. That's where the Packer tool from HashiCorp comes into the picture.

According to their website, Packer is an open source tool that lets you create identical machine images for multiple platforms from a single source template. All you need is to develop a Packer template using their HashiCorp Configuration Language (HCL) and ask the Packer tool to build an AMI based on the template. In other words, Packer helps you in practising IaaC or Infrastructure as a Code.

This article explains the way to build an image using the Packer tool.

Installing Packer

The Packer tool is available on most of the popular platforms, and installing it is easy and simple.

The following commands install Packer on any Linux platform:

```
curl -fsSL https://apt.releases.hashicorp.com/gpg | sudo apt-key add -
sudo apt-add-repository "deb [arch=amd64] https://apt.releases.hashicorp.com $(lsb_release -cs) main"
sudo apt-get update
sudo apt-get install packer
```

In case your DevOps team is using Mac, then the Homebrew can be used to install Packer. All you need is to run the following commands after installing the latest version of Homebrew.

```
brew tap hashicorp/tap
brew install hashicorp/tap/packer
```

Once Packer is installed, you can verify if the installation is proper by running the following command on the command prompt:

```
packer
```

The result of running the above commands appears in Figure 1. It suggests the correct syntax and lists the available commands.

You may want to try the easiest command:

```
packer version
```

In my case, I found that Packer version 1.8.6 is running on my Mac.

Building an AMI using Packer

To demonstrate the Packer tool, let us build an AWS machine image or AMI. AWS offers a free-tier account, using which you can build AMIs and launch small VMs.

As the first step, visit *https://console.aws.amazon.com/* to create an AWS account and make a note of the security credentials. Packer communicates with AWS using these credentials over SSH. You may either specify the credentials in the Packer template or store them on the machine, which Packer can locate whenever it needs.

Obviously, the latter approach is much more convenient and secure. To this end, use the AWS CLI to store the AWS security credentials securely. For example, use the following command on a Mac to install AWS CLI:

```
brew install awscli
```

```
krishna $ aws configure
AWS Access Key ID [****************624H]:
AWS Secret Access Key [****************50tv]:
Default region name [us-east-1]:
```

Figure 2: Configuring AWS credentials

Once AWS CLI is installed, use the following command to store the AWS security credentials:

```
aws configure
```

The command prompts for AWS access key ID and AWS secret access key like in Figure 2. Supply the details accordingly. You may also supply the default region name.

Packer template

We are all set to write the Packer template. The previous versions of Packer supported JSON for templating. However, the current version recommends using HCL instead. HCL can be used to write templates not only for Packer but also used in other tools of HashiCorp like Terraform.

An HCL template file uses *.hcl* as the extension. In the case of Packer, the file name uses *.pkr.hcl* as the extension. Let us name our template file *ami.pkr.hcl*. The following is the initial template.

```
source "amazon-ebs" "glarimy-ubuntu" {
      region = "us-east-1"
      source_ami = "ami-007855ac798b5175e"
      instance_type = "t2.micro"
      ssh_username = "ubuntu"
      ami_name = "glarimy-ubuntu"
}

build {
      sources = [
        "source.amazon-ebs.glarimy-ubuntu"
      ]
}
```

The above template consists of two blocks, namely, *source* and *build*. An HCL block also contains arguments inside the curly braces. The arguments vary depending on the kind of block.

The Packer *source* block describes the base image from which a new AMI is to be built. In our case, the *source* block specifies the type of source, namely, 'amazon-ebs'. It implies that we are going to use an AMI of EBS type, provided by Amazon. Packer detects this information and tries to interact with AWS. The source block also has a label 'glarimy-ubuntu'. This is just to refer to the block locally in other parts of the template. Together, this block can be referred to uniquely as *source.amazon-ebs.glarimy-ubuntu*.

The *source* block usually expects us to specify the base image and other basic information.

In our case, we specified that *ami-007855ac798b5175e* is to be used as the base AMI and use the t2.micro type of EC2 instance to build the new AMI. We also specified that 'ubuntu' is to be used as the user name for connecting to the instance for further provisioning, if any. Though we specified the region, Packer can actually detect this information along with the AWS security credentials from the local store that we already created. These credentials are required for the Packer to connect to AWS. Finally, the *ami_name* indicates the name that we want to give to the new AMI.

So, in summary, Packer connects to AWS using the security credentials, spins a t2.micro instance, deploys *ami-007855ac798b5175e* on it, and builds the new AMI named *glarimy-ubuntu*. Though the new AMI is in no way different from the source AMI, it is a good starting point.

The other block in our template is the *build* block. It does not have any type or label associated with it. It has only one argument, namely, sources. Though we specified only *source.amazon-ebs.glarimy-ubuntu* as the source, this argument takes a list of any number of sources. Packer builds the AMIs based on these sources.

Building the AMI

Once the *packer* template file is ready, run the following command in the folder where the template file is located. It downloads any plugins that are required.

```
packer init
```

We can also verify if the template file is valid.

```
packer validate ami.pkr.hcl
```

A valid template file can be formatted properly according to HCL style using the following command:

```
packer fmt ami.pkr.hcl
```

Fine! After initialisation and validation, we are all set to build the actual images.

```
packer build ami.pkr.hcl
```

The command given above takes a few minutes to

```
krishna $ packer build .
amazon-ebs.gmi: output will be in this color.

==> amazon-ebs.gmi: Prevalidating any provided VPC information
==> amazon-ebs.gmi: Prevalidating AMI Name: glarimy-ami-1683011997
    amazon-ebs.gmi: Found Image ID: ami-007855ac798b5175e
==> amazon-ebs.gmi: Creating temporary keypair: packer_6450b99d-6d6e-2f2a-eed4-942
==> amazon-ebs.gmi: Creating temporary security group for this instance: packer_64
==> amazon-ebs.gmi: Authorizing access to port 22 from [0.0.0.0/0] in the temporar
==> amazon-ebs.gmi: Launching a source AWS instance...
    amazon-ebs.gmi: Instance ID: i-0f6f68bc8ab42cd69
==> amazon-ebs.gmi: Waiting for instance (i-0f6f68bc8ab42cd69) to become ready...
==> amazon-ebs.gmi: Using SSH communicator to connect: 3.86.224.19
==> amazon-ebs.gmi: Waiting for SSH to become available...
==> amazon-ebs.gmi: Connected to SSH!
==> amazon-ebs.gmi: Stopping the source instance...
    amazon-ebs.gmi: Stopping instance
==> amazon-ebs.gmi: Waiting for the instance to stop...
==> amazon-ebs.gmi: Creating AMI glarimy-ami-1683011997 from instance i-0f6f68bc8a
    amazon-ebs.gmi: AMI: ami-0033e806df9f271e4
==> amazon-ebs.gmi: Waiting for AMI to become ready...
==> amazon-ebs.gmi: Skipping Enable AMI deprecation...
==> amazon-ebs.gmi: Terminating the source AWS instance...
==> amazon-ebs.gmi: Cleaning up any extra volumes...
==> amazon-ebs.gmi: No volumes to clean up, skipping
==> amazon-ebs.gmi: Deleting temporary security group...
==> amazon-ebs.gmi: Deleting temporary keypair...
Build 'amazon-ebs.gmi' finished after 3 minutes 31 seconds.

==> Wait completed after 3 minutes 31 seconds

==> Builds finished. The artifacts of successful builds are:
--> amazon-ebs.gmi: AMIs were created:
us-east-1: ami-0033e806df9f271e4
```

Figure 3: Packer build output

finish as it involves interaction with AWS, spinning a new EC2, deploying the source AMI, and building the new AMI. See Figure 3 to understand the steps involved in the whole process.

In my case, a new AMI with ID *ami-058213aa644fe1b60* in the us-east-1 region is created. It will be listed under the *My AMI* tab on the AWS AMI console page.

Using variables

The Packer HCL template files can also define and use variables for reusability. For example, the *ssh_username* in the following template is referred to a variable named *var.uname*.

```
source "amazon-ebs" "glarimy-ubuntu" {
     ssh_username = var.uname
     ami_name = "glarimy-ubuntu-{{timestamp}}"
     ...
}
...
```

Where is this variable defined and initialised? The variables can be defined in the same template file or in a separate file. The latter is a better approach for obvious reasons. The default name of such a file is *variables.pkr.hcl*. Packer automatically locates this file locally.

In our case, the *variables.pkr.hcl* file has the following block:

```
variable "uname" {
  type = string
  default = "ubuntu"
}
```

As you can see, the file contains a definition for a variable named *uname*. The arguments specify that the *uname* is a string and 'ubuntu' is its default value. You can also supply validation rules and descriptions.

Once the variables are defined, you can initialise them. There are multiple ways to do this. You may use environmental variables, command line arguments, etc. In our case, we don't do anything as we are happy with the default value.

Another interesting argument in the above *source* file is the way we name the new AMI. Instead of giving a static name like 'glarimy-ubuntu', we are using the current timestamp as the prefix. That way, every time Packer runs this template as part of the CI/CD routine, it creates a new AMI with a unique name, which we can refer to later.

Validating, formatting, and building the above template should happen smoothly. Note that the command syntax to build the image should be changed to the following, as Packer needs to locate multiple files for building.

```
packer build .
```

In the above, we replaced the template file name with a dot to tell Packer to locate all the *.pkr.hcl* files in the current folder.

Selecting base AMI dynamically

Another feature of Packer is that it can dynamically locate the source image from a named data source. For example, our templates so far specify a particular AMI with ID *ami-007855ac798b5175e* as the source. This may not always be helpful, especially in automated CI/CD processes. What if you want to pick up the latest Ubuntu AMI automatically as the source, no matter what its ID? Let's enhance our template for that purpose.

```
data "amazon-ami" "ubuntu" {
     region = "us-east-1"
     filters = {
       virtualization-type = "hvm"
       name = "*ubuntu-xenial-*"
       root-device-type = "ebs"
     }
     owners = ["amazon"]
     most_recent = true
}

source "amazon-ebs" "glarimy-ubuntu" {
     source_ami = data.amazon-ami.ubuntu.id
     ...
}
...
```

The revised template has a new block named *data*. It defines a data source. We specified 'amazon-ami' as the data source type and 'ubuntu' as the label. This block uses several filters such as virtualization-type, name expression, and root device type. You may use any number of filters to narrow down the search. Packer will be connecting to the AWS AMI Registry to find all matching AMIs. The argument 'most-recent=true' selects the latest AMI among them and returns its ID and other information.

In the *source* block, we are referring to the selected ID dynamically with the following argument:

```
source_ami = data.amazon-ami.ubuntu.id
```

Run this template to see the results. Here are a few output lines of interest for us.

```
Prevalidating any provided VPC information
Prevalidating AMI Name: glarimy-ubuntu-1683014188
Found Image ID: ami-0b0ea68c435eb488d
```

You can observe from the above output that Packer found *ami-0b0ea68c435eb488d* as the source image matching our filters.

Customising the AMI

So far, we have just cloned the source AMI to build our own AMI. But this is not of good use. The real reason for us to use Packer is to build custom images. We have already talked about the immutable VMs. In this approach, we build a new AMI with all the required provisioning.

As you know, Ubuntu does not come with a Docker engine installed. Let us make our new images Docker-ready. For this, we need to add provisioning in the template file. Keep the *data* block and *source* blocks as they are, and edit the *build* block as follows in the *ami.pkr.hcl* file.

```
build {
     sources = [
       "source.amazon-ebs.glarimy-ubuntu"
     ]
     provisioner "shell" {
       inline = ["while [ ! -f /var/lib/cloud/instance/boot-
finished ]; do echo 'Starting'; sleep 1; done"]
     }
     provisioner "shell" {
       scripts = ["docker.sh"]
     }
}
```

As you can observe, the *build* block has two *provisioner* blocks of type 'shell'. It tells the Packer to run certain *shell* commands after deploying the source AMI on the instance and before creating the snapshot for the new AMI.

The *shell* commands can be supplied inline or in a separate file. For example, in the above template, the first provisioner is supplied with an inline command. This command essentially waits till the booting is complete so that we can run the other commands.

The second provisioner is supplied with a file named 'docker.sh'. A file with that name must be present for the Packer to run. The content of the file has commands to install Docker.

```
sudo apt update
echo "Y" | sudo apt install docker.io
```

Running the above template gives an output from which certain lines are presented here.

```
Prevalidating any provided VPC information
Prevalidating AMI Name: glarimy-ubuntu-1683016633
Found Image ID: ami-0b0ea68c435eb488d
Launching a source AWS instance...
Instance ID: i-07c012aa3a2bcaee2
Waiting for instance (i-07c012aa3a2bcaee2) to become ready...
Using SSH communicator to connect: 3.90.36.190
Waiting for SSH to become available...
Connected to SSH!
Provisioning with shell script: /var/folders/ly/02f312xx3q5_
gqwlk82m1c7h0000gn/T/packer-shell1060700513
Starting
Starting
Starting
Starting
Provisioning with shell script: docker.sh
Unpacking docker.io (18.09.7-0ubuntu1~16.04.7)
Setting up docker.io (18.09.7-0ubuntu1~16.04.7)
```

```
Adding group `docker' (GID
116)
Done.
Builds finished. The artifacts
of successful builds are:
AMIs were created:
us-east-1: ami-
05f11967dcc85276f
```

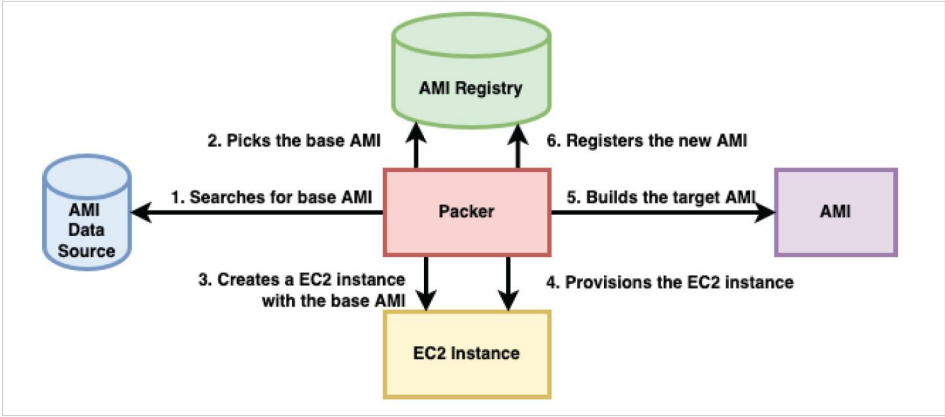


Figure 4: Packer build process

As you can see, both the shell provisioners are run before building the AMI. Also, it can be found from the output that Packer logged into the instance using SSH, and ran the scripts to install *dockers*.

We can verify if the provisioning is done properly. For that, log in to the AWS console, launch an EC2 instance using the newly built AMI with a key pair, and note down the public IP address. Download the key pair as a *.pem* file onto the local machine.

Run the following commands. Replace the *.pem* file and IP address appropriately. In my case, the *.pem* file is named *glarimy*, and the IP address is 52.23.169.102.

```
chmod 400 glarimy.pem
ssh -i glarimy.pem ubuntu@52.23.169.102
sudo docker images
```

Notice that the newly launched EC2 instance is equipped with a Docker engine and you can interact with it.

Handling the artifacts

Well! We have covered good ground. We now know how to select a source AMI, how to clone it, how to provision it, and how to create a new AMI. In the Packer jargon, the newly generated files such as AMIs are referred to as artifacts. Packer also helps in handling these artifacts. For instance, Packer can help you in registering an AMI with a different registry, or move to a different place, etc. Such artifact handling is achieved using *post processors*.

In the following example, it simply creates a file on the local machine with the details of the newly created artifact.

Add a *manifest post-processor* to the *build* block in the *ami.pkr.hcl*. This entry directs the packer to create a file named 'glarimy-ubuntu.json' with the manifest details of the newly built AMI artifact.

```
build {
    ...

    post-processor "manifest" {
        output = "glarimy-ubuntu.json"
    }
}
```

Running the updated template generated the following manifest in my case, in the *glarimy-ubuntu.json*.

```
{
    "builds": [{
      "name": "glarimy-ubuntu",
      "builder_type": "amazon-ebs",
      "build_time": 1683021300,
      "files": null,
      "artifact_id": "us-east-1:ami-0903822880fb16d20",
      "packer_run_uuid": "9f755e3c-7321-679d-
8f95-f57256125039",
      "custom_data": null
    }],
    "last_run_uuid": "9f755e3c-7321-679d-8f95-f57256125039"
}
```

In this article, we have seen the usage of Packer in building an AWS AMI. The source code used in this article can be found at *https://bitbucket.org/glarimy/glarimy-university/src/master/glarimy-packer*. The whole process is summarised in Figure 4.

HashiCorp Packer is a powerful and easy-to-use tool to build identical images targeting multiple platforms. Using similar steps, images can be created in Azure, Google Cloud, and Docker, to name a few. END

By: Krishna Mohan Koyya

The author is the founder and principal consultant at Glarimy Technology Services, Bengaluru. He has mentored and upskilled more than 250 technical teams in architecting, designing, and developing enterprise applications on-premises as well as on cloud.

R N I No. DELENG/2012/49440, Mailed on 27/28th of Advance month
Published on 27th May 2023
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