LEVERAGE TECHNOLOGY TO ACCELERATE DIGITAL TRANSFORMATION

BUILD VISION, RETHINK ARCHITECTURE, MAKE DATA ELASTIC, AND ADAPT ENGINEERING PRACTICES

Before embarking on a digital transformation journey, it's essential to create a vision with a set of clear targets, adopt a holistic yet realistic approach, and partner with the right advisor who can support you on both the technology and strategy aspects of your initiative. Companies that pursue these strategies need answers to a range of questions: which assets to modernise and when? which processes to transform and how to transform them? how to re-design the architectures to supports the vision? how to build, use, and deploy software?

To help clients answer these questions and define how to execute their transformation, Plain Source advocates a practical approach based on four fundamental catalysts:

- Build vision with clear targets
- Rethink architecture to support the vision
- Make data elastic and discoverable
- Adapt engineering practices

This paper aims to describe how to leverage these catalysts to accelerate the digital transformation.

Build vision with clear targets. when planning digital transformation, you must set the appropriate vision for your business. For some companies, the transformation will mostly be about using technology and data to improve operations. For others, it might involve building new business modes. The visioning exercise should include: obtaining senior management commitment; setting clear and realistic targets and identifying the key initiatives needed to achieve targets.

Figure 1: Key catalysts of our approach

1. Obtain senior management commitment. Without the CEO and the senior leadership team support, any transformation will eventually fail.
2. Set clear and realistic targets. Without clear targets, people find it hard to accept the inefficiency of the old ways of doing things. Example of clear targets might be: the percentage of processes that will be automated or the fraction of new code that will be tested automatically.
3. Identify the key initiatives needed to achieve targets. Initiatives that are strategically important, pay back quickly, and reduce complexity are the ones to execute first.

Rethink architecture to support the vision. In the past, a company's IT often kept everything together in big, monolithic systems. By adopting a digital architecture, they can approach building out systems with greater flexibility. You can bring agility to your system landscape by separating out and delivering those same business functions, using application

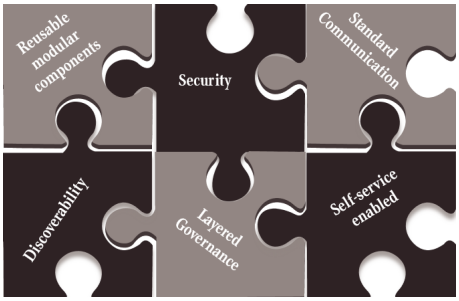


Figure 2: Rethink architecture to support the vision

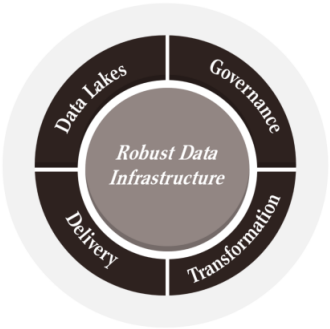


Figure 3: Make data elastic and discoverable

programming interfaces (APIs), cloud-based service platforms, microservice architectures and real-time data lakes, leveraging essential parts of the core with greater agility. Moving away from monolithic design and going towards a *network of smaller applications* requires re-thinking of the key architecture elements described below.

Standard communication: The base architecture for the application network provides a standard communication model which simplifies data exchange between applications on the network.

Security: Any application connected to the application network will be subject to the same security policies and access controls. Different domains can have different security policies, which allows enterprises to segment and provide access to that data depending on the data source, consumers, geographic location, or other factors.

Reusable, modular components: All data resources on the application network are made available through reusable interfaces, meaning that any application on the network is composable.

Layered governance: The application network mandates well defined interfaces to access resources. It also provides a layered governance model that makes it possible to track data consumption from consumer applications (such as mobile apps, web apps, dashboards, analytics) all the way through to the back-end data. It can also track dependencies between applications and even perform change impact analysis.

Discoverability: The application network is designed to exchange information between applications and people. This means that assets on the network are discoverable. In addition, different consumption models can be designed around different groups of users within or outside the enterprise. It provides a set of core APIs, allowing solutions to be built that enable different consumption models for different groups of people across the enterprise.

Self-service enabled: It provides a core set of services to enable consumers (developers, analysts, data scientists, creative teams, mobile developers, ops and admins) to access the network in ways that makes sense for them through tools they understand. It also has a set of APIs that allow new consumer models to be built.

Make data elastic and discoverable. To make data elastic, you need to create data lakes by pulling data about transactions and markets instantly into a data lake and apply machine learning algorithms to derive insights and solutions. Using data effectively requires the right data architecture and governance, built on a robust infrastructure.

Infrastructure: The new bidirectional integration of technologies is dynamic in nature; it happens in real time. For example, your existing master data-management catalog will need to be extended to include third-party data and potential integration with external master-data providers. This requires a robust infrastructure and data platform that can be easily adopted by external parties.

Governance: Security, quality, and integrity of data, including auditing and access controls. Data Operators can ensure that people have access only to the right data, that data has been secured according to policy, and that they can trace activity through a verifiable audit trail.

Delivery: Distribution and provisioning of data environments. Environments should be fully functional and ready for use by data consumers, and speed of provisioning is critical. It includes ability to provision new environments in places radically different from the source, such as provisioning a new environment in the public cloud from on-prem data.

Transformation: Modification of data, including masking and platform migration; such converting data in a relational database to a NoSQL platform. Data consumers need data in a certain form to make it usable. This might include redaction of sensitive information, changing data platforms or versions, or refactoring data from an asynchronous API to support data-driven API development.

Data Lakes: allow you to transmit data to end users faster.

1. Data are captured in applications as part of business processes
2. Data are extracted from core applications and prepared for staging. Hadoop-based data lakes are common location to store large volumes of raw or unstructured data from multiple sources
3. Data are structured for end-user consumption. View stores data are configured into snapshots of information that can be easily transmitted and consumed by end-user applications and portals
4. Data are pushed to end users for consumption. Client portal provides a single view of data to end users, with limited latency. End users can easily drill down to the next level of detail

Putting a data lake at the center of your architecture allows you to untrap valuable data from legacy systems and gives all your applications access to a continuous stream of intelligence.

Adapt engineering practices to bring agility. The value of architectural change accelerates when companies also embrace newer ways of working that speed up development and delivery tied to agile approaches. This enables you to rapidly launch new applications and processes that were once hard to build and scale. DevOps is the key enabler which accelerate the software development and delivery.

Automation DevOps relies heavily on automation and that means you need tools. Open source tools as well as the proprietary tools. DevOps relies on tool-chains to automate large parts of the end-to-end software development and deployment process.

Continuous Integration (CI) is a software engineering practice in which isolated changes are immediately tested and reported on when they are added to a larger code base. The goal of CI is to provide rapid feedback so that if a defect is introduced into the code base, it can be identified and corrected as soon as possible.

Continuous Testing is not just a QA function. In fact, it starts in the development environment. Developers make sure that, along with delivering error-free code, they provide test data sets. They also help test engineers configure the testing environment to be as close to the production environment as possible. Test engineers meet the challenge of quick turnaround by not only automating much of the test process but also redefining test methodologies.

Continuous Delivery (CD) DevOps extends to the entire release chain, including QA and operations. The result is that individual releases are far less complex and come out much more frequently.

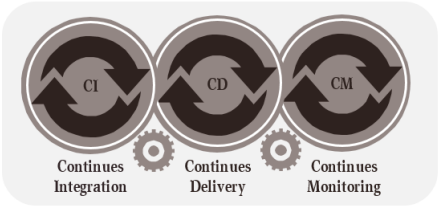


Figure 4: Adapt your engineering practices

Continuous Monitoring Given the sheer number of releases, there's no way to implement the kind of rigorous pre-release testing that characterizes waterfall development. Therefore, in a DevOps environment, failures must be found and fixed in real time. How do you do that? A big part is continuous monitoring.

Redefine processes and governance You may want to look across the entire spectrum of software-delivery processes to determine which will need to be redefined or fully automated so that development teams can take advantage of infrastructure as a service, as needed, and so that code can be ported into testing and production environments in a standardized way. To fully benefit from the DevOps, you should enforce "self service" for developers; teams can test, promote, and deploy code in production environments without requiring constant hands-on involvement from infrastructure-operations teams, although both teams share responsibility for code performance. You should also impose rigorous, automated testing of new code at all stages of the application-development process. Additionally you should take advantage of advanced analytics and other tools to preemptively scan code for exceptions and send developers automated reports about the code segments that are most likely to create errors.

The value of implementing DevOps can be significant with respect to both productivity and time to market. But the implementation of DevOps is not simply about the deployment of new IT methodologies. It must be treated as a company- wide transformation.

✦ ✦ ✦

Digital transformation is not a science; it can not be achieved by following a cookbook approach. The only way forward is to establish a clear vision with realistic targets and leverage technology and data architecture as well as agile thinking to accelerate the transformation. Along the transformation path there will be important markers

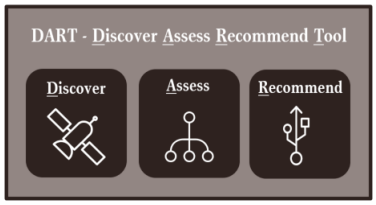
of success. IT strategy will become clearer as early pilot initiatives provide insight into decisions relating to technology architecture, data architecture, platforms and engineering practices.

The term digital transformation puts the focus on technological change. But it becomes clear to anyone who understands digital technology and its potential that the digital transformation is more of a fundamental rethink of the corporate model, for which architecture and digital technology are the catalysts.

About Plain Source

Plain Source is a small consultancy company specialised in Enterprise Architecture. We combine strong architecture and technical leadership with a background that includes architectural strategy definition and complex system integration. We work with you to create architectural strategy, ratify it with stakeholders, and help you implementing it.

At Plain Source, we blend strategic thinking and architecture to help our clients succeed in the digital age.



Discover, Assess and Recommend Tool

To be efficient, we use our own assessment tool, DART (Discover, Assess & Recommend Tool). The tool is built on our homegrown consulting experience which supports a principle-driven data capture and scoring features. Our assessment yields directional recommendations and is the launching pad for digital transformation.

Contact Information

Avenue Des Perdrix 17
1950 Brussels Belgium

www.plainsource.com
info@plainsource.com
+ 32 2 7259 519