Quick guide

DevOps Platform

Eficode ROOT

For management

03 - 14

For experts

15 - 27

DevOps Platform

For management

in a nutshell

With ever-digitalizing world, the ability to react to innovation and customer needs has emerged as a significant success factor.

This requires transformation that can be achieved by combining practices, tools, and environments into one manageable ecosystem.

Quick guide for management

DevOps Platform

5

Everything is Code

The opportunities provided by digitalization and automation change the field of operations for businesses with ever-increasing speed. The technical superiority of products is no longer a significant competitive factor as companies must be able to fulfill their customers' needs more comprehensively than previously – using new channels and operational models.

The speed of change has increased and the ability to react to customer needs has become a significant success factor in many fields. This change is reflected by the fact that many companies engaging in traditional sales of products or devices are following the example of recent international success stories such as Airbnb and Uber in producing new, customer-oriented services.

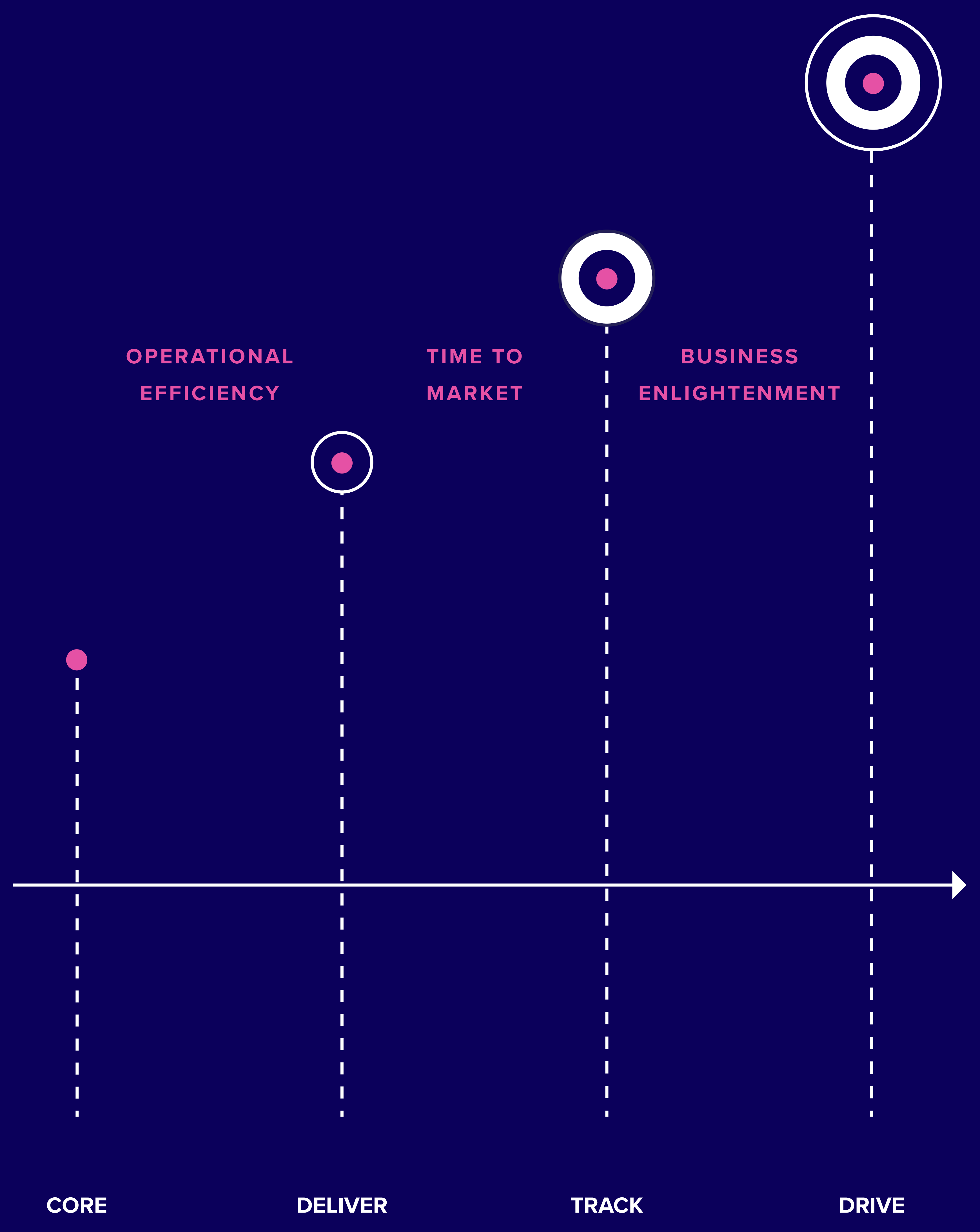
The ability of a business to respond to and realize the evolving needs of customers is directly dependent on the level of software and IT expertise of the company.

Software development and its implementation in a reasonable manner has been transformed from a function fulfilling the organization's internal needs into a business operations enabler that is the key to the development of new competitive advantages and the ability to react quickly to a changing market.

The traditional product development model where software production is managed using heavy processes and produced as a comprehensive project is becoming history. It has been replaced with models where companies can build easily scalable software and microservices which focus on realizing limited responsibilities in an agile and efficient manner. This enables bringing new, lightly structured business operation models and ideas to the market quickly and simultaneously making it possible to validate their functionality and profitability to the end customers.

This new model is based on the seamless cooperation of software development, IT management, and business operations. DevOps focuses on building and constantly developing this cooperation. **DevOps enables agile, high-quality software production that corresponds to the customers' needs and enables highly automated operations.** It is a collection of tools and methods that can be used by companies to acquire a real-time view of their business operation. It shows how customers come in contact with the company's products and how the various functions of the company can respond to the needs created by these encounters. This comprehensive operational model enables the easy identification of problematic areas and helps the business in decision-making and the creation of a successful strategy. **A company managed through data is quicker and better prepared to respond to the changes in the competitive field and to create completely new revenue generation models in different value chains.** All of this is based on collecting and analyzing all available information on the company's operations and its customers' behaviors as a continuous operation for decision-making.

Companies must get used to the idea that improving existing business operations and creating new ones is a continuous process that can be developed. The best operators in the field adopt the models and practices that enable continuous development and want it to become a part of the company's culture. It is important to note that customer-oriented service development based on constant experimentation defines the tools with which the customers' problems can be solved. **A modern and successful company is constantly trying out and implementing new technologies and models, simultaneously rejecting those that do not provide adequate added value for the customer.** ■



The demands of business operations for more agile and decentralized digital service development create increased challenges for IT operations.

Environments have become complicated entities. The number of services, products and users has increased and the range of available tools has diversified. Management of this complexity, in practice, requires controlled implementation of requirements.

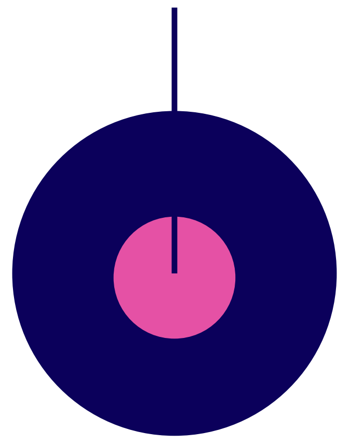
Digital business operations based on constant and effective DevOps have three phases: Deliver, Track, and Drive.

Quick guide for management

DevOps Platform

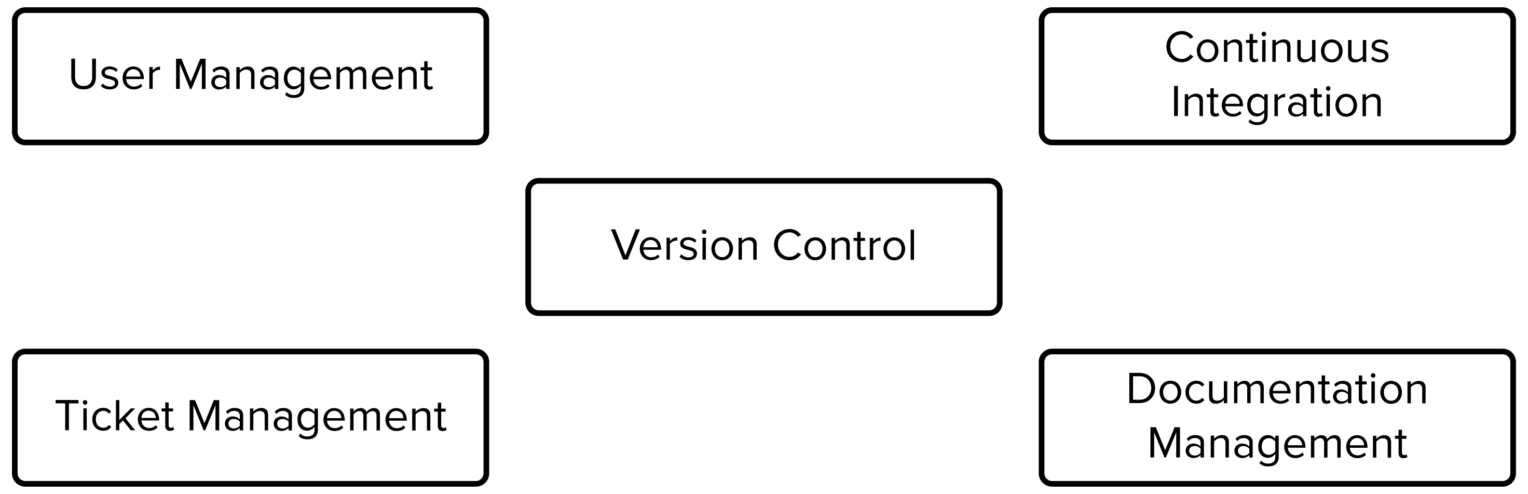
LEVEL 1

CORE



DELIVER

Operational Efficiency



LEVEL 1

Operational efficiency requires the company to standardize the code under development, the related documentation, and work management, combining them behind centralized user IDs. The necessary development practices are harmonized and are monitored automatically between the various tools.

This creates a sturdy base for the next phase of the DevOps journey and enables increasing the efficiency in a controllable way.

At this stage, monitoring the development of a single product is typically very project-specific; the retrieval of historical information is largely based on the user's activity.

DELIVER

The first step of the journey begins with harmonizing the product development tools into one manageable entity, i.e. the DevOps environment. The idea is to collect and share the right kind of information and to implement it not for individual developers but for all the stakeholders who need the information.

When the tools used in the various stages of product development have been effectively linked with each other, the actualization of needs required for delivery can also be monitored efficiently. This will avoid the pitfall of managing development merely by relying on communication between individual persons. In the same way, the completion of features and the time used for development can also be monitored in a controlled manner using the information gathered reliably in the DevOps environment. **Achieving operational efficiency means the ability to provide new products to the market predictably and reliably according to the business operations' and customer needs.**

Benefits

For those in management roles, the described environment provides the ability to manage users of different tools in a centralized manner. **This means that the monetary initial expenses, personnel changes, and other similar costs can be minimized easily.** When the projects follow the harmonized approach, finding, managing, and monitoring the added value produced will no longer be a challenge.

The developers and testers can focus on the essential when the approach is clear and logical – regardless of the project. They receive constant feedback on their work from test-, build-, and deployment automation, thus being able to direct their work to be more productive. Automation increases efficiency and handles the non-standardized, previously manual phases, making it easier to monitor and improve the quality of a software project. When quality deviations are exposed, it is easy to automatically restore earlier versions to production.

CASE

A Government Institution

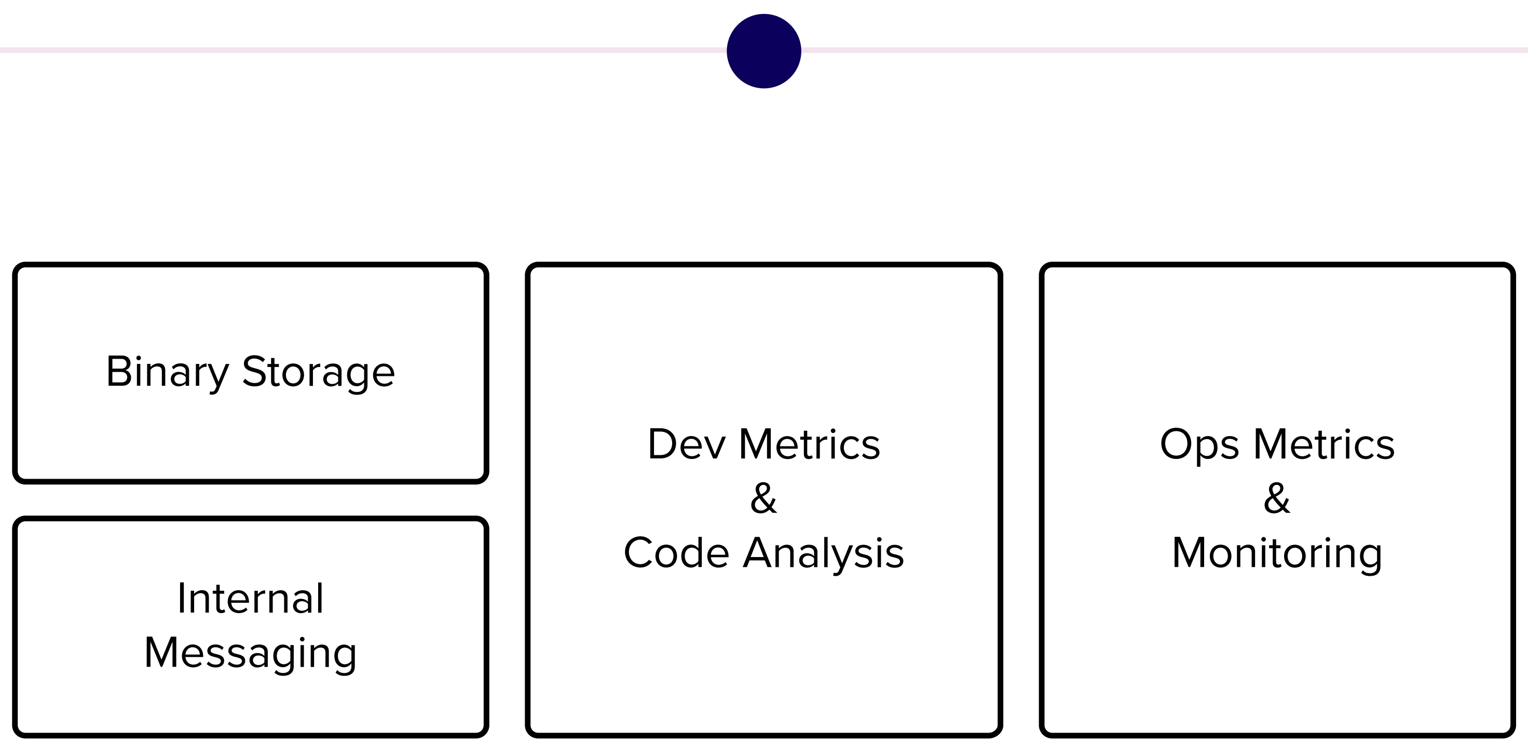
Our customer utilizes a wide variety of highly specialized software, the development of which has been outsourced to partners. **Hoping to increase the efficiency and agility of their operation, they noticed that constantly delivering attractive digital services demands that the basic tools which are used for different projects do not significantly deviate from one another.**

To expedite the transfer into the DevOps culture, the institution has started to offer its partners a platform that encourages them to develop mutually compatible methods and enables the efficiency and automation of operations. The platform provides tools that correspond to the DevOps methods and user management for all of the institution's projects.

For the customer, the environment provides the opportunity to direct the development of software development methods in their organization and to easily monitor the added value produced by their projects. The environment provides better visibility to track project progress and creates cost savings when compared to decentralized tools. As the owner and buyer of the services, the institution wants to invest in the transparency and quality of software development - the new platform enables this throughout the entire organization and purchasing chain.

TRACK

Time-to-Market



When aiming for faster turnaround times, the adopting company will also add binary management into its delivery process, enabling simultaneous management of dozens of environments.

The information from development and production is collected real time and using different metrics based on the information enables the development of business operations which in turn enables operational level enhancements.

As the speed of development increases, the amount of needed communication must be reduced by using visible metrics and automation.

TRACK

In the second phase, indicators for the success rate of the organization's core business operations as well as risk management are created based on the information collected in the DevOps environment. In measuring, manually collected reports are replaced by the utilization of real-time information. Because the DevOps environment acts as a basis for quality and delivery, the collected information precisely shows how efficient the operations are.

New tools are introduced to facilitate the management of larger and more complicated entities. This makes it possible to automate deployments of services and enable continuous delivery, even when the services consist of several smaller systems. **For business operations, the result is faster time-to-market meaning improved ability to react to changes in the market and customer needs.**

Benefits

For business representatives, the described environment provides tools that can be used to measure development, production, and the reactions of the projects' end users. **Utilizing various measuring methods, this information can be used to improve and direct operations.**

In turn, the developers and testers are able to manage wider software systems and focus on new features instead of complex build and deploy processes. Real-time feedback will help them allocate tasks better and provides the opportunity to develop their expertise.

The quality and safety of products can be improved when the implemented metrics show in real time what is actually happening in the software and where the next potential problem or target for development is.

CASE

A Nordic Bank

Our customer has recognized that the success of a new service often depends on how quickly and how high-quality the first product version can be delivered to the end users for validation.

Earliest possible release means that the direction of development can be steered more effectively and the risk of failure is diminished – especially when development is still in its early stages.

For this reason, all the components of Java based software production from code delivery to quality assurance and documentation are performed in an environment shared by all parties. Because more than 3,000 programmers around the world use the software delivery system at the same time, the service must function seamlessly and efficiently. The environment contains the SVN and Git version control services, a requirements management system, a documentation and identity management system, CD servers and artifact repository - all as a unified platform.

The centralized platform enables quick development of work practices and increasing automation. This also means that various integrations and automations can be taken into use without having to bother the administrators. At the same time, the environment provides good visibility to the projects, a quick feedback cycle, and increases the flexibility of development processes. The environment is available 99.95 percent of the time, independent of the time or day of the week.

LEVEL 3

DRIVE

Business Enlightenment

- Service Desk
- Cloud
- Big Data
- Virtualisation
- Environments on-demand
- Containers
- Test Data Management

LEVEL 3

To innovate new business operations, the adopting company's software and IT production virtualizes the environments in use, with server installations being completely automatic. The company has developed its IT structure to a level where it operates on the basis of experiment-driven culture.

A company at this level is usually a leader in its field creating new ecosystems, taking over new markets, and revolutionizing the field's prevalent operational approaches and models. It can be said that the company has reached business enlightenment.

DRIVE

When software development and IT work together seamlessly, the company is able to try out the latest technologies risk-free and create new business. This could, for example, mean implementing new cloud services or solutions based on Big Data. **Virtualizing software infrastructure enables the automatic re-creation of environments and provides the company the opportunity to take its service to the same level with Google, Apple or Amazon.**

Benefits

The harmonized, centralized management of environments and tools ensures the predictability of costs, maintaining service level agreements, the development of internal processes, and facilitates resourcing. **Decision-makers can then focus on the customer's needs, the project, and the team instead of using time and resources on searching for essential information.**

In turn, the developers have all the necessary equipment to solve any technical issue efficiently: code reviews, automatic quality assurance, static code analysis, various other metrics, and on-demand development environments facilitate the development work when the routine tasks have been completely automated. The same benefits are also provided to the testers. They can freely choose the most efficient tools to standardize the acceptance-, integration-, safety-, and performance testing and generate reporting automatically.

Comprehensively testing a project also increases its security in addition to quality. The focus is shifted from evaluating vulnerabilities and their criticality to preventing them with good practices. Static code analytics, automated deployment processes, and scheduled automatic security tests enable meeting the agreed upon levels. **Virtualized and re-creatable environments mean that managing various subcontractors is possible, thus decreasing vendor lock-in.**

CASE

Case You?

How many businesses operating on the global level, as mentioned above, are there? One of the most significant factors is the inability to get acquainted with and to adopt emerging technologies that could be used to create new business operations – like the big players do. This is caused by the fact that we are still living in a world where IT and software development are separate departments, just like any other unit. This silo mentality prevents effective communication and modern innovations.

The delivery time of new services and products to the market is measured in weeks so that their functionality can be validated for the end users and customers. After this, further development of the product is directed based on feedback and the release of new features is prioritized. This cannot be achieved if we have to wait for interdepartmental meetings. Instead, each project team should be cross-functional having members from all areas. Competency can be added to the project team with new members as needs emerge, or decreased when some expertise is no longer needed, scaling dynamically.

The IT infrastructure is an essential part of making the projects described above possible. The utilization of third-party services becomes essential so that time does not have to be spent on tinkering with equipment investments. At the same time, the fulfillment of the regulations and legal requirements should, however, be ensured.

The implementation of new business operations models, the customers' changing service models, and the building of ecosystems requires the seamless harmonization of IT systems. A unified, sensible, and manageable solution is necessary for the company's ability to compete. Companies must make decisions: how is this implemented and who to choose as a partner?

Eficode ROOT provides an excellent solution to companies who want to focus on the development of their core business and partner up with a leading operator in the field. By outsourcing the development environment you can decrease costs with the implementation of the newest tools and practices and provide the opportunity to increase turnover in the form of new services faster and more reliably. Our solution provides all the modern tools in one place and the possibility to customize the service according to the buyer's needs. You can order the Eficode ROOT as a dedicated private cloud service or alternatively installed behind your company's firewall. The typical delivery time is approximately 4–6 weeks.

The practices, cultures, and organizational structures of companies vary from company to company and their objective is to respond to the customers' needs in the best possible way. Therefore, it is important that centralized management of tools and environments is introduced into the organization in a controlled manner, taking its practices into consideration. **It is recommended that Eficode ROOT is first implemented only in selected product or service sectors, depending on the receptive ability of the organization.** The functionalities of Eficode ROOT have been built to make traversing the DevOps transformation journey a part of the natural development, taking the starting level of the organization into consideration. This enables constant up-to-date monitoring for the deployment of the DevOps policies within the organization.

After the implementation, management has an immediate access to the essential parameters from a ready-made dashboard which shows the most important information and numbers for individual projects throughout their entire development time. At the same time, the controlled transfer of each project into Eficode ROOT can be monitored. The dashboard provides all the project-related information easily and smoothly, thus facilitating development follow-up. **As a result, both your own and the outsourced software developers can focus on producing the best possible code while business operations focuses on the development of new services.**

One of the objectives of Eficode ROOT is to free up resources in the long run for operationally critical infrastructure and software projects and to provide the decision-makers the best possible visibility and predictability of the various stages of software development. Additionally, Eficode ROOT has been built to be modular so that it can be developed further by adding new tools and easily modified to match the company's needs. **The tried-and-true choices of technologies and tools ensure a flexible, vendor-independent environment that can meet the requirements of your company - now and in the future.**

Eficode has dedicated itself to the continuing development of ROOT and we believe that our customers will receive commercial, measurable benefits from using our service, both from a cost perspective as well as from the point of view of business operations development. **Devops and Eficode ROOT are developing together, providing companies the ability to respond to constant changes in the operational environment, to decrease costs, and to create new business.**

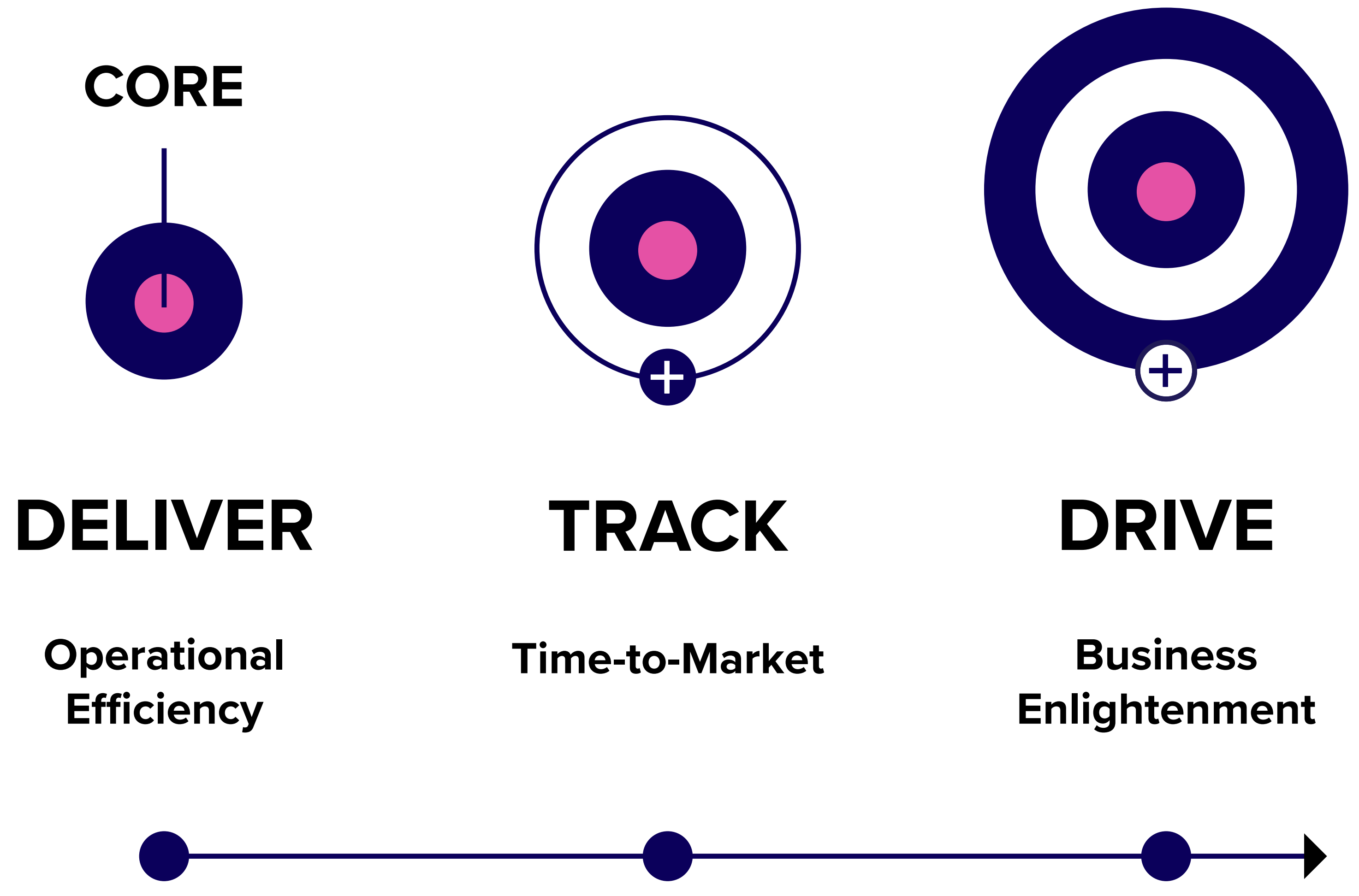
DevOps Platform

For experts

FOR EXPERTS

Modular, scalable,
and -above all - fit for purpose.

A good platform consists of levels matching
the three phases of the DevOps journey:
Diliver, Track & Drive



CORE

DELIVER

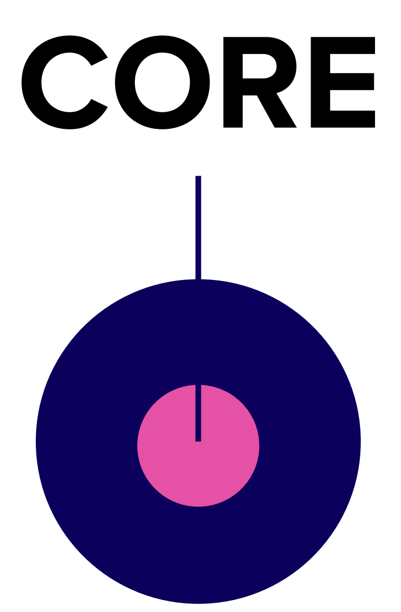
Operational Efficiency

TRACK

Time-to-Market

DRIVE

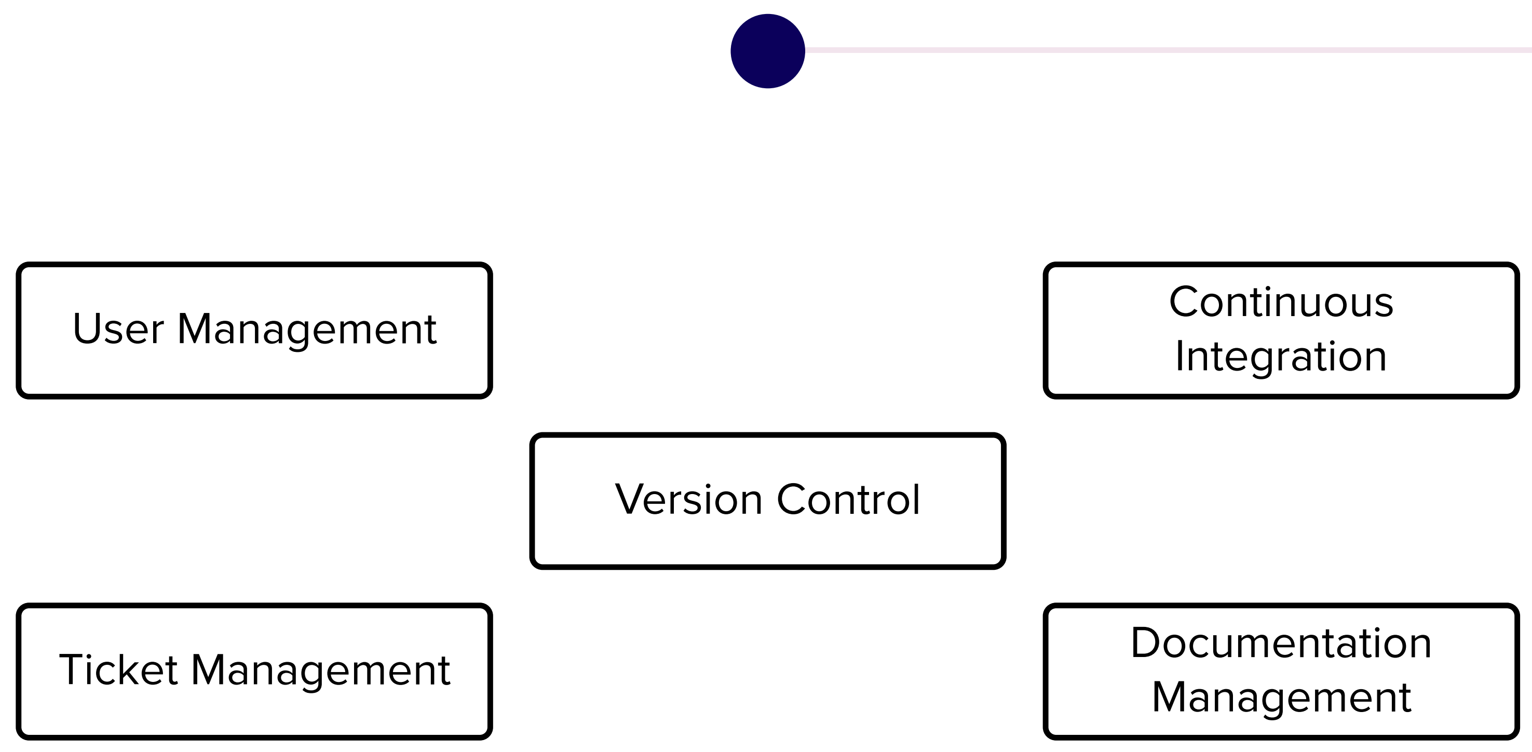
Business Enlightenment



CORE

DELIVER

Operational Efficiency



User Management

Continuous Integration

Version Control

Ticket Management

Documentation Management

At the first level, software development is stabilized and harmonized between all projects. This allows for retaining the projects' IPR creating a unified procedure based on automation.

Simultaneously, the management of both documentation and work tasks are unified to ensure understanding of what has been accomplished and what will be done in the future.

All these tools are integrated in such way that it is possible to access all of them using the same user account. The tools are also integrated with each other, making it possible to move information in the tools seamlessly.

Centralized User Management

All tools and systems should work with a single user specific login. This enables greater utilization and an easy-to-use, group-based user management. With group-based user management, the users are provided access to the systems without having to specify rights separately for each user and service. The management of entities is simplified and personnel changes can be controlled more easily. Group-based user management should be introduced by first defining three basic groups: system administrators, developers, and users with read-only access.

Without an easy-to-use group-based centralized user management, it is difficult to provide an adequately flexible environment to enable the other areas of the DevOps culture.

Version Control

Version control ensures sustainable development for a company when all software code can be found within one system. **The IPR owned by the company remains under its control** and version control system enables easily restoring any previous version. Additionally, version control enables simultaneous work within a team, without the risk of losing a team member's work.

Each version control tool has its own ways of controlling code. Therefore, it is essential to understand how the tool handles logical entities – for example, is the unit of change a new functionality or a singular edit in the code? By using the tool appropriately, tracking changes later is easy. Version control tools also have their own ways of managing a situation where two developers have changed the same file, thus facilitating resolutions to conflicts.

Functional version control provides the ability to define useful processes, for example, related to code review or release management. Version control must have sufficient data security and ensure that the work done by developers will not be lost, even accidentally.

Work Management

The cornerstone of tracking and directing work is the ability to see, analyze, and control it. The best approach to this is to create entries for singular tasks into a system than can be used to manage larger entities and to visualize, for example, fulfilled and incomplete tasks for a certain time period in different ways. **Profitability is increased when bottlenecks can be identified and resources focused on areas in which progress is critical compared to other simultaneous tasks.** By managing the dependencies between tasks, the amount of work can be optimized and the time used for waiting by employees minimized.

Due to its generalized nature, the work management system should be flexible enough to handle many functions. It should be able to monitor both software development as well as IT acquisitions. Easily understandable reporting, covering all of the collected information, should be available to all those who need it. When the general view of the work is available, a new perspective can be provided into how work is actually going and how it could be improved.

The work management system should also contain a clear record of what has been agreed upon and what progress has occurred in the matter. It is hard to argue against a clear history of the actual development and used approaches.

Documentation Management

A good documentation management system provides a centralized location to which all relevant information can be stored. It provides ready-made, adaptable templates for the creation of repeating information and enables easy management of user access and capsulating of project-specific information. With each new project, it should be easy to create a project-specific workspace for information storage and each project should have the option to prevent access to any potentially confidential information from persons outside the project. Minutes of meetings and the creation and sharing of calendar entries within a certain user group is an important part of interaction within teams. The system should also be suited for storing information that is of significance to everyone – for example, instructions regarding the company's external communication. **The documentation system should therefore be a channel for communication both within teams and throughout the whole organization.**

The documentation system should also be easily used for planning and communicating with persons outside the organization. It should be noted that information intended internal to the organization must remain that way also in these cases.

Continuous Delivery

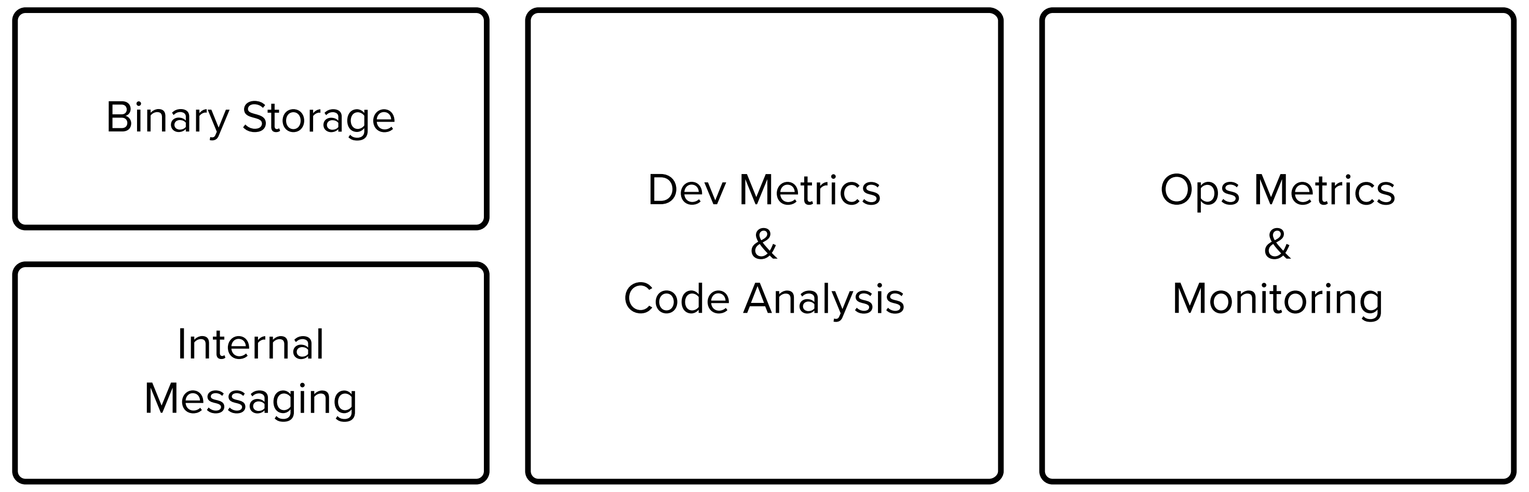
Continuous Integration (CI) has become an ever-widening concept, having evolved to also mean the support for continuous delivery. This brings new challenges for development, as it is no longer enough to standardise just the building and testing phases of development. The automation must be extended to the storage of completed software packages, provisioning of environments, and production delivery. The shortening of development cycles has forced development to automate testing, building, and delivery as far as possible.

The development of continuous delivery pipelines must be constant and able to respond to demands arising during development. This requires a constantly evolving ability to deliver software to various environments. For example, web-based services are expected to function in different browser versions on all operating systems. Another growing trend in continuous delivery are scalable build- and deploy environments which are defined dynamically using code, instead of the previous practice of manual installations in advance.

A good continuous delivery system supports the organization's current needs and evolves on demand. **When continuous delivery lags behind, modern, agile software development cannot be achieved.**

TRACK

Time-to-Market


Binary Storage

Internal Messaging

Dev Metrics & Code Analysis

Ops Metrics & Monitoring

This level is built on the previous one, providing a growing ability to communicate and to monitor the state of software development.

Harmonization between projects continues and real-time analytics and the refined metrics based on it are introduced to the repertoire of the projects.

Based on the metrics and analytics it is now possible to compare different projects against each other. This way good development practices can be identified and expanded throughout the company.

Binary Storage

During projects, various files need to be stored. It makes sense that these are also managed in a centralized manner to ensure smooth versioning. In this way, the same binary can be taken from testing all the way to production, even years later. Reusable libraries that are needed by several products can also be easily stored, versioned, and distributed within the organization. **In addition to version control, binary storage is a key element of IPR maintenance.**

When choosing a binary storage system, supported file types and technologies used in software projects should be taken into consideration to avoid unpleasant surprises later. Additional considerations include the amount of disk space reserved for the system and the implementation of its backups.

Internal Messaging

Email as a means of communication is frustratingly slow these days. When products are developed at a rapid pace, internal communication should be organized through an instant messaging system – responses must be available within minutes, not hours. It is es sential that **all interest groups within a project, from project management to subcontractors, are included in the same system to ensure fast communication and effortless flow of information.**

In functional, modern software development, it is essential that the various operators in business, sales, development and IT can discuss requirements and functionalities in real time. The continuous integration and delivery systems must be able to be integrated with the instant messaging system to produce shared, up-to-date information about the status provided by automation for everyone.

Operational Analytics

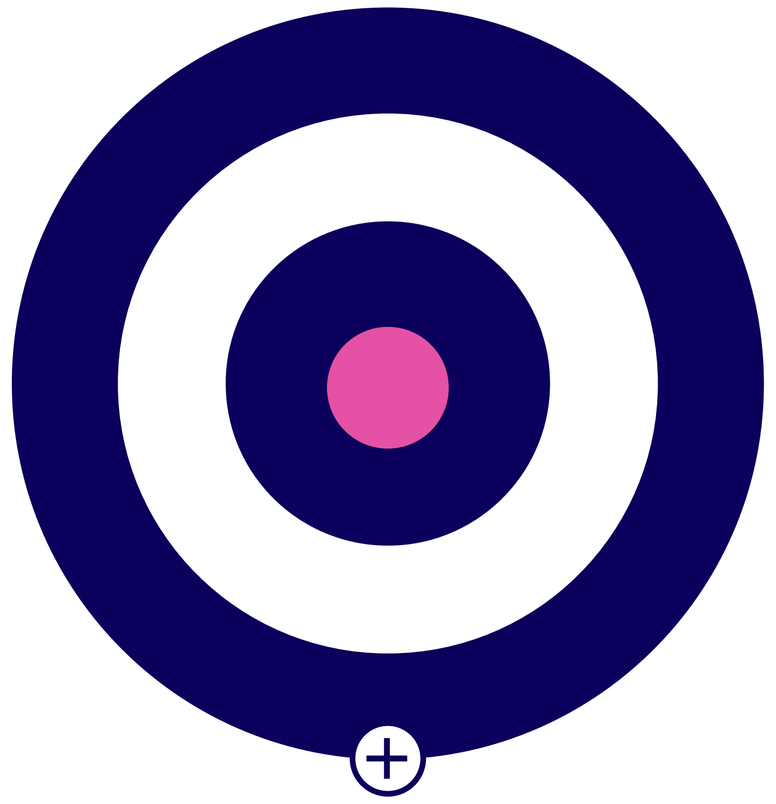
The number of projects in production is constantly growing and therefore the need to monitor them also increases. It is essential to be able to know in real time that all services are running and no immediate problems have been discovered. Additionally, the monitoring system should provide the opportunity to compile reports for the entire lifespan of the service, thus forming the basis for smart business decisions.

Successful monitoring system does not only detect issues when they occur, but is also able to warn in advance. Situations can be anticipated and service outages should be performed in a controlled fashion unlike in operations based on reacting. Monitoring should automatically provide basic metrics, reporting based on them, and be easily extendable.

Development Analytics

During development, seeing how software fulfills different quality standards is essential. Additionally, comparing various project teams with each other must be possible so that potential bottlenecks and other targets for improvement may be identified, thus increasing the overall internal quality of all projects. Development analytics must support and help the developers in their daily work, enabling the developers to help themselves to become even better.

Static code analysis provides information on, for example, how complex the code is, how common development conventions have been followed, and which parts of the software contain potential issues or vulnerabilities. By making static code analysis a part of continuous delivery, valuable information about the quality of code is provided regularly, thus enabling the monitoring of changes in quality between different versions. This can works as an automatic quality gate. **Directing development into the right direction along with achieving cost savings is obvious when problems are noticed before they require fixing.**



DRIVE

Business Enlightenment

- Service Desk
- Cloud
- Big Data
- Virtualisation
- Environments on-demand
- Containers
- Test Data Management

At this level, a way of offering environments that operate on the entire scale of the internet, yet still in a controllable manner, is needed. It is important to choose the right technologies and integrate continuous delivery with dynamic environments.

For example, the use of cloud services in development and testing increases the efficiency of software development, whether the cloud is internal to the company or a third-party service.

Now both visibility and an agile way of releasing software can be provided "on web scale".

Service Desk

Today, the management of provided IT services must be able to provide real-time information on the status of various deliveries and the opportunity to report and analyze costs. IT service acquisition processes must be able to be included in the software itself so that ordering and handling them is no longer prone to human error.

In addition restricting access also becomes relevant, especially when the IT services are provided by a subcontractor. Monitoring the realization of SLAs per subcontractor must also be possible. Subcontractors may also wish to integrate their systems with the Service Desk system as it is usually necessary to connect it to other internal systems. For this reason, extendability is one of the most important features of a working Service Desk solution.

The Service Desk system can also be used as a channel of continuous feedback in software development, especially in situations where the company has its own software product. In this case, the Service Desk system can also additionally operate as a customer service system. The transparency throughout the entire organization in accordance with Devops principles are easily achieved when developers have a view into the life of the product in the hands of the end users: when required, they can directly contact the end user, have a better understanding of business prioritization, and are better able to locate error situations.

Choosing a Service Desk system is often challenging, because different business operations and technical operators have their own expectations regarding the service. It is not enough that the Service Desk works and is easy to use, it must also support appropriate reporting. **The objective of the Service Desk system is not to make work more difficult, but to improve the customer satisfaction of the various IT services.**

Cloud Services

Utilization of cloud services in business operations has become attractive to many companies. Realizing the potential benefits efficiently and securely are among the biggest challenges. Collecting analytics, automatic release of products, and testing become complicated in cloud services so maintaining the real-time overall picture on the situation of development achieved in the previous levels must be ensured.

It should be noted that cloud services are quickly becoming a fundamental element of modern development and technology decisions. For example serverless architectures, big data or machine learning cannot be employed for the benefit of business operations without a cloud solution of some kind.

The implementation of cloud technologies in a company requires a well-thought-out cloud strategy. The key elements of a good strategy are predictability of costs, fulfilling the security requirements especially concerning legislation and the authorities, finding beneficial applications of cloud services within the company, predicting needs for flexibility and scalability of resources, the current service range of the market, and the monitoring possibilities for these services. The factors presented above should be formed into a roadmap, after which the creation of a concrete strategy is possible. It should be noted that **agile, iterative experimentation is a vital element of the creation of a cloud strategy.**

Virtualization

Virtualization is a way of using several different operating systems on one physical server. This makes it possible to, for example, run Linux environments on a Windows server. Virtualization is key when wanting to implement scalable and re-creatable environments. Virtualization enables the copying and creation of ready-made templates part of continuous delivery. This way, environments can be harmonized between different software projects and environments can be created on demand.

Virtualization enables development directly on the target operating system without needing to order, wait for, and install servers. Testing is made easier as the full environment is precisely the same from the developer's computer all the way to production. Making backups is also easier because, for example, virtual machines allow taking snapshots of the entire system, providing easy access to high availability and disaster recovery. **Virtualization is required for providing intra-corporate cloud infrastructures.**

On-demand Environments

Managing software in different environments is perhaps the biggest challenge in IT at the moment. Installation of servers is still largely manual work and constitutes one of the largest costs in IT organizations. **New technologies are finally making it possible to control environments in a way that makes it possible to implement new, better approaches.**

Whether dealing with light containers, cloud environments or other virtualization options, it is important to aim to provide environments in an agile manner to those who need them. Environments are not only needed by developers, but also business operations, testers, sales personnel, and the people in charge of the production systems. A demonstration of software project in a sales meeting is a typical example of a situation where a dedicated environment is required for a relatively short period of time. Therefore, the best approach is to include the installation of environments as a part of an existing delivery system. Thus, you can independently release products dynamically in a new environment, automatically freeing up resources when the environment is no longer needed.

Containers

Containers are a relatively new technology that can be used to provide entire software systems as dedicated, isolated packages. Containers can be thought of as virtual machines, but ones that consume far less hard drive space and processor power than entire virtualized operating systems.

As the benefits of virtual machines are provided with fewer resources than previously, the services can be scaled much more efficiently. **Building modern services with microservice architectures is especially disadvantageous without the use of some type of containers.** In addition to this, the developers can develop their projects with their own work laptops in precisely the same types of environments that are used in production and testing, thus decreasing environment-related bugs. The dependencies between the systems can also be managed more easily as the containers are easier to version compared to binaries.

Big Data

The amount of existing data is enormous and its volume will continue to grow in the future. Perhaps the largest unused potential in companies lies in data management, storage, and analysis, especially when talking about utilizing it in the software development.

It should be noted that different data sources require different methods of storage. With Big Data, if not earlier, it is important to handle both structural data with traditional SQL databases as well as non-structural data with NoSQL databases. In the future there will no doubt emerge more database solutions – without support for these, the opportunities to understand factors that affect business operations will be lost. Similarly, it should be noted how fast different types of information might be needed. All information does not have to be real-time, in some cases even monthly updates may suffice. However, **it is advantageous to the company if data collection is constantly developed and fine-tuned so that larger opportunities are not accidentally passed by.**

One benefit that is often overlooked with Big Data is ensuring and monitoring the adherence to legal regulations. When the user wants to know what information is being collected on them, this must be provided from all the systems they use – preferably aggregated from all of the various services into one handy real-time database, from which the report can easily be produced. With Big Data, distributed data can also be compiled and analyzed to improve the efficiency of marketing, further development, and operations. For example, marketing can use the data to understand why customers choose a product over competing products or reject it. The same compiled information will also show how the users use the product.

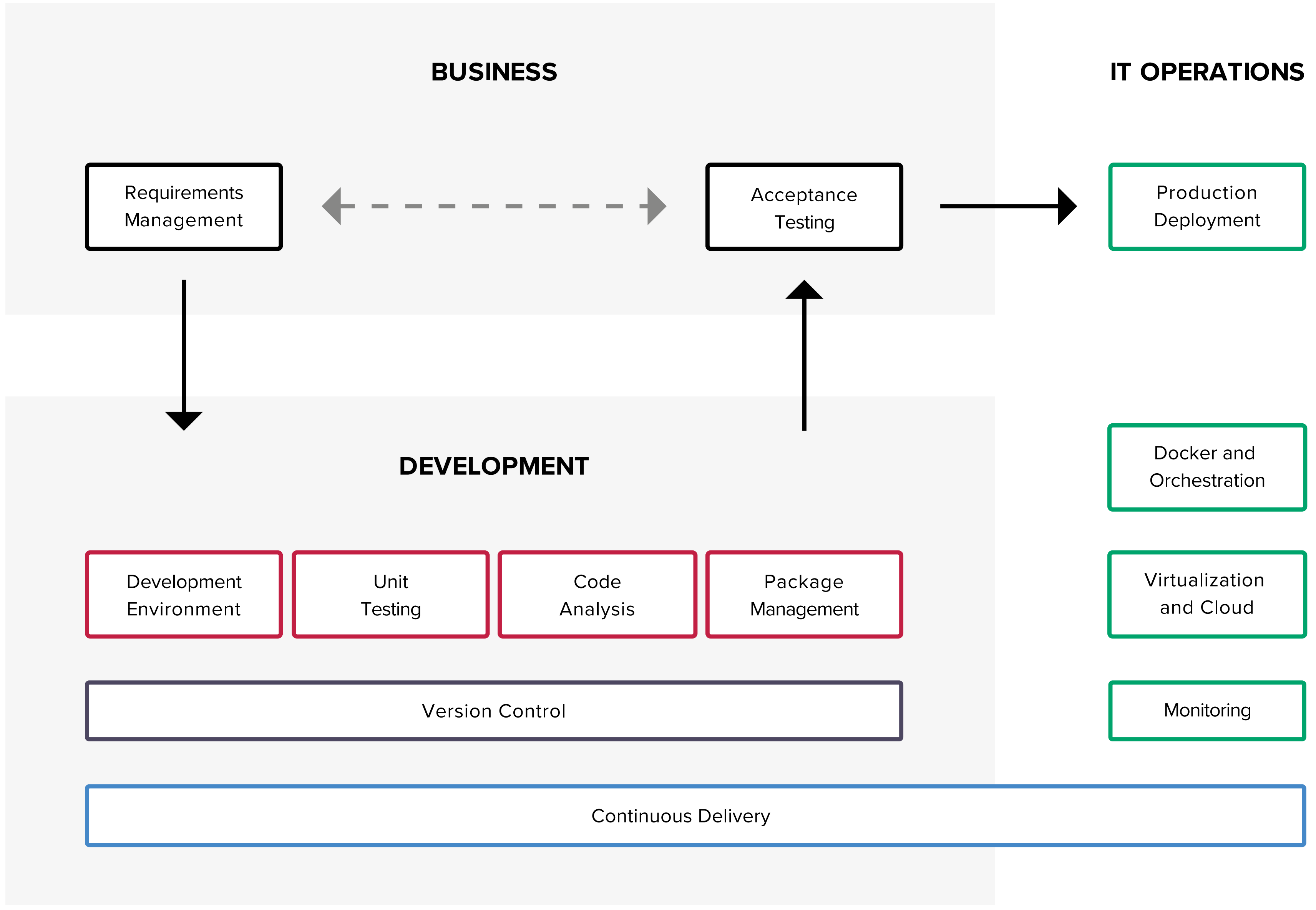
Test Data Management

In the near future, Test Data Management (TDM) will become an increasingly significant area of testing, among other reasons, due to EU's General Data Protection Regulations (GDPR). Companies must be able to prove what data their software has collected and where it is stored. This does not only concern production environments, but all other environments used in software development as well. It must be demonstrated that sample data used in development, testing, staging, and other environments is not a direct copy of production data, but instead data from which unique identifiers have been removed and replaced with generated data to provide meaningful testing.

Because the average software collects extremely large amounts of information covered by the aforementioned GDPR requirements, the data transferred between different databases can no longer be reasonably handled manually by a few database specialists, but they must instead use tools that handle the basic procedures – such as data masking – automatically. Thus, the specialists are able to focus on data management and building automatic reporting so that the laborious documents for authorities no longer have to be created manually.

For testers and developers, it is essential to be able to test the software being developed with real data as certain software errors, which only occur when end users have accessed the system, are often related to assumptions made during development. At the same time, the regulations mentioned above must be taken into consideration which means that the data must appear to be correct, but not include any unique identifiers. Additionally, it is difficult to transfer a copy of an entire production database from one environment to another when the database contains dozens of gigabytes of data or more. Good test data management makes it possible to analyze database copies and partially replace data with generated, correctly formatted data automatically. As a result of analysis, it is easier to compile smaller database copies, containing the representative data, that can be more easily transferred between different environments.

THE FUTURE



Software development is a constantly changing and developing environment. It is important to be a frontrunner, utilizing the newest, yet battle-proven technologies to support software and IT development.

For example, new legal requirements must be smoothly integrated as part of the development as otherwise it may not be possible to correct them later without massive costs. Security and performance testing are also areas where suitable automation tools are just beginning to emerge. As virtualization, especially with containers, develops further, ever-larger entities can be managed using automation.

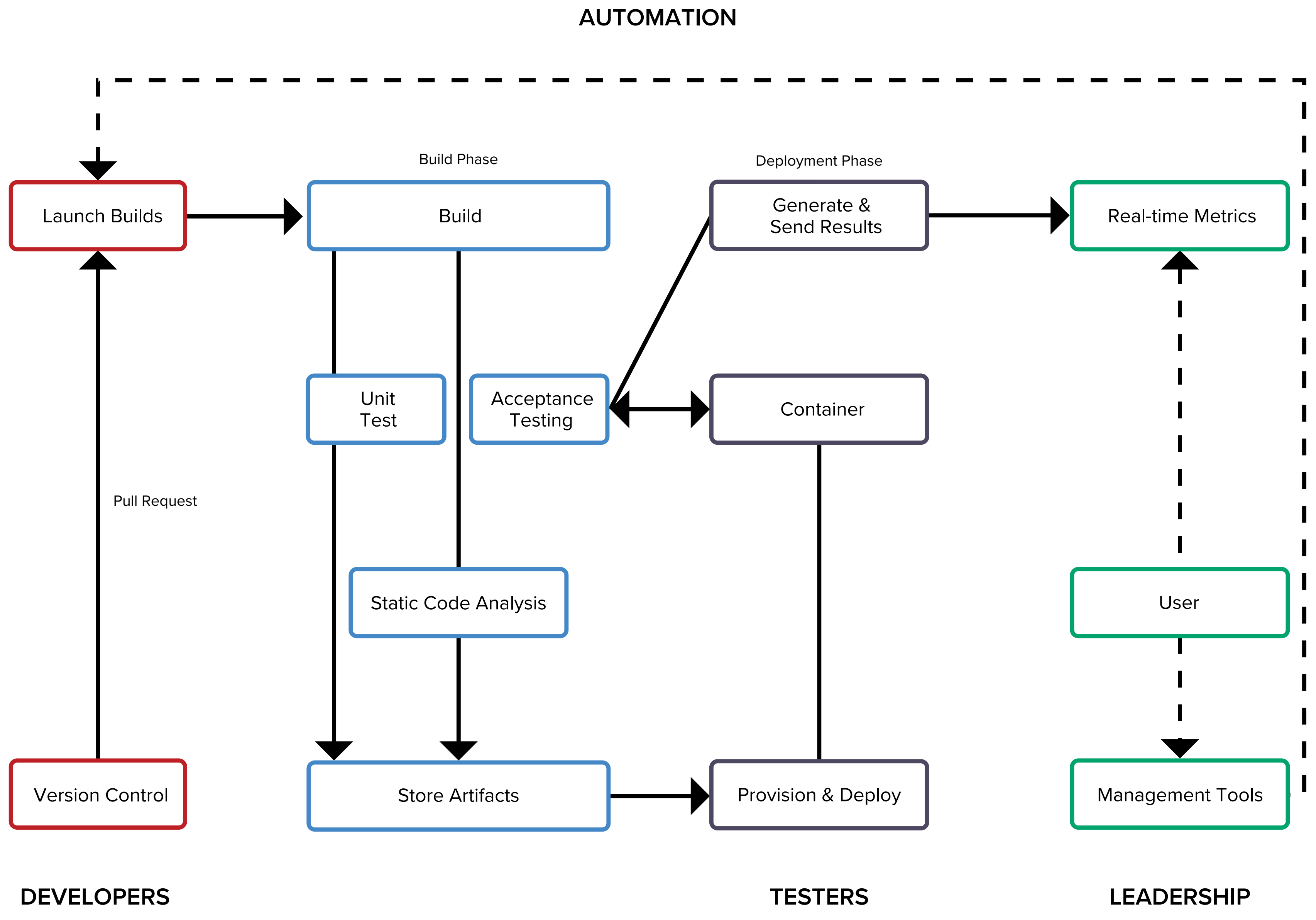
The utilization of new technologies, improving processes, and easing resourcing never ends. When DevOps way-of-life has been deployed comprehensively, continuous development and learning has become a part of the company's DNA.

HOW TO IMPLEMENT A DEVOPS ENVIRONMENT

While you could implement everything described in this guide yourself, would developing, maintaining, and upgrading such environment still be a part of your company's core business?

Eficode ROOT is an outsourced service that provides a DevOps environment as described, with predictable costs and the best customer support.

It provides all the modern tools in one place, including customization options – letting you focus on the essentials instead of the tools.



ADOPTION OF EFICODE ROOT IS EASY

01

INSTALLATION

There are two options for installing ROOT. The first option is a dedicated cloud solution for the company, in which case Root will be available for use within four weeks. The other option is to install ROOT inside the company's IT infrastructure, in which case some additional time will be required for the installation

At this stage, the basic tools used in the pilot phase will be chosen. The tool selection can be increased later as requirements evolve.

02

PILOTING

The tool chosen in the piloting phase are installed, validated, and configured in such a way that they support the company's current and expected operations. At the same time, a group of key users will be trained, chosen from the personnel of your organization. After training, the first projects can be transferred to ROOT. We will also begin the weekly meetings, with which the progress of the piloting is monitored.

We will only move forward from this stage when everyone is completely satisfied with the first projects transferred to ROOT.

03

FURTHER DEVELOPMENT

During the further development phase, the rest of the organization's projects independently move over to ROOT. The transfer is supported as necessary.

The management of users is scaled flexibly and the implementation and deployment of development ideas and new tools that have emerged during the pilot phase can begin – in accordance with DevOps, ROOT must also be in a state of continuous development.

04

MAINTENANCE

Implementation and support do not end when projects have been transferred, but instead Root keeps developing and updating to match the needs of your organization.

When entering into a maintenance agreement, it is important to specify the necessary SLAs concerning the ROOT's operations. It makes no sense to pay for 24/7 maintenance, if actual use of the system is only 9/5. The language requirements for support must also be taken into consideration.

CONTACT US

For Management

Marko Klemetti
CTO

044 522 5927

Tatu Kairi
Don’t Stop Believin’

Kalle Sirkesalo
DevOps Cutthroat

Mika Aho
Bleeding Edge

Pohjoinen rautatiekatu 25
00100 Helsinki

Åkerlundinkatu 11 A
33100 Tampere

Uplandsgade 56, 1 sal,
2300 København S

Birger Jarlsgatan 18A, 2tr,
114 34 Stockholm

Sörhallstorget 10, 3tr,
41763 Göteborg

De Entree 143,
1101 Amsterdam

DeMarcel-Breuer-Str,
15 80807 Müchen

For experts

Tuomas Keränen
Leader, Devops Platform

040 530 1672

Tatu Kairi
Don’t Stop Believin’

Kalle Sirkesalo
DevOps Cutthroat

Mika Aho
Bleeding Edge

Eficoderoot.com