IT Service Systems for Enterprises

Information Technology Infrastructure Library (ITIL) (Advised Skills, 2023)

Diving deeper into Information Service Systems Enterprises or ITIL, it is significant to highlight where and how IT Service Management (ITSM) differs from ITIL.

Information Technology Infrastructure Library (ITIL) is considered the most popular framework centered on delivering IT services. It has become the standard for organizations to use as it aims to provide guidance on how services must be defined, developed, built, and operated. Microsoft, IBM, and Hewlett-Packard Enterprise are known to be using ITIL for their internal operating guidelines.

ITIL Processes

In V3, it is said that ITIL is made up of processes and functions. A **process** is a structured set of activities made to accomplish specified objectives and takes different inputs to turn them into defined outputs. It can be envisioned as a set of activities performed one after the other to achieve something.

Such as in cooking, some steps need to be followed to get the desired dish. The main aspect of the process is the interconnectivity between these individual steps, without interchanging, in working toward a common goal.

ITIL Functions

In ITIL V3, a **function** is described as a team or group of people and the tools and resources they use to conduct processes or activities. These teams could be the networking team, the Java team, the web development team, or any group of people that are selected based on their depth of knowledge on a specific specialty.

In ITIL, there are 26 processes and only four (4) functions. Processes do not run by themselves and need people to perform the individual process activities that come from functions. Simply put, functions give the resources needed by the processes to complete their objectives.

ITIL Practices

But in ITIL 4, processes and functions are replaced by practices. A process takes in inputs and when the trigger kicks in, an output is achieved. **Practice** can be considered an extension of a process as it not only defines the activities but also brings various capabilities and tools together to accomplish a set of objectives.

ITIL 4 fused functions with processes to produce the concept of practices. For instance, a problem management team in an organization is considered a function in ITIL 3, and they work on problem management processes to meet their objectives. In ITIL 4, it becomes problem management practice which is a system whose objective is to deliver all problem management outputs. This is changed in ITIL 4 to collaborate better and to deliver value efficiently.

ITIL Principles

ITIL has seven (7) guiding principles which are a set of recommendations for ITSM that help organizations make smart decisions and take actions aligned with their goals.

- **Focus on Value**: Everything the organization does must always bring value to its stakeholders, specifically the customers. The services should be designed with a customer's satisfaction in mind. It is key to determine what the customers want to precisely match it with a value they will appreciate. But this is something that changes over time that can benefit an organization to grow with their customers.

- **Start Where You Are**: This does not mean resetting everything from the start. This means focusing on what is already available and improving that existing system. Existing systems already have value on their own, so it is best to assess their strength and use that as the starting point. Focus on optimization and improvement as much as possible.

- **Progress Iteratively with Feedback**: This reminds the people involved not to accomplish everything at once, as it can be overwhelming. Divide tasks into smaller parts and focus on each step to assess the value it can provide. Getting feedback after every step is also advised to ensure problems are solved, and oversights are reworked before it is too late. Progressing iteratively with sustained evaluation assures results are useful, tangible, and completed promptly.

- **Collaborate and Promote Visibility**: This aims to take advantage of the different sets of strengths and weaknesses that each member of the IT team has through delivering collaborative efforts. It is best to prioritize honesty and visibility to collaborate effectively. Communicate each progress, whether good or bad and allow to be open to whatever feedback others might give.

- **Think and Work Holistically**: To create the best service, it all must be connected, as no service, practice, process, or function stands alone. All activities should align with the same focus on the delivery of value. This is critical when working in a collaborative environment to ensure that the work does not affect any part of the organization and create any hindrance or disruption.

- **Keep it Simple and Practical:** This is a reminder to not over-complicate anything to be done. Complicating any process increases the risks of going wrong and having more errors. Continuously assess the work and ensure that only the necessary steps are involved. Eliminate any process, step, or action that does not add value to the organization or the customer.

- **Optimize and Automate**: Optimization makes things functional, so automation of any activity should be optimized first to utilize the resources to the best of their limits. Automation relieves the work on frequent and repetitive tasks. The human resources that could have been working on those tasks can be put into more complex tasks that contribute to the value.

The organization cannot only choose one or two principles to apply but must consider the relevance of all principles and how they can be applied together in their ITSM practices. This involves reviewing and updating existing processes, training employees on the principles, and using the principles to guide better decision-making and action-taking.

DevOps (Kaiser, 2023)

In the IT industry, ITIL and DevOps are often pitted against each other, and one must only choose between the two, but they are not mutually exclusive. They can be complementary approaches with their benefits.

DevOps is the combination of **dev**elopment and **op**eration teams to deliver an objective. It brings people together from across disciplines to collaborate as a unit with an open mind and to remove inefficiencies.

In DevOps, the responsibility of accomplishing a task is not considered an individual responsibility, but a shared one. Wherein if the person in charge fails or succeeds, the entire team gets the consequence or the praise. This creates a **blameless culture** that acknowledges the fact that humans make mistakes. Amazon, Etsy, Netflix, and Target are known to use DevOps in their organization.

Elements of DevOps

The three (3) elements (PPT) support and enable DevOps:

1. **People** – the heart of DevOps, which is composed of a *DevOps team* consisting of a pool of tooling professionals. A DevOps team is built around an application; if an application is being developed, all the people responsible for its development and operations must be grouped to create a single team. For example, Application Z is an Internet banking program for individuals and small business owners. The possible DevOps team could be the following:
   - Product Owner – the owner of the product and service backlog
   - Developer – for coding and unit testing
   - Tester – involved in developing test scripts
   - Architect – designs the software and are shared across multiple DevOps teams
   - IT Security – for managing aspects of IT security

2. **Process** – The CI/CD pipeline consisting of Continuous Integration, Continuous Delivery, and Continuous Deployment is used for effective DevOps Methodology. This is a set of DevOps best practices to help teams deliver software faster and more efficiently.
   - **Continuous Integration** – provides a central repository for developers to use throughout a project. This automates anything in the central repository to reduce errors and compile artifacts into organized builds to perform initial tests.
   - **Continuous Delivery** – delivers the builds into the test environment and then needs human intervention to deploy the tested builds into production. It aims to get code changes like fixes or configuration changes into production faster and safely.
   - **Continuous Deployment** – performs beyond Continuous Delivery and deploys automatically to production. This is viewed as a build on the established practices of Continuous Delivery, but this focuses more on a complete end-to-end automated approach, while Continuous Delivery requires human intervention.

3. **Technology** – often regarded as the most important of processes. There are several tools that claim to support DevOps, see *Figure 1.* However, choosing the right tools is just as crucial since not all tools can be used for all technologies.

For example, for the source code repository function, Git and Subversion can be used as both provide a versioning capability for the source code and can be integrated with other toolsets for automation. Looking deeper, Subversion is under the Central Version Control System (CVCS) category, while Git is under the Distributed Version Control System (DVCS).

CVCS has a server-client relationship to store and retrieve the code, and the developer is required to first check out the existing code, make changes, and check it back in. In **DVCS**, each developer has the entire code on their computer that requires no check-out and check-in, allowing multiple developers to work seamlessly. As the source code is locally available in DVCS, accessing, adding, combining, and delivering the code is faster than in CVCS.

It is important to weigh and compare the capabilities and compatibilities of each tool before choosing one.

DevOps Principles

DevOps principles are everchanging, but these are the most widely believed set of principles, known as CALMS:

- **Culture**: Culture cannot be changed in an instant as it is embedded into human behavior and would require an overhaul to do so. But there are some behavioral traits that DevOps seeks to change:
  - Taking responsibility for the entire product and not just the task performed
  - Stepping out of the comfort zone to innovate
  - Communication, collaborating, and developing an affinity with the teams
  - Experimenting as much as possible

- **Automation** – an enabler of faster delivery and crucial for providing rapid feedback. Repetitive tasks that do not require human intelligence are the kinds of activities that can be efficiently automated. For instance, building infrastructure before was a major task for hardware architects and administrators and building servers took a significant amount of time. Now, cloud infrastructures are much utilized, and servers can be created through coding.

- **Lean**: DevOps borrowed from Lean technology which is about keeping things simple and not overcomplicating them. This allows the teams to focus and work on things that are easy to comprehend and simple to work with. The first part of the Lean principle addresses that the primary issue is not to bloat the way to do things, just do them straightforward and minimally. The second part is to reduce waste from the methodology such as defects and convoluted processes. Defects and convoluted processes delay the overall delivery, and it takes much more time and money to address and fix them.

- **Measurement**: The only way to find out whether the outcome is favorable is by measuring it. This principle provides direction about the measures to implement and keep tabs to feel the reception of the overall software delivery. Feedback could be used as a measurement to automate processes. Measurements in monitoring such as in applications and infrastructure, can indicate when an event is a warning or an exception.

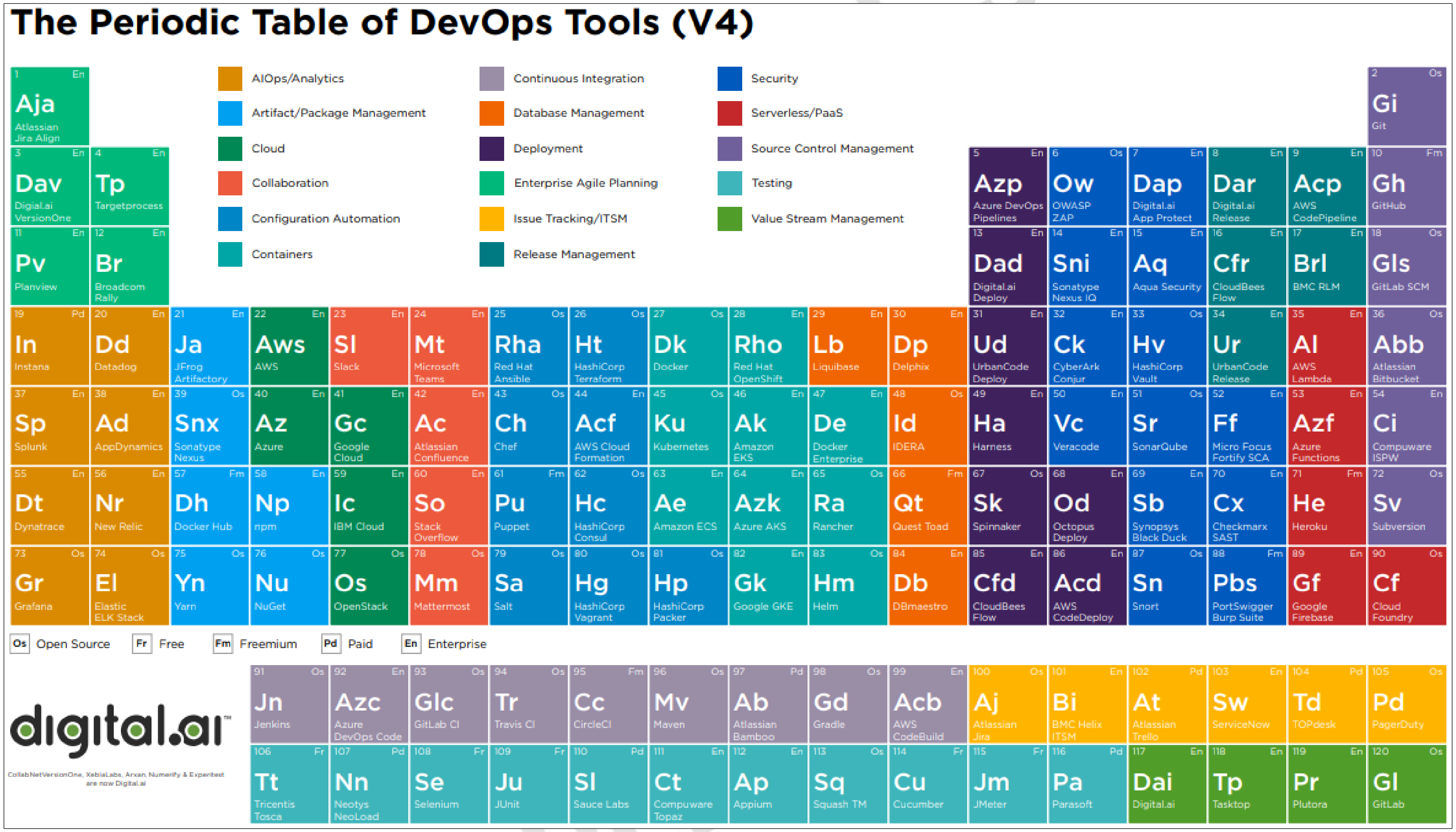
- **Sharing**: This is the need for collaboration and knowledge sharing between people. The shared knowledge, experience, ideas, and thoughts can help others in making them better and more profound. This leads to collaboration instead of competition and skepticism. In DVCS, accessing, adding, combining, and delivering the code is faster compared to CVCS.

References:

Advised Skills (2023). *ITIL 4 guiding principles explained.* [Web Article]. Retrieved on July 31, 2023, from https://www.advisedskills.com/about/news/213-itil-4-guiding-principles-explained

Kaiser, A. (2023). *Reinventing ITIL and DevOps with digital transformation*. Apress.

Wilkes, A. (2022). *The people, process, and technology that make DevOps effective.* [Web Blog]. Retrieved on July 31, 2023, from https://www.launchableinc.com/blog/the-people-process-and-technology-that-make-devops-effective/



The Periodic Table of DevOps Tools (V4)

- AIOps/Analytics
- Artifact/Package Management
- Cloud
- Collaboration
- Configuration Automation
- Containers
- Continuous Integration
- Database Management
- Deployment
- Enterprise Agile Planning
- Issue Tracking/ITSM
- Release Management
- Security
- Serverless/PaaS
- Source Control Management
- Testing
- Value Stream Management

1 En Aja Atlassian Jira Align																	2 Os Gi Git
3 En Dav Digial.ai VersionOne	4 En Tp Targetprocess											5 En Azp Azure DevOps Pipelines	6 Os Ow OWASP ZAP	7 En Dap Digital.ai App Protect	8 En Dar Digital.ai Release	9 En Acp AWS CodePipeline	10 Fm Gh GitHub
11 En Pv Planview	12 En Br Broadcom Rally											13 En Dad Digital.ai Deploy	14 En Sni Sonatype Nexus IQ	15 En Aq Aqua Security	16 En Cfr CloudBees Flow	17 En Brl BMC RLM	18 Os Gls GitLab SCM
19 Pd In Instana	20 En Dd Datadog	21 En Ja JFrog Artifactory	22 En Aws AWS	23 En Sl Slack	24 En Mt Microsoft Teams	25 Os Rha Red Hat Ansible	26 Os Ht HashiCorp Terraform	27 Os Dk Docker	28 En Rho Red Hat OpenShift	29 En Lb Liquibase	30 En Dp Delphix	31 En Ud UrbanCode Deploy	32 En Ck CyberArk Conjur	33 Os Hv HashiCorp Vault	34 En Ur UrbanCode Release	35 En Al AWS Lambda	36 Os Abb Atlassian Bitbucket
37 En Sp Splunk	38 En Ad AppDynamics	39 Os Snx Sonatype Nexus	40 En Az Azure	41 En Gc Google Cloud	42 En Ac Atlassian Confluence	43 Os Ch Chef	44 En Acf AWS Cloud Formation	45 Os Ku Kubernetes	46 En Ak Amazon EKS	47 En De Docker Enterprise	48 Os Id IDERA	49 En Ha Harness	50 En Vc Veracode	51 Os Sr SonarQube	52 En Ff Micro Focus Fortify SCA	53 En Azf Azure Functions	54 En Ci Compuware ISPW
55 En Dt Dynatrace	56 En Nr New Relic	57 Fm Dh Docker Hub	58 En Np npm	59 En Ic IBM Cloud	60 En So Stack Overflow	61 Fm Pu Puppet	62 Os Hc HashiCorp Consul	63 En Ae Amazon ECS	64 En Azk Azure AKS	65 Os Ra Rancher	66 Fm Qt Quest Toad	67 Os Sk Spinnaker	68 En Od Octopus Deploy	69 En Sb Synopsys Black Duck	70 En Cx Checkmarx SAST	71 Fm He Heroku	72 Os Sv Subversion
73 Os Gr Grafana	74 Os El Elastic ELK Stack	75 Os Yn Yarn	76 Os Nu NuGet	77 Os Os OpenStack	78 Os Mm Mattermost	79 Os Sa Salt	80 Os Hg HashiCorp Vagrant	81 Os Hp HashiCorp Packer	82 En Gk Google GKE	83 Os Hm Helm	84 En Db DBmaestro	85 En Cfd CloudBees Flow	86 En Acd AWS CodeDeploy	87 Os Sn Snort	88 Fm Pbs PortSwigger Burp Suite	89 En Gf Google Firebase	90 Os Cf Cloud Foundry
			91 Os Jn Jenkins	92 En Azc Azure DevOps Code	93 Os Glc GitLab CI	94 Os Tr Travis CI	95 Fm Cc CircleCI	96 Os Mv Maven	97 Pd Ab Atlassian Bamboo	98 Os Gd Gradle	99 En Acb AWS CodeBuild	100 Os Aj Atlassian Jira	101 En Bi BMC Helix ITSM	102 Pd At Atlassian Trello	103 En Sw ServiceNow	104 Pd Td TOPdesk	105 Os Pd PagerDuty
			106 Fr Tt Tricentis Tosca	107 Pd Nn Neotys NeoLoad	108 Fr Se Selenium	109 Fr Ju JUnit	110 Pd Sl Sauce Labs	111 En Ct Compuware Topaz	112 En Ap Appium	113 Os Sq Squash TM	114 Fr Cu Cucumber	115 Fr Jm JMeter	116 Pd Pa Parasoft	117 En Dai Digital.ai	118 En Tp Tasktop	119 En Pr Plutora	120 Os Gl GitLab

Os Open Source   Fr Free   Fm Freemium   Pd Paid   En Enterprise

digital.ai™

CollabNetVersionOne, XebiaLabs, Arxan, Numerify & Experitest are now Digital.ai

Figure 1. DevOps tools periodic table. https://digital.ai/learn/devops-periodic-table