NORDDESIGN CONFERENCE
REYKJAVIK, ICELAND, 12-14 AUGUST 2024

Faster and Faster: Automating the Hardware Engineering Process

Michael S. Peters, Steven D. Eppinger

Massachusetts Institute of Technology

Abstract: Organizations today face pressure to bring products to market faster, at lower cost, and with increased quality. One approach to achieving this goal is the automation of engineering workflow activities. This paper studies an organization that has successfully adopted automation within its engineering workflow. Using a field study approach, we document the tasks this organization has automated, the impact of automating those tasks, and the organization's process for selecting tasks to automate. We also discuss how other organizations might consider applying automation within their processes.

Keywords: Engineering Automation, Design Optimization, New Product Development, Industrial Devops, Design Automation

1 Introduction

Engineering organizations across the world are at the center of a historic transformation as they look for ways to bring products to market faster, with increased quality, and at lower costs. Recent years have seen many engineering and design processes transformed into fully digital workflows. But what is required to truly realize the benefits of modern concepts like agile, devops, and digital thread? This paper examines an organization that has successfully increased the speed and productivity of their hardware engineering development operations by employing automation at key points in their engineering workflow. We explore how they have selectively automated critical engineering tasks via an integrated digital workflow that enables rapid iteration in a hardware engineering environment supported by agile and devops methodologies.

2 Background

Product development organizations have long faced pressure to bring products to market faster, at lower cost, and with increased quality. In volatile and complex environments, shifting customer demands and increased market competition have emphasized the need for organizations to increase engineering productivity and shorten product development cycle times (Clark and Wheelwright, 1992). Organizations developing complex hardware products have traditionally responded to this pressure by adopting waterfall or stage-gate-based approaches within their product development process (PDP), whereby phases of the product development lifecycle, executed by interdisciplinary teams, are followed by gate reviews by senior-level management (O'Connor, 1994). However, the recent success of software development organizations in increasing their product velocity through alternative development approaches, such as agile and devops, as well as through the selective application of automation, has inspired many hardware-centric organizations to consider applying these approaches to hardware development.

Agile development methods have emerged as a leading method for software teams to iteratively develop products based on core principles that value "individuals and interactions, working software, customer collaboration, and responding to change" (Agile Manifesto, 2001). Several agile methodologies have manifested based on these core principles, including kanban, lean, extreme programming, and scrum (Rigby et al., 2021). Agile methods such as scrum are focused on enabling a small team, consisting of a scrum master, a product owner, and developers, to deliver valuable product increments through a series of fixed-duration (typically a month or less) development "sprints" (Schwaber and Sutherland, 2020). Within each sprint, the scrum team conducts sprint planning activities, daily scrum meetings, a sprint review, and a sprint retrospective. The scrum master is the team's overall coach, ensuring that scrum processes and sprint events are executed efficiently. The product owner manages the product backlog and maximizes the value of the product increment developed during each sprint. The developers are responsible for executing the day-to-day tasks required to develop the product increment within each sprint. The challenge of managing these scrum processes increases as the product complexity and the number of scrum teams working within the broader product team increases. Scaled agile frameworks have been introduced to address these challenges, which include governance models for managing the collaboration and coordination between individual scrum teams (Ebert and Paasivaara, 2017).

While agile methodologies had historically focused on the development processes for software products, organizations employing agile noticed a divide forming between development and deployment operations. In many cases, bottlenecks were forming at the handoff between development teams (e.g., scrum teams) and operations teams (e.g., test and release engineers), resulting in inefficiencies and delays in releasing newly built product increments (Hemon et al., 2019). These inefficiencies led to the integration of development and deployment operations in what has come to be known as devops. The concept of devops focuses on the interdisciplinary nature of product development, whereby development and operations functions are encouraged to collaborate across functions through continuous software engineering activities (e.g., continuous planning, continuous integration, continuous testing, continuous deployment, and continuous monitoring) (Fitzgerald and Stol, 2017). The continuous engineering activities associated with devops are underpinned by many of the same cultural enablers that support agile methodologies, such as shared team goals, constant communication between teams, and cross-functional collaboration. However, the successful execution of devops also requires technological enablement in support of task automation (Smeds et al., 2015) to reduce tedious, repetitive engineering activities in favor of more high-value, creative activities.

Despite the success of software engineering teams in implementing agile and devops practices, hardware engineering organizations have encountered significant challenges in achieving similar successes. Past studies have emphasized the challenge of adopting agile methods, including the constraints of physicality and cultural and mindset shift challenges (Drutchas and Eppinger, 2022; Atzberger and Paetzold, 2019). Additional challenges documented when hardware organizations have sought to implement devops practices include adopting a multi-disciplinary approach to operations and the high cost of rapid hardware development (Koren et al., 2023). Despite these challenges, case studies have highlighted organizations that have successfully adopted these frameworks within hardware development through the adoption of enabling digital engineering technologies such as generative design tools (to aid rapid design iteration) (Schrof et al., 2019) or distributed version control systems (that allow for branching and merging of designs) (Stirling et al., 2022).

In addition to product development methodologies, digital transformation initiatives have also focused on supporting increased development speed of hardware products through more efficient design collaboration and data sharing between engineers, designers, and their downstream customers. Specifically, “digital thread” initiatives have been aimed at increasing information sharing and visibility between organizational functions (e.g., design, manufacturing, operations, aftermarket) by linking together core systems and tools such as model-based systems engineering (MBSE) tools, computer-aided design (CAD) tools, product lifecycle management (PLM) systems, and enterprise resource planning (ERP) systems. Integrating these systems and their underlying data results in a common data model that can be used as the source of truth between functions (Singh and Wilcox, 2018). Recent advancements in digital engineering tools and their integrations with collaboration tools (e.g., Slack, MS Teams) have resulted in increased accessibility and lower barriers to adoption. Additionally, increased process integration and higher performance of the tools have resulted in the adoption of systems thinking approaches and a focus on solution selection (as opposed to solution generation) within large-scale engineering organizations (Marion and Fixson, 2020).

Design automation presents the opportunity for increased design and development speed when applied to tedious or repetitive engineering tasks, those that are prone to error, or tasks where optimization analysis may be utilized in support of concept definition (Vidner et al., 2022). Examples of more widely used design automation techniques include the utilization of generative design and topology optimization tools. Both tools allow the engineer to be supported by a computer program that automatically iterates on potential design options to satisfy a variety of different constraints (e.g., allowed volume, weight, load constraints) (Peterson and Summers, 2021). When used with additive manufacturing techniques, these tools also support rapid prototyping and part fabrication within an organization, as opposed to outsourcing to contract manufacturers.

3 Research Approach and Objectives

Our field research is based on a study of engineering automation at Ocado Technology Solutions, a business segment of Ocado Group (an online grocery services provider with 2022 revenues exceeding £2.5B). Ocado Technology Solutions is responsible for the development of Ocado Group’s core managed services product platform, the “Ocado Smart Platform” (OSP) (Ocado Group, 2023). The OSP is a seamless end-to-end, digital grocery shopping and delivery platform sold to grocers worldwide. It includes a mobile grocery order interface that is directly integrated into customer fulfillment centers (CFC), where smart robots operate on a 3D grid of bins containing thousands of grocery items. Robots transport the bins for picking and packing orders and to optimize the warehouse using machine learning to achieve maximum efficiency. The robots are developed internally by Ocado Technology Solutions (hereafter referred to as “Ocado”). The most recent iteration of this robot, known as the “600 series bot”, is the central product of the engineering workflow technology and processes studied as part of this paper.

Ocado’s 600 series robot is designed and developed via an agile development approach by interdisciplinary teams from across Ocado’s six core functions known as “crafts.” Team members from across the crafts of mechanical engineering, electrical engineering, mechatronics, manufacturing and assembly, testing and insights, and industrial devops self-organize into teams that execute projects aimed at addressing priority system “problems” from a manicured backlog over the course of three-week sprints. Previous work (Drutchas and Eppinger, 2023) considered the uniqueness of Ocado’s application of scaled agile frameworks to their hardware development process.

We seek to understand how Ocado has substantially increased the speed and productivity of its hardware engineering processes by automating specific engineering tasks. Ocado's automation is enabled by their development and adoption of a digital workflow operating within an agile environment which is inspired by similar successes of their software engineering teams. Our study examines three aspects of engineering automation: 1) which engineering tasks they have decided to automate, along with what technology has enabled these automations, 2) how they selected these tasks for automation, and 3) the impact of automation on the time required to develop their 600 series robots. Additionally, we propose an approach for how other organizations may decide where to selectively apply automation to their engineering processes and consider which types of engineering scenarios are most suited for automation.

The method used to conduct this research was a field study approach. Interviews and observations were conducted over a period of 10 weeks, virtually and on-site, with 20 technical leaders and team members from Ocado to gain insight into the technology and processes that allow them to bring their product to market faster. Over more than 20 hours of interviews, team members and technical leaders described their engineering processes, provided demonstrations of the technical solutions that enable engineering process automation, explained how they decided which engineering tasks to automate, and provided internal data to quantify the benefits of process automation.

Interview questions were tailored to the interviewee to ensure an appropriate perspective was gathered on relevant engineering processes during each interview. The questions covered topics such as past and present engineering workflows, Ocado's internal technology architecture, descriptions of technologically enabled automation, reflections on the impact of automation, and the process for deciding where to apply automation. Detailed notes were taken for all meetings and interviews to provide the background for understanding how Ocado utilizes technology and novel processes to enable automation of their hardware engineering processes. Follow-up interviews were conducted as necessary to clarify interviewee responses and to explore further topics that were raised across interviews.

We also collected several existing data sources to complement and support the findings and insights from interviews and observations. Ocado provided structured quantitative data generated during a previous internal time study to illustrate the time savings realized by reducing common and recurring manual errors during physical design. These data were analyzed using simple trend analyses based on the cumulative number of design merges during the development of Ocado's series 600 robot. Ocado also provided a variety of unstructured data sources in support of this study, including artifacts and documents such as process pain point documentation from past interviews, end-to-end organizational value stream maps, digital whiteboards used in support of automation planning, and roadmaps for future automation development. The data extracted from these artifacts were synthesized and reviewed in the context of the information provided during interviews with Ocado employees.

4 Results and Key Observations

4.1 Engineering Workflow Automations and Enabling Technologies

As do many engineering organizations today, Ocado develops its products in a fully digital engineering environment built upon an integrated ecosystem of digital tools. This digital engineering environment is formed around the cloud-based CAD platform Onshape, which serves as the repository for the centralized source of truth for all hardware design and development. Onshape is integrated with digital simulation tools for advanced design analysis, topology optimization tools for design optimization, an additive manufacturing tool suite for nesting 3D printed parts, and planning and internal team communication and collaboration tools. Integrating these tools provides traceability across functions and eliminates the need for static artifacts (digital files) to be created and handed off between teams, thus avoiding many of the challenges of version control.

Onshape also enables increased visibility across the organization to the authoritative source of truth for design through its branching and merging function. Branching and merging in Onshape ensures that no local files are being worked on individual engineers' hard drives and that everyone within the organization works from a common version of the data stored in the "main" branch. Test branch versions may be created off the main branch to explore new design concepts during individual sprint cycles before reviewing and merging approved changes to the main branch. This process mimics the concept of "continuous integration" within software engineering.

With this enabling digital engineering environment in place, Ocado has been able to automate engineering tasks by building custom feature scripts within Onshape and other integrated tools. Four examples of engineering tasks where automation has been deployed within Ocado's engineering process include CAD design checks and reviews, topology optimization processes, nested part serialization, and manufacturing pack creation.

CAD Design Checks: Automating CAD design checks allows engineers at Ocado to ensure that the individual CAD parts they are designing have proper attribution consistent with best practices and rulesets being followed within the overall system design. Engineers can run a design validation check directly on data in Onshape to check CAD metadata for completeness and correctness. With the click of a button, engineers can review that critical CAD attributes such as the Ocado part number, component weight, and component material are correctly filled out. These automated checks result in high quality, standardized design data that reduce the overall technical debt of the design and the risk of future rework. Additionally, the team has gone a step further and integrated these automated CAD checks with their team communication and collaboration toolset, Slack, to allow the automated checks to be run directly from channels where individual sprint teams discuss and review designs. The results from the CAD design checks are immediately and automatically posted into the Slack channel for the entire team to review and discuss in real time.

Topology Optimization: Automation of topology optimization processes has enabled engineers at Ocado to rapidly optimize the 3D shape of structural components directly from CAD data in Onshape. The integration they have set up between Onshape and their simulation tool allows engineers to quickly generate structural load tables within the simulation tool that can then be exchanged with Onshape before pairing the load data with the design volumes to be optimized. From Onshape, users can then kick off topology optimization processes via a GUI integrated with their topology optimization toolset. After the optimization tool runs for several hours (or overnight), engineers are delivered an output that contains a folder of optimized design options along with a graph of the Pareto front comparing performance against crucial design parameters (such as weight and stress) for all design options. Engineers and designers are left with the value-added decision-making process around which designs to select and merge back into the main branch within Onshape.

Nested Part Serialization: In addition to automating design checks and design optimization, Ocado has also introduced automation into the process of gathering data from build and test operations during development. They have done this by creating a program that automatically assigns serial numbers to parts and part features that are nested within their additive manufacturing tool suite for each prototype printing operation. Once parts are printed, post-processing data such as time required to print, print quality, and part breaks can be recorded against the serialized part and part features. This information is then displayed in a live dashboard that enables visibility of issues encountered during prototype build events. The data from this dashboard can be used to inform better decisions made by the design teams during future sprints.

Manufacturing Pack Creation: Finally, automation supporting manufacturing pack creation has enabled release managers at Ocado to efficiently export released product-defining data from Onshape for transfer to robot contract manufacturers. To support this automation, OnShape’s API was leveraged to automatically generate release notes, change logs, BOM exports, lists of additive manufacturing parts, part drawings, part step files, and standard operating procedures for manufacturing the bot. This automation eliminates the time required for release managers to manually search, download, and package these artifacts together. This automation also validates that the data package sent to contract manufacturers is the latest version of product-defining data, eliminating the potential for manual user error when searching and downloading data.

4.2 Selectively Applying Automation with a Focus on Value

To decide which parts of their hardware engineering process to automate, Ocado considered how various types of tasks had been automated within their software engineering processes. They also sought to ensure that the tasks that they automated enabled their engineering organization to deliver hardware that was “right the first time” and that could be delivered “rapidly and cheaply.”

With these principles in mind, the Industrial DevOps team at Ocado took a three-step approach to identify engineering tasks that could be automated. The first step in this process was a set of interviews across all six crafts and the broader product team (e.g., product managers and the senior leadership team). The objective of these interviews was to understand the biggest pain points in the engineering process and where bottlenecks and slowdowns in productivity existed. Feedback from these interviews was gathered and distilled into discrete problem statements that were then organized by individual craft.

The second step the team followed was to gather data on the hardware engineering process by creating a value stream map of Ocado’s full end-to-end product development process. Unlike in their software engineering process, where data automatically provided from tools enabled visibility into the process, Ocado found that many of their hardware engineering tools limited their ability to thoroughly interrogate their process. As such, manually creating a value stream map allowed the team to visualize which tasks within the end-to-end process had a large number of sequential steps, which tasks were repetitive, and which tasks took significant time to execute as compared to other tasks. Once the end-to-end process was fully understood, the problem statements identified during interviews were mapped to the corresponding tasks in the value stream map where the original pain points were being realized. Additionally, problem statements were logically grouped into categories to allow for consolidation and development of unified solutions.

Finally, once problem statements had been mapped to the value stream, the team synthesized data on task frequency, duration, and the severity of the associated problem statements, along with input from the senior leadership team to generate a prioritized list of problem statements to be addressed. High-priority problem statements were then road-mapped according to the logical groupings created during value stream mapping. Initial road map creation was done by focusing on the high-priority problem statements, whose solutions would be foundational to future ways of working, before focusing on problem statements that were more advanced and could be built upon foundational solutions. The roadmap was time-phased (now, next, later), and problem statements were broken down into deliverable increments, including solution discovery and solution development phases to ensure that work was done to fully architect solutions that would be developed for "production" use by engineers and designers.

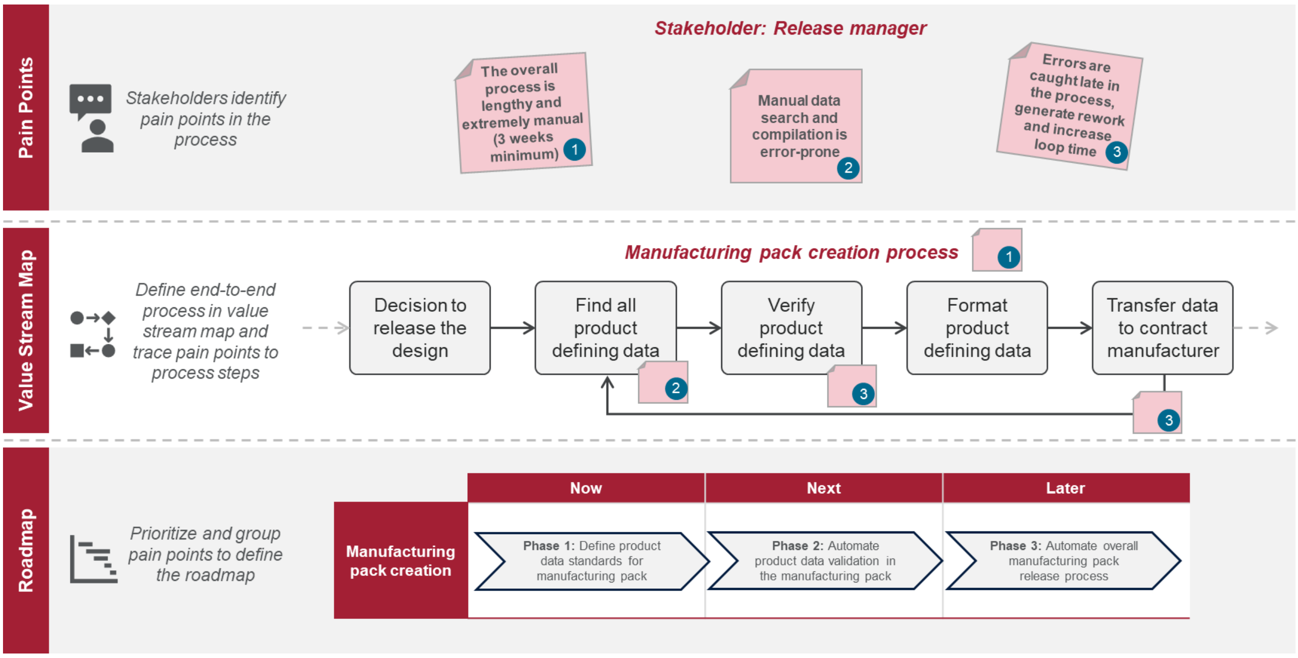


Figure 1: Example of how pain points are translated to solution roadmaps for the manufacturing pack creation process

An example of a problem statement that was identified as an opportunity for automation during this process was manufacturing pack validation and release. During qualitative interviews, the team at Ocado gathered pain points such as, "The process is manual; it is impossible to not make a mistake" and "Errors caught late in the process generate rework." These pain points were mapped to the end-to-end value stream map and logically grouped within the category of "manufacturing pack automation" before allocating development increments to the engineering automation roadmap to address the need articulated by these pain points. Figure 1 depicts how the steps described above were followed to gather pain points, map the pain points to the value stream map, and identify solutions that would be incrementally built over a defined period of time.

4.3 Increasing Speed and Productivity via Automation

The impact of automating engineering tasks within Ocado's organization has resulted in both qualitative as well as quantitative benefits to the organization. From a qualitative perspective, leaders that we spoke to within the organization believed that the automations they had adopted within their engineering workflow had resulted in higher quality data outputs for downstream consumers (e.g., contract manufacturers or internal build and test teams) by enabling systematic ways of finding design issues before they manifested themselves in a physical product. Additionally, engineering leaders reported that automations had eliminated the non-value-added time typically needed for engineers to check data for correctness, search and manipulate data, manually document results, and package data for both internal and external use.

From a quantitative perspective, we found that automation has saved Ocado measurable time during the design and development of their 600 series robot, as shown in Figure 2. In advance of the 600 series robot's release to production, data exported from Ocado's CAD tool show that there were nearly 6000 merges of individual design changes to the main branch in Onshape. However, the vast majority of these merges occurred after the introduction of CAD design check automations, which increased the ease of doing design checks and allowed for smaller batches of designs to be checked into the main branch at any given time (as opposed to the large batch size of design check-ins typical for traditional engineering environments). During this time, automated CAD design checks have saved Ocado over 2500 engineering hours (on average as validated by data from an internal Ocado time study), or the equivalent of more than 350 person-days of effort (based on an 8-hour day estimate).

Other examples of the benefits of automating engineering tasks include reducing the time needed to optimize parts via topology optimization techniques. Before automations were built into the process to enable simple extraction and application of load cases, topology optimization would take up to a week per part. Following the implementation of automation, the topology optimization process can optimize ten parts or more per week.

Additionally, automation has improved internal collaboration between design, build, and test teams. With the automation that enables part serialization before printing, there has been increased traceability of issues occurring within prototype manufacturing, eliminating guesswork in root cause analyses and improving the feedback loop between manufacturing and design.

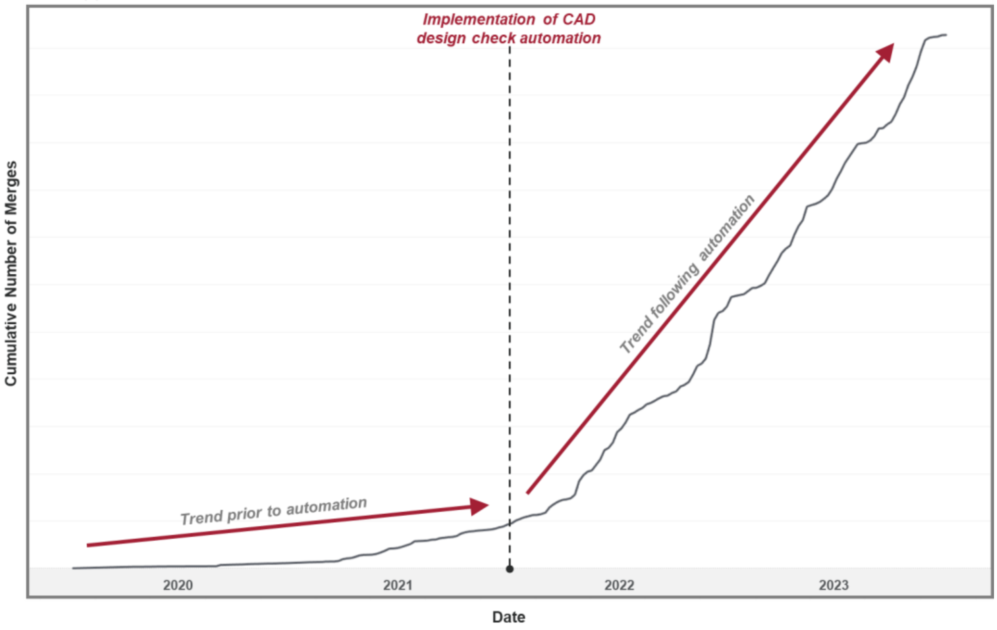


Figure 2: Cumulative merges over time and the impact of CAD design check automation

In the examples above, automation of individual tasks has significantly benefited Ocado. However, as more tasks are automated within their engineering process, these automated tasks build upon one another and the benefits compound. This is exemplified by the collaboration with contract manufacturers. Before implementing automation of CAD design checks and manufacturing pack creation, contract manufacturers would take three weeks or more to verify the product-defining data sent by Ocado. With the implementation of automation in the engineering workflow, contract manufacturers have been able to verify data in as little as three days after receipt of data from Ocado. The contract manufacturers benefit from both the accuracy of data that are validated via automated CAD checks as well as the speed at which manufacturing packs can be automatically generated and sent to them.

Senior leaders within Ocado believe that applying engineering automation practices will benefit their organization beyond just the speed of development. They believe this unique approach to hardware engineering improves employee engagement by creating a more enjoyable work environment that strips away tedious and repetitive tasks and allows engineers to focus on the higher value-added, challenging, and creative problems they enjoy solving. Leaders also believe that engineering workflow automation improves their ability to onboard new employees by reducing the learning curve associated with joining a new organization. These intangible benefits present opportunities for positive feedback loops to emerge and enhance the growth and success of the organization over the long term. These benefits also represent the cultural impacts of engineering automation on Ocado's organization. Despite the significant effort to develop and adopt these tools up front, as they become more heavily ingrained within Ocado's culture, the increased use of tools will magnify the benefit to the organization.

5 Discussion – Adoption and Application of Engineering Automation

Having described which tasks Ocado has automated within their hardware engineering workflow, how they decided where to apply automation, and the impact of this automation on their broader product development organization, we now propose a framework for how other organizations may consider which engineering tasks to automate, as well as the applicability of engineering automation to a range of organizational archetypes.

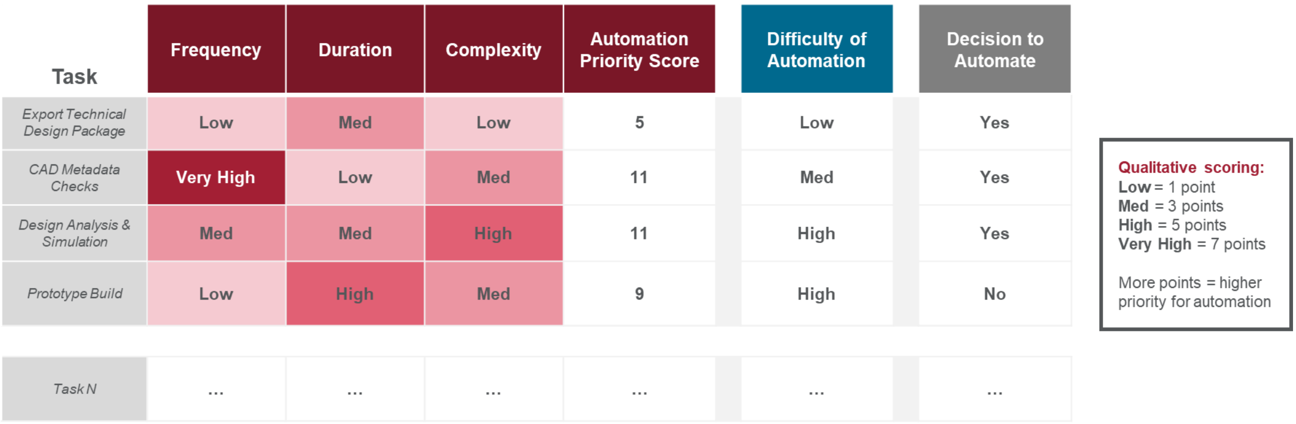
5.1 Adopting Engineering Automation – A Proposed Framework

When Ocado’s organization was considering which engineering tasks to automate, they followed a step-wise approach whereby they conducted qualitative interviews of the engineering organization’s largest pain points, developed a value stream map of all internal functions across the product development process, and then prioritized and roadmapped the pain points based upon the relative frequency of the pain points and input from their senior leadership team.

For other organizations aiming to automate pieces of their own engineering workflow, these steps may also be beneficial to their overall decision-making process and planning for engineering task automation. However, we would like to propose a selection framework to support prioritizing tasks for automation. Our proposed framework, as shown in Figure 3, allows individual engineering tasks to be ranked according to an overall priority score that is determined based on the attributes of frequency, duration, and relative complexity of individual engineering tasks. The engineering tasks with the highest overall priority score are most appropriate for automation.

To execute this framework, organizations should follow these five steps:

1. Generate a list of candidate tasks for automation by identifying the recurring and common tasks across phases from past program execution or from within an organization's standard PDP. Examples of engineering tasks may be “CAD metadata checks,” “design analysis and simulation,” or “build prototypes.”
2. Rate each task from "low" to "very high" on the attributes of frequency, duration, and complexity.
3. Compute the automation priority score for each task by assigning individual scores to each task attribute on a scale of 1 to 7 (with "low" = 1, "medium" = 3, "high" = 5, and "very high" = 7) and then by summing the attribute scores for each task. Tasks with the highest automation priority scores are the best candidates for automation. In contrast, tasks with the lowest automation priority scores are the least good candidates for applying automation.
4. Rate the difficulty of automation for each task on a scale of "low" to "high," where tasks with a "low" difficulty are those that do not require much effort to automate (e.g., an existing technology solution can automate the task) and tasks with a "high" difficulty are those that would require significant effort to automate (e.g., new technology implementation/development requiring significant time and resources).
5. Decide which tasks to automate based on those with the highest overall automation priority score and those requiring the lowest effort to automate. Once the analysis has been completed as described here, organizations may easily compare and decide upon which engineering tasks they should first consider for automation and begin to develop a roadmap for engineering task automation.



Task	Frequency	Duration	Complexity	Automation Priority Score	Difficulty of Automation	Decision to Automate
Export Technical Design Package	Low	Med	Low	5	Low	Yes
CAD Metadata Checks	Very High	Low	Med	11	Med	Yes
Design Analysis & Simulation	Med	Med	High	11	High	Yes
Prototype Build	Low	High	Med	9	High	No
Task N	...	...	...	...	...	...

Qualitative scoring:
Low = 1 point
Med = 3 points
High = 5 points
Very High = 7 points

More points = higher priority for automation

Figure 3: Proposed framework for identifying and prioritizing engineering tasks for automation

When considering this framework, it may be interesting to study the types of engineering processes that would score high on the automation priority scale and lend themselves towards the increased applicability of process automation. To explore this framework, we have grouped engineering processes by the development phases of planning, design, build, test, and deployment to easily identify where in the overall PDP task automation would be most appropriate, before evaluating some specific examples.

For any development phase, there are tasks within the phase that are frequently repeated, while others may be one-time activities. For example, activities within the planning phase tend to be singular, occurring upfront and in advance of other phases. For this reason, planning activities may be less good candidates for automation. However, iterative design, build, test, and deployment phases present significant potential for automation due to the increased frequency and duration of these phases and the significant complexity of the tasks within each phase. For example, within the design phase, tasks such as geometric dimensioning and tolerancing (GD&T) analyses can be complex and time-consuming for engineers to complete. Automating such a task would reduce the engineering hours spent during each design iteration on this task. It may also improve downstream manufacturing operations as automated calculations reduce the manual user error associated with engineering hand calculations. Within the build phase, part procurement tasks often slow design prototyping. Automation may enable the elimination of some procurement tasks through the introduction of rapid prototyping technologies such as additive manufacturing. During the test phase, data gathering and failure analysis require significant engineering time to sort through data and compare test results against design requirements. The introduction of automation to collect and validate data against design requirements may result in reduced time for tests and improved feedback during follow-on design cycles. Finally, during the deployment phase, data exchange with suppliers and the manufacturing plant is often a cumbersome activity due to the volume and complexity of transferred data. Automating the compilation, validation, and release of data to downstream consumers would reduce the potential for manual errors and rework experienced by the release engineers communicating with the supply base and the plant.

5.2 Why Engineering Automation Works at Ocado

To evaluate the approach that Ocado has taken to engineering automation, we consider why this approach has worked within their organization. From our research, it is clear that much of Ocado’s success in designing and implementing engineering automation centers around the fact that they have applied a software development mindset to hardware development. Nearly all the senior engineering leadership we spoke with emphasized that they were focused on applying the core principles from software engineering to hardware engineering, with the objectives of being “right the first time” and producing products “rapidly and cheaply.”

In addition to a software-centric approach to hardware development, Ocado has benefited from the early adoption of core technologies that enable engineering automation. Specifically, the early adoption of a cloud-based CAD platform in Onshape has been critical to their success in implementing engineering automation. As a cloud-native CAD platform, Onshape has enhanced collaboration through the ease of sharing data within their organization. Onshape also allows easy integration with other design tools through its REST API for other web-based platforms and flexibility in programming language for integration to other legacy, on-premises systems. Finally, Onshape also enables users to develop custom features within the tool using an internal programming language, “FeatureScript.” This has enabled Ocado to build simple internal automations for their CAD tool with minimal effort.

By adopting advanced manufacturing techniques, such as additive manufacturing, Ocado has also reduced the constraints imposed on its product design and increased engineering flexibility during development. This engineering flexibility has enabled Ocado to utilize other advanced design tools, such as topology optimization software, which has allowed them to design parts that would be difficult to manufacture using traditional fabrication techniques (e.g., CNC milling, die casting or injection molding). Using advanced manufacturing and design tools such as additive manufacturing technologies and topology optimization tools underscores Ocado’s commitment to advanced digital technologies and lends itself to automation practices.

5.3 Application of Engineering Automation to Other Organizations

While this approach to applying engineering automation has yielded tremendous results for Ocado, it is interesting to consider the factors that may enable similar successes within corporations of varying organizational archetypes. Attributes of organizational archetypes that may influence the ability to adopt engineering automation workflows include but are not limited to product requirement flexibility, product architecture complexity, organizational size, the digital maturity of the organization, and the average product development team size.

Organizations can be evaluated upon these attributes across their individual dimensional scales. The requirement flexibility across an organization’s product portfolio may range from highly flexible requirements (that are fluid and can be easily adjusted) to extremely rigid requirements (that may require regulatory compliance and are difficult to change). Product architecture complexity may range from low complexity (few components or subsystems) to very high complexity (having many subsystems, outside suppliers, and domain specialization). Organizational size may range from small organizations (local or small businesses with few employees) to large multinational organizations (with thousands of employees spread across many locations and geographies). The digital maturity of an organization also ranges from low maturity (having few digital tools available for product development) to high maturity (having an integrated backbone of digital tools that enable traceability across the organization). Team size can range from small teams (with few people with simple collaboration methods) to very large teams (with many people, functionally oriented and complex collaboration networks).

While all of these attributes can provide insight into an organization’s readiness for adopting engineering automation technologies and processes, we believe that the two most critical attributes to consider when evaluating the applicability of automation within an organization are an organization’s product complexity and digital maturity. Figure 4 shows a matrix of these two attributes and highlights the “sweet spot” for the application of engineering automation. We believe that organizations within the sweet spot of low to medium product architecture complexity and high digital maturity have the highest opportunity to develop engineering automation capabilities. These organizations have high enough product complexity that the investment in automation would benefit their development efforts. Very high-complexity organizations may require significant investment both internally as well as with their external partners (e.g., suppliers and subcontractors) to make automation worthwhile, while very low-complexity organizations may not be producing products of high enough complexity for automation to measurably impact their day-to-day development operations.

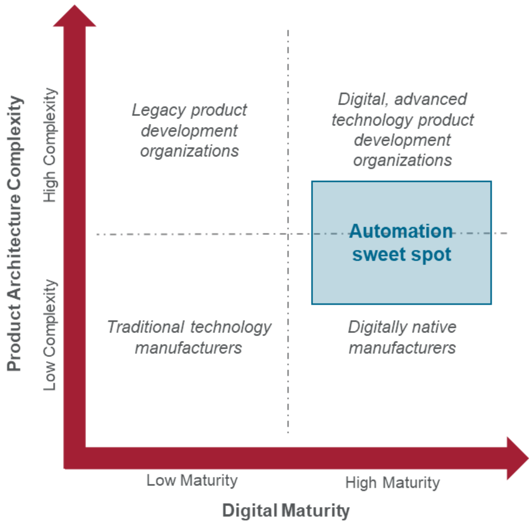


Figure 4: Examining the ideal organizational archetypes for automation

Additionally, a high level of existing digital maturity is necessary for automation. Organizations not already operating within digital engineering environments (e.g., using CAD, PLM, and digital simulation tools) may find it difficult to implement meaningful automation. However, given that some organizations are still in the midst of digital transformations that will likely improve their digital maturity, it may be helpful for organizations to consider if engineering automation is aligned with their organizational objectives and ensure that their target digital maturity will support that automation in the future.

5.4 Consequences to Engineering Automation

While this paper has primarily presented the benefits associated with automating engineering tasks (e.g., speed and productivity), it is essential to acknowledge that such automation may also have negative consequences.

A critical risk to automating tasks is the potential reduction in institutional knowledge required to execute a task manually. As internal competencies erode, challenges may arise in repairing the automation solution or changing the automation solution if necessary. Furthermore, if the automation technology were removed, there may be a lack of employees within the organization who know how to manually perform the previously automated task to keep operations up and running.

A secondary risk posed by automation is creating a false sense of trust and reliance upon organizational automation solutions. Suppose engineers and designers trust automation technology over themselves. In that case, results from automation tools may not be regularly verified, resulting in the potential for non-conforming designs and ideas to be passed on to production with inaccurate data.

We raise these risks not to say that automation should be avoided for these reasons but to point out that automation needs to be carefully applied and that the unintended consequences and emergent organizational behaviors associated with automation should be considered and planned for before introducing automation within any process.

6 Conclusion

Through interviews and observation, we have documented the successful development and adoption of engineering workflow automation within Ocado’s hardware engineering process. Automation of CAD design checks, topology optimization, printed part serialization, and manufacturing pack creation have resulted in both measurable benefits to the speed of their development process as well as intangible benefits that manifest themselves in increased employee engagement. While Ocado has adopted these automation practices for the development of its series 600 robot, other organizations seeking to increase development speed may also consider similar automation approaches for their products.

Ocado took a step-wise approach when selecting where to apply automation in their engineering process, starting with user interviews, followed by value stream mapping, and ending with prioritization and roadmapping of the top candidates for automation. We suggest that other organizations follow a similar approach and apply a specific prioritization method which considers the frequency, duration, and complexity of tasks when deciding which tasks to automate. Additionally, we suggest that organizations look for quick wins and leverage existing technological enablers they have already deployed when considering the ease of automation of any individual engineering task.

As organizations continue to look for ways to increase their speed to market, reduce costs, and increase product quality, automating organizational operations, such as engineering workflows, may become more and more prevalent. We believe there is a need and a significant opportunity for future research in this space, particularly in the exploration of different organizational archetypes where engineering workflow automation may be appropriate. While this study was specifically focused on what automations Ocado employed and how they selected individual tasks for automation, we believe other organizations should evaluate their readiness for adoption of engineering automation based on organizational attributes such as digital maturity and product architecture complexity (among many others).

References

Agile Manifesto (2001). Manifesto for Agile Software Development. [online] Agilemanifesto.org. Available at: https://agilemanifesto.org/.

Atzberger, A. and Paetzold, K. (2019). Current challenges of agile hardware development: What are still the pain points nowadays? Proceedings of the Design Society: International Conference on Engineering Design, 1(1), 2209–2218. https://doi.org/10.1017/dsi.2019.227

Clark, K. B. and Wheelwright, S. C. (1992). Revolutionizing product development: Quantum leaps in speed, efficiency and quality. MacMillan.

Drutchas, J. and Eppinger, S. (2022). Guidance on application of agile in combined hardware and software development projects. Proceedings of the Design Society, 2, 151–160. https://doi.org/10.1017/pds.2022.16

Drutchas, J. and Eppinger, S. (2023) ‘Adjusting Scaled Agile for Systems Engineering’, in Proceedings of the International Conference on Engineering Design (ICED23), Bordeaux, France, 24-28 July 2023. DOI:10.1017/pds.2023.48

Ebert, C. and Paasivaara, M. (2017). Scaling agile. IEEE Software, 34(6), 98–103. https://doi.org/10.1109/ms.2017.4121226

Fitzgerald, B. and Stol, K.-J. (2017). Continuous Software Engineering: A roadmap and agenda. Journal of Systems and Software, 123, 176–189. https://doi.org/10.1016/j.jss.2015.06.063

Hemon, A., Lyonnet, B., Rowe, F., and Fitzgerald, B. (2019). From agile to devops: Smart skills and collaborations. Information Systems Frontiers, 22(4), 927–945. https://doi.org/10.1007/s10796-019-09905-1

Koren, I., Rinker, F., Meixner, K., Matevska, J., and Walter, J. (2023). Challenges and opportunities of DevOps in Cyber-Physical Production Systems Engineering. 2023 IEEE 6th International Conference on Industrial Cyber-Physical Systems (ICPS). https://doi.org/10.1109/icps58381.2023.10128073

Marion, T. J., and Fixson, S. K. (2020). The transformation of the innovation process: How digital tools are changing work, collaboration, and organizations in new product development*. Journal of Product Innovation Management, 38(1), 192–215. https://doi.org/10.1111/jpim.12547

Ocado Group. (2023). 2022 Annual Report. Ocado Group. https://www.ocadogroup.com/investors/annual-report

O’Connor, P. (1994). Implementing a stage-gate process: A multi-company perspective. Journal of Product Innovation Management, 11(3), 183–200. https://doi.org/10.1111/1540-5885.1130183

Peterson, M., and Summers, J. D. (2021). Recommended methods supporting adoption of the agile philosophy for hardware development. Volume 6: 33rd International Conference on Design Theory and Methodology (DTM). https://doi.org/10.1115/detc2021-70621

Rigby, D., Sutherland, J., and Takeuchi, H. (2021, August 27). The secret history of agile innovation. Harvard Business Review. https://hbr.org/2016/04/the-secret-history-of-agile-innovation

Schrof, J. I., Atzberger, A., Papoutsis, E., and Paetzold, K. (2019). Potential of technological enablement for Agile Automotive Product Development. 2019 IEEE International Conference on Engineering, Technology and Innovation (ICE/ITMC). https://doi.org/10.1109/ice.2019.8792665

Schwaber, K. and Sutherland, J. (2020). The Scrum Guide The Definitive Guide to Scrum: The Rules of the Game. [online] Available at: https://scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-US.pdfa#zoom=100.

Singh, V. and Willcox, K.E. (2018). Engineering design with digital thread. AIAA Journal, 56(11), 4515–4528. https://doi.org/10.2514/1.j057255

Smeds, J., Nybom, K., and Porres, I. (2015). DevOps: A definition and perceived adoption impediments. Lecture Notes in Business Information Processing, 166–177. https://doi.org/10.1007/978-3-319-18612-2_14

Stirling, J., Bumke, K., Collins, J., Dhokia, V., and Bowman, R. (2022). HardOps: Utilising the software development toolchain for hardware design. International Journal of Computer Integrated Manufacturing, 35(12), 1297–1309. https://doi.org/10.1080/0951192x.2022.2028188

Vidner, O., Wehlin, C., and Wiberg, A. (2022). Design Automation Systems for the Product Development Process: Reflections from five industrial case studies. Proceedings of the Design Society, 2, 2533–2542. https://doi.org/10.101

Contact: Peters, Michael S., Massachusetts Institute of Technology, Cambridge, MA, United States of America, mspeters@mit.edu