OLIVER WYMAN

AGILE ENGINEERING

THE CASE FOR CHANGE IN AUTOMOTIVE R&D

MARSH   GUY CARPENTER   MERCER   OLIVER WYMAN

MARSH & McLENNAN
COMPANIES

WHY AGILE ENGINEERING?

Traditionally, engineering has been at the heart of automotive companies, both in original equipment manufacturing (OEM) and supplier organizations. Today, however, R&D departments face intense pressure on two fronts:

On the output side, the challenges of ever increasing complexity, shorter and heterogeneous product lifecycles, increased competition from non-automotive players, dynamic regulations, multiple technology shifts, and tight program deadlines have become part of the daily travails of R&D executives worldwide.

On the input side, the need to cut down on costs (sometimes by as much as 30 percent), a "ceiling" for in-house R&D capacity, an increasingly scarce supply of talent, and a lack of resources in critical "new" competences are equally worrisome.

Moreover, there is the inherent risk that the managers currently in place – many of them "petrolheads", enthusiastic about traditional vehicle engineering – are missing the needs of the next generation of car buyers. Specification books are typically based on the predecessor and on those of traditional competitors, and are written three to four years before a product enters the market. Moreover, the link with current and future customers is diluted by organizational silos, international communication barriers, and a rigid hierarchy.

Automotive engineering is increasingly coming under pressure – Agile engineering is a solution for many current R&D challenges.

AGILE WORKING MODEL

Agile engineering has been the norm in software-developing organizations for three decades. It has gained popularity in recent years, as digital MVP (minimum viable product) prototyping for apps and services has become common across industries. Software development teams benefit substantially from the process's customer-centricity, flexible capacity allocation, high process transparency, and easy organizational learning. Agile working environments have become an important asset in attracting top talent, as many young engineers want to see the impact of their work daily and directly in a small team, and enjoy the diversity of flexible job allocations. These benefits would seem to be relevant and advantageous for "hardware-developing" organizations too. Why then are they trailing behind?

HURDLES FOR AGILE ENGINEERING

There are, however, substantial obstacles to implementing agile working principles on the product side. These include relatively high cost/long lead times for prototype parts, existing supplier relationships and business models, product architectures and "modular kits" designed in traditional waterfall approaches, and a limited and costly rollout opportunity for most of the components and systems in scope. It is quite clear that the pure-play software agile working model may not be suited to industrial and manufacturing organizations, and will need to be adapted. This is why efforts to transform "hardware" engineering organizations have found little success.

EXAMPLES FOR AGILE ENGINEERING IN AN INDUSTRIAL "HARDWARE" WORLD

Over the past years, numerous startup companies and creators of various devices have worked agile engineering into their product development processes. Many industrial organizations are now learning from those pioneers, placing special emphasis on individual product aspects and focusing on mixed software/hardware environments. Many organizations want to try out agile principles but not risk the company and its full product pipeline.

Yet by focusing only on individual aspects of agile, organizations risk missing its full benefits. Agile, if implemented just in bits and pieces, risks becoming just a buzzword or, worse, a new label for business as usual.

Exhibit 1: Many industrial companies have already started to implement agile in a mixed hardware/software world

AGILE PRINCIPLES	ADOPTED PRINCIPLES OF COMPANY					
	Railway	Automotive	Startup	Communication	Industry	Electronics
Customer satisfaction	●	●				●
Welcome changing requirements	●			●		
Daily team cooperation	●	●	●	●		●
Frequent HW/SW delivery			●	●		●
Feature/end product orientation	●					
Good design focus						●
Face-to-face conversation			●	●	●	●
Simplicity			●			
Reflection				●	●	
Self-organizing teams		●	●	●	●	●
Motivated individuals		●	●	●	●	●
Sustainable development		●		●		

EXAMPLES FOR COMPANIES ADOPTING AGILE

RAILWAY MANUFACTURER
Currently developing a concept to introduce agile across **all engineering and manufacturing departments**

AUTOMOTIVE
Set-up a start-up platform within the firm (**incubator for new business ideas**). Implemented **innovation clusters**, for e.g., connected industry or mobility

STARTUP
Startup that innovates by applying **scrum development techniques** Use **open source tools** and **lean mgmt.** methods to improve productivity

COMMUNICATION
Set-up a "**physical internet lab**" unit to **develop HW daily** (prototypes and products, e.g., connected objects)

INDUSTRY
Created **self-directed work structure** ("Teaming") across entire supply chain

ELECTRONICS
Implemented a **scrum team pilot** to significantly speed up development of **new integrated circuit boards**

Source: Oliver Wyman research and analysis

THE CHALLENGE
AGILE ENGINEERING ECOSYSTEM

Given these challenges, we believe that automotive firms in pursuit of change will need to cultivate a completely new engineering ecosystem internally. In the process, agile will affect three major dimensions of R&D: organization, product, and engineering system.

While the scope of this article prevents us from going into greater detail, certain key questions arise up quickly when thinking about an agile R&D world in the automotive industry:

ORGANIZATION

- How can "agile" and "traditional" engineering teams collaborate inside the organization?
- How can the performance management system be adapted to include agile teams?
- Are there different career models for agile vs. non-agile engineers?
- How to collaborate with suppliers? Will they be an integral part of agile engineering teams?

PRODUCT

- How are products inherently structured to reflect the responsibilities of engineering teams?
- How will the product architecture change to reflect agile engineering?
- What new product interfaces are required?
- What will be the impact on product lifecycle?

ENGINEERING SYSTEM

- How will the product development process evolve into agile sprints?
- Are the engineering tools, such as PLM, prepared to support agile teams?
- How will handovers be managed?

We summarize some of our experience in the illustration below, but clearly the answers to the above questions will be different for every organization. Agility can be implemented only if the links between organization, product and engineering system are understood and the migration path reflects these links.

Exhibit 2: Agility means profound changes for the R&D working model

MAJOR TARGETS OF AGILITY

ORGANISATION
Structures & roles
Governance & performance management
People & skills

- Supplier integration allowing 100% open mixed teams/co-development
- R&D governance/KPI/HR model adapted to agile teaming requirements
- Physical infrastructure supporting agile teams (e.g,. co-location)

PRODUCT
Product architecture and interfaces
Software architecture

- Functions/layers as the main product structure framework
- Fully modularized vehicle/electronics architecture to allow de-coupling of innovation cycles
- Fully standardized technical interfaces

ENGINEERING SYSTEM
Processes
Systems and tools

- PLM/engineering tools supporting agile processes (e.g., handover points, sprint planning, backlog)
- Fully virtualized engineering, validation and sign-off process
- Defined and approved interfaces to non-agile parts of PDP (e.g., vehicle platforms)

Source: Oliver Wyman

WAYS TO IMPLEMENT AGILE ENGINEERING

Given these interdependencies, there is no "cherry-picking" of individual elements of agile for R&D organizations, but only a limited number of paths that can be followed. Organizations may opt for individual process steps, permanent agile teams, agile vs. non-agile programs or even an all-out agile re-organization of the R&D function.

INDIVIDUAL PROCESS STEPS

In this implementation strategy, the traditional hierarchies continue to exist for the most part. Developers/engineers work in agile teams for specific durations in projects (agile

Agility means profound change to not only engineering organizations, but also the products they develop and the systems and processes supporting their daily business.

working mode), after which they return to their traditional roles in the organization. Early phase concept work, for example, often can be done in agile teams. Chances are, however, that either the agile teams will not fully embrace this methodology or – if the process step takes long – the return to waterfall projects will produce frustrations.

PERMANENT AGILE TEAMS

In this strategy, specific parts of the organization chart permanently become agile teams. These could be, for example, software teams or predevelopment teams. The key challenge here is to define the handover (time, product, maturity, deliverables) from agile to traditional teams. Moreover, permanently running two different organization models in parallel will require much effort from management and the HR function to ensure that things continue to run smoothly.

AGILE VS. NON-AGILE PROGRAMS

This is probably the most common implementation strategy. While selected development programs turn agile, others do not. Appropriate programs can be identified based on their technology content, maturity, proximity/interfaces to traditions parts of the organization, supplier involvement and ability to plan the program right and completely from the start. Here, parts of the R&D function remain waterfall and hierarchy driven. As the implementation progresses and new programs get started, the share of agile programs will probably expand to a point where the majority – if not all – programs are working in an agile setup.
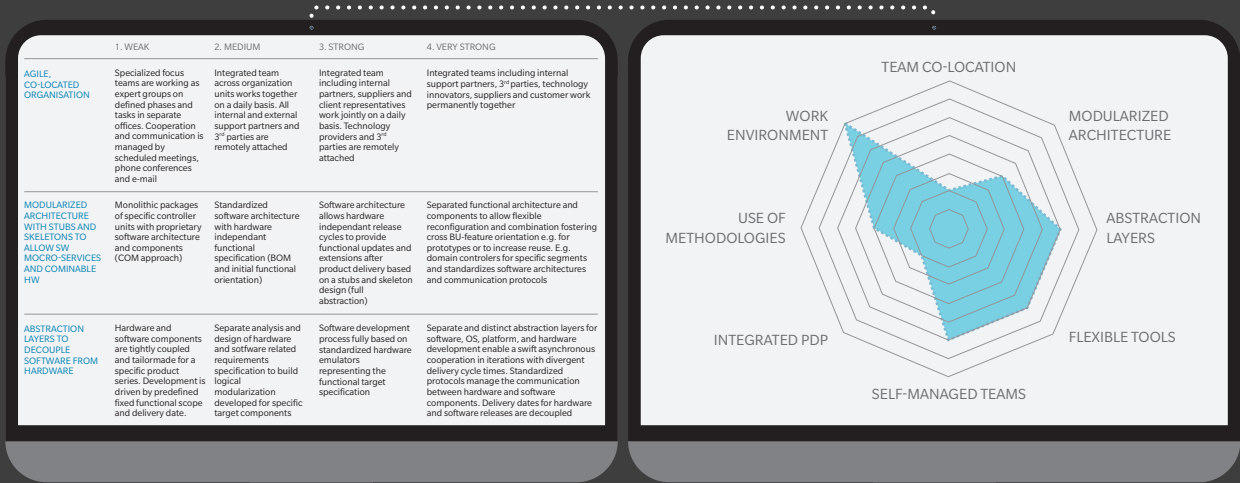
ALL AGILE

Few companies try to do this at once, because there will always be running programs and activities for which any change produces delays. A complete re-organization can be costly in terms of time, money and attrition. It can probably be successfully executed only if there are additional reasons for re-organization (such as a relocation, post-merger integration, and massive cost-down program).

Based on their ambition and target picture, R&D organizations should select the appropriate implementation strategy. However, they need to be aware that even the target state may evolve as things progress.

HOW TO GET GOING

From our experience, we suggest two key items to run through when planning for organizational change toward agility. First, a thorough diagnosis of the "as-is-status" will be necessary. Organizations will have to distinguish between buzzword vs. real organizational practice, and develop a clear-cut plan for implementation built upon this diagnosis and reflecting the individual organization's ambition.

ORGANISATION

PRODUCT

ENGINEERING SYSTEM

	1. WEAK	2. MEDIUM	3. STRONG	4. VERY STRONG
AGILE, CO-LOCATED ORGANISATION	Specialized focus teams are working as expert groups on defined phases and tasks in separate offices. Cooperation and communication is managed by scheduled meetings, phone conferences and e-mail	Integrated team across organization units works together on a daily basis. All internal and external support partners and 3rd parties are remotely attached	Integrated team including internal partners, suppliers and client representatives work jointly on a daily basis. Technology providers and 3rd parties are remotely attached	Integrated teams including internal support partners, 3rd parties, technology innovators, suppliers and customer work permanently together
MODULARIZED ARCHITECTURE WITH STUBS AND SKELETONS TO ALLOW SW MOCRO-SERVICES AND COMINABLE HW	Monolithic packages of specific controller units with proprietary software architecture and components (COM approach)	Standardized software architecture with hardware independant functional specification (BOM and initial functional orientation)	Software architecture allows hardware independant release cycles to provide functional updates and extensions after product delivery based on a stubs and skeleton design (full abstraction)	Separated functional architecture and components to allow flexible reconfiguration and combination fostering cross BU-feature orientation e.g. for prototypes or to increase reuse. E.g. domain controlers for specific segments and standardizes software architectures and communication protocols
ABSTRACTION LAYERS TO DECOUPLE SOFTWARE FROM HARDWARE	Hardware and software components are tightly coupled and tailormade for a specific product series. Development is driven by predefined fixed functional scope and delivery date.	Separate analysis and design of hardware and software related requirements specification to build logical modularization developed for specific target components	Software development process fully based on standardized hardware emulators representing the functional target specification	Separate and distinct abstraction layers for software, OS, platform, and hardware development enable a swift asynchronous cooperation in iterations with divergent delivery cycle times. Standardized protocols manage the communication between hardware and software components. Delivery dates for hardware and software releases are decoupled

Radar chart axes:
TEAM CO-LOCATION
MODULARIZED ARCHITECTURE
ABSTRACTION LAYERS
FLEXIBLE TOOLS
SELF-MANAGED TEAMS
INTEGRATED PDP
USE OF METHODOLOGIES
WORK ENVIRONMENT

OLIVER WYMAN APPROACH

Run initial diagnosis to assess the organization unit's current agility.

Utilize a combination of:
- Interviews with project stakeholders (R&D team, pilot teams)
- Diagnosis workshops
- Reviewing pilot documentation, organization, time and resource plan
- Outside-in view of Oliver Wyman experts (benchmarks, best practices)

Every transformation journey should be based on honest and thorough analysis, and reflect guiding principles derived from successful programs.

For the diagnosis, Oliver Wyman has developed a tool which does not only identify the "agility status" of an organization, but also its most pressing "pain points".

The diagnosis should systematically compare the organization unit's agility vs. external benchmarks and best practices, but also discuss stakeholders' experiences and concerns. Quite often, the "internal clients" of R&D organizations are the ones that most need to be convinced of the benefits of change. The case for change and the relevant options for redesigning should reflect the views of the stakeholders as well as the engineers' and program leaders' own opinions.

Based on the diagnosis, an organizational transformation program can be designed individually. From our experience, any transformation program should be built on the following principles:

1. **Agile target picture itself needs to be developed in several iterations.** Given the overall complexity of the engineering ecosystem "as-is" and the target agile state, from our experience, organizations are typically not able to foresee all the implications on day one. Much as the engineering work itself, the agile transformation program will need to be planned in several sprints. This does not mean, however, that the process should lack clear objectives: we suggest a clear-cut target picture, based on measurable objectives, for each and every sprint. Obviously, the next sprint's target state will be described in much more detail than the later ones.

2. **Quick and robust first step should be based on selected piloting approach.** As in any transformation program, momentum will have to be built up, and early, tangible successes will be a precondition for stakeholders' buy-in. Piloting might be an appropriate way to achieve this. Based on a quick and efficient drawing of the overall target, much effort should be spent on designing the first transition step, the measurement of success, and the communication with stakeholders. A realistic first program will be key to starting the journey.

3. **"R&D shift" transition approach works for many automotive companies.** As discussed above, a "program-by-program" selection and shift to agile working models will work well in many organizations that cannot afford to lose time. Shrewd selection of programs, staffing them with program leaders that believe in the benefits of agile, and parallel coaching by experts will be essential for success.

CONCLUSION

Agile principles can be the answer to many pressing needs in the R&D functions of automotive and other manufacturing companies. The transformation, however, is not a quick or easy fix, as it requires deep organizational, product, and engineering system changes. Given that the transformation journey can easily take between three and five years, it is best to start thinking seriously now, pushing forward constantly, to be ready for the future. While there are different ways to implement an agile working model, we suggest proven diagnosis tools and transformation program principles that can guide engineering organizations as they migrate to the next level.

Oliver Wyman is a global leader in management consulting that combines deep industry knowledge with specialized expertise in strategy, operations, risk management, and organization transformation.

Oliver Wyman's global Operations Practice specializes in end-to-end operations transformation capabilities to address costs, risks, efficiency, and effectiveness. Our global team offers a comprehensive and expert set of functional capabilities and high-impact solutions to address the key issues faced by Chief Operating Officers and Chief Procurement Officers across industries.

AUTHORS

DR. JUERGEN REINER

Partner, Germany
juergen.reiner@oliverwyman.com

SIMON SCHNURRER

Partner, Germany
simon.schnurrer@oliverwyman.com

LARS STOLZ

Partner, Germany
lars.stolz@oliverwyman.com

CONTACTS

LARS STOLZ

Global Head of Operations
lars.stolz@oliverwyman.com

GREG KOCHERSPERGER

Europe, Middle East and Asia Head of Operations
greg.kochersperger@oliverwyman.com

JOHN SEELIGER

America Head of Operations
john.seeliger@oliverwyman.com

XAVIER NOUGUÈS

Head of Value Sourcing
xavier.nougues@oliverwyman.com

MICHAEL LIEROW

Head of Supply Chain
michael.lierow@oliverwyman.com

BRIAN PRENTICE

Head of Manufacturing and Process Operations
brian.prentice@oliverwyman.com

ALEX LYALL

Head of Digital Operations
alex.lyall@oliverwyman.com

www.oliverwyman.com

OLIVER WYMAN