Aberdeen Group
A Harte-Hanks Company

System Engineering

Top Four Design Tips to Increase Profit Margins for Mechatronics and Smart Products

October 2009

Michelle Boucher

Executive Summary

The current economy has driven many companies to look for ways to take cost out of their products and make their products even more appealing with more feature rich, intelligent products. This is driving manufacturers to improve their system engineering processes to enable them to develop products consisting of mechanical components, electronics, and embedded software. These products offer tremendous opportunities in terms of greater profitability when system engineering is done well, but when executed poorly, excessive cost is a significant risk. This report offers guidance to implement successful system engineering practices that will lead to greater profitability and avoid the risks of excessive costs.

Research Benchmark

Aberdeen's Research Benchmarks provide an in-depth and comprehensive look into process, procedure, methodologies, and technologies with best practice identification and actionable recommendations

Best-in-Class Performance

Aberdeen used four key performance criteria to distinguish Best-in-Class companies. When compared to the Industry Average, the Best-in-Class are:

- Earning profit margins that are 2.3-times higher
- Taking three-times more cost out of products
- 20% more likely to meet product launch dates
- Experiencing development cycles that are 6.2-times shorter

"Companies that desire to be Best-in-Class should consider better requirements definition early on. This includes all aspects of requirements management such as better process, tools, workflows, ownership, traceability, change management, and ability to use multiple formats. Requirements should also be linked to customer needs, which reduces drastic scope change later on. By achieving this desired state, we believe it will lead us to have more competitive solutions for our customers and more predictability in development costs and timelines (fewer overruns)."

~ Maryane Chapman, Director, Integrated Systems Engineering, Pitney Bowes

Competitive Maturity Assessment

When compared to competitors, firms enjoying Best-in-Class performance share several common characteristics that support system engineering including:

- 40% more likely to evaluate design alternatives on multiple criteria
- 39% more likely to use system modeling to verify that design requirements have been met
- 36% more likely to link customer needs to requirements

Required Actions

In addition to the specific recommendations in Chapter Three of this report, to achieve Best-in-Class performance, companies must:

- Use multiple design criteria to define system architecture and add that criteria to the system requirements
- Requirements should be linked to higher level system functions as well as to the overall customer need it meets
- Leverage a model driven design approach to overcome communication barrier and verify requirements have been met

Table of Contents

Figures

Tables

Telephone: 617 854 5200
Fax: 617 723 7897

Chapter One: Benchmarking the Best-in-Class

One of the top pressures facing engineering executives today is increasing market demand to build "smarter" product as found in Aberdeen Group's June 2008 report, *Engineering Executive's Strategic Agenda*. Further supporting this growing need, Aberdeen's November 2008 report, *Engineering Evolved: Getting Mechatronics Performance Right the First Time,* found that 66% of the products developed last year contained embedded systems. Clearly developing smart products is an important trend manufacturers must pay attention to in order to be competitive in today's market. However, it is inherently difficult, and getting it right requires new approaches to developing products. One of these approaches requires making system engineering a core engineering discipline. To understand successful system engineering approaches, Aberdeen studied the experiences of 150 companies from August to October 2009 to understand how which systems engineering practices result in bringing in more revenue while keeping costs down and staying on budget.

Fast Facts

When compared to the Industry Average, successful system engineering enables Best-in-Class companies to:

√ Earn profit margins that are **2.3-times** higher

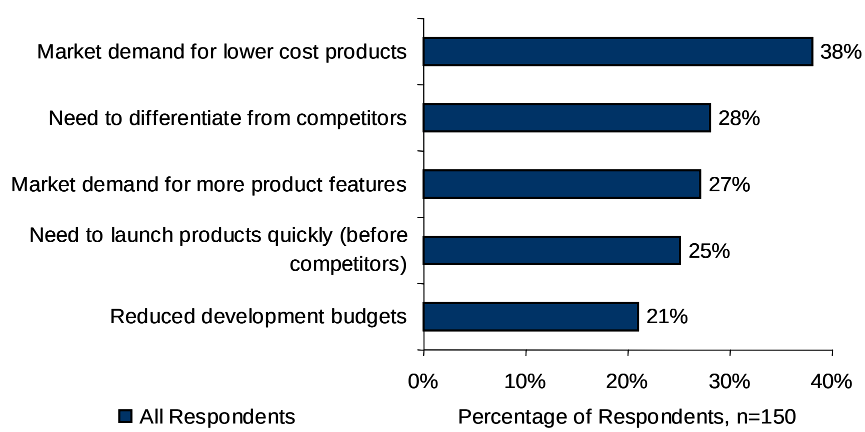
√ Take **three-times** more cost out of products

√ Experience development cycles that are **6.2-times shorter**

The Business Need for System Engineering

To understand the external factors affecting system engineering, respondents were asked to pick the top two pressures driving them to improve system engineering processes. The top pressures all indicate a focus on customers (Figure 1).

Figure1: Top Business Pressures Driving System Engineering Improvement



Source: Aberdeen Group, October 2009

The number one pressure on organizations is the need to take cost out of products. Clearly this is a reflection of the current economy that indicates the market has become more cost conscious. Companies are recognizing the cost saving opportunities of replacing mechanical components with software. However, they are realizing that to be successful, they need to be good at system engineering and this is driving improvement initiatives.

The second and third pressures show a need to align products with what customers want by differentiating them and providing more features. Companies are adding embedded software as a way to do this, but they are realizing this is also requiring improvements in system engineering to be successful. Launching before competitors is a time to market pressure which means companies are looking for better system engineering practices to improve the efficiency of their development process so that they can get their products to market first and capture market share.

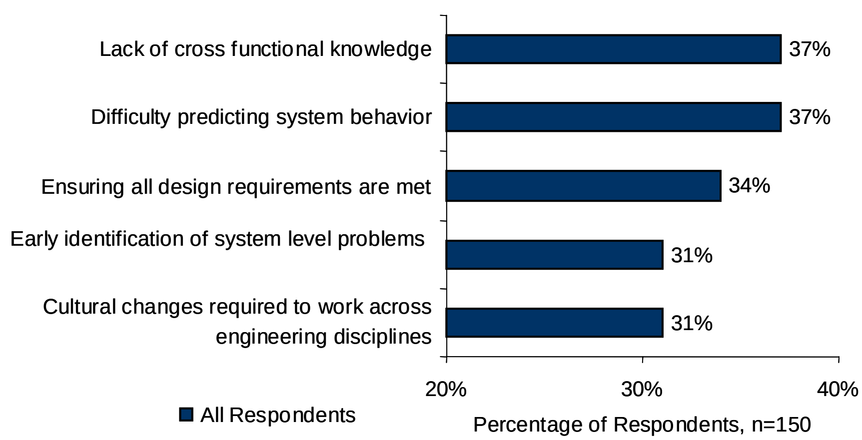
Why is System Engineering Difficult?

Obviously the pressures driving improvements in system engineering demonstrate the effect system engineering has on the top and bottom line of a company. However, what makes system engineering so difficult in the first place? Figure 2 displays the top challenges of system engineering. Respondents were asked to pick their top three challenges.

"Do a good job initially on developing customer requirements. That makes verification and validation much easier.

~ James Lipscomb, President, X-Bar Diagnostic Systems, Inc

Figure 2: Top Challenges of System Engineering



Source: Aberdeen Group, October 2009

The top challenges fall into three themes. The first is associated with bringing together a team of engineers from different engineering disciplines. With a team consisting of mechanical, electrical, and software engineers, there are natural silos of knowledge that must be overcome in order for them to work together to develop an integrated system. The second theme is predicting how the system will behave and identifying problems with system behavior as early as possible. Because the different components are all designed in tools developed for specific engineering disciples, it is very difficult to get much insight into how the system will behave when it is put together. When problems are not found until the first physical prototype is built, it is often very late in the development cycle when it is far more expensive and time consuming to make changes and corrections. The final

"Communication between engineering disciplines is key. Design / technical reviews across engineering teams in addition to those within teams help uncover issue earlier in the development phase."

~ Thomas Wright, Vice President of Engineering, ATSI Holdings, Inc and third theme is making sure the product that was originally intended is actually what is designed and build. System level problems found late in the process often require sacrificing requirements just so that the system will work. In addition, simply not understanding the requirements means they will not be implemented correctly. This confusion is even more likely when requirements cross multiple engineering disciplines.

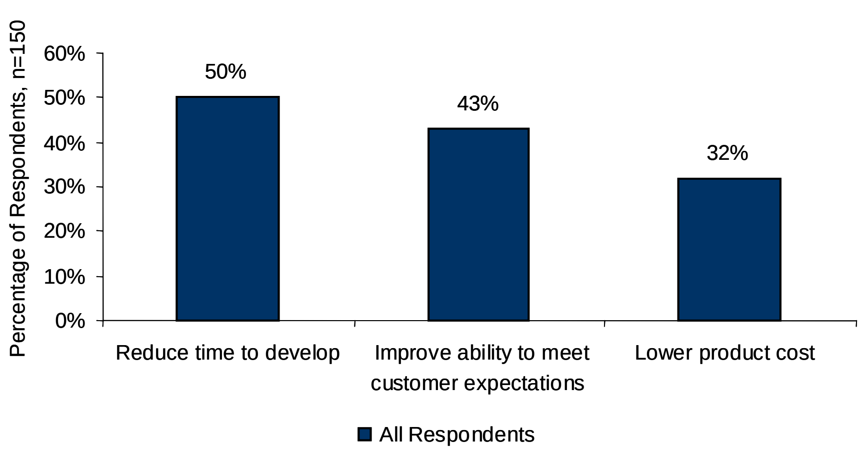
What is interesting is that all of these challenges point to things that are solved with good system engineering practices. System engineers have some expertise in each of the engineering disciplines involved so they are in a good position to bridge the lack of cross functional knowledge. They can use this expertise to make good decisions about the system architecture and plan the system requirements. By doing this work up front, the system is more likely be to designed and built as originally intended and it will be easier to validate that the requirements are met throughout the development process. With a better understanding of the requirements, it is also easier to catch problems with the design earlier in the process.

The Maturity Class Framework

To understand successful system engineering practices and the business impact it has upon companies, Aberdeen benchmarked the performance of study participants and categorized them as either Best-in-Class (top 20% of performers), Industry Average (mid 50%), or Laggard (bottom 30%).

To ensure that organizations are categorized according to the criterion that most accurately captures what organizations are trying to accomplish, Aberdeen first identified the top business objectives for improving systems engineering (Figure 3).

Figure 3: Top Business Objectives for System Engineering Improvements



Source: Aberdeen Group, October 2009

Based on these answers, four key performance measures that indicate success with achieving these objectives were used to distinguish the Best-in-Class from Industry Average and Laggard organizations. The performance of each of these tiers is displayed in Table 1.

Table 1: Top Performers Earn Best-in-Class Status

Definition of Maturity Class	Mean Class Performance
Best-in-Class: Top 20% of aggregate performance scorers	▪ 83% of products met product launch deadlines ▪ 13% reduction in development time ▪ 10% reduction in product cost ▪ 8% increase in product profit margins
Industry Average: Middle 50% of aggregate performance scorers	▪ 69% of products met product launch deadlines ▪ 2% reduction in development time ▪ 5% increase in product cost ▪ 3% increase in product profit margins
Laggard: Bottom 30% of aggregate performance scorers	▪ 36% of products met product launch deadlines ▪ 10% increase in development time ▪ 13% increase in product cost ▪ 3% decrease in product profit margins

Source: Aberdeen Group, October 2009

"To facilitate collaboration and take full advantage of the expertise of the entire development team, have people in physical proximity to each other, or have frequent face to face working sessions plus very frequent networked meetings. Really reward cooperation and collaboration among team members, especially those from different disciplines. Celebrate successes and failures alike."

~ Engineering Manager, High Tech Company

By meeting scheduled release dates, the Best-in-Class are able to address time to market pressures and can therefore start recognizing revenue sooner. This means they are better positioned to capture market share from competitors.

Also addressing the top objective of reducing development time, the Best-in-Class have been able to improve their efficiency and as a result have reduced development time over the last two years by 13%. Compared to the 10% increase in development time seen by Laggards, the Best-in-Class are clearly at a competitive advantage. They can start enjoying a return on their development investment much sooner than their competitors.

The Best-in-Class have also been extremely successful with taking cost out of their products. In fact, over the last two years, they have been able to remove three-times more cost than the Industry Average. By taking cost out of their product, they can address the top pressure driving system engineering improvement by meeting market demand for lower cost products and lower their prices. Alternatively, they can keep prices the same and enjoy higher profitably with larger profit margins.

Getting products to market on time and taking cost out of products enables the Best-in-Class to enjoy higher profit margins. However, even more important, this also indicates that they are doing a superior job of understanding what their customers want and delivering that. By releasing a product that aligns to what customers want, there is more demand for the product. Consequently, the Best-in-Class can charge a premium for their products that results in higher profit margins and greater profitability. In fact, their profit margins are 2.4-times higher than the Industry Average.

Clearly there is a lot of opportunity for success and greater profitability with good system engineering practices. On the flip side, if it is not done well, it can be extremely expensive for a company and puts them at a competitive disadvantage. The question is, what practices have those Best-in-Class companies deployed that lead to their success?

The Best-in-Class PACE Model

Using system engineering to achieve corporate goals requires a combination of strategic actions, organizational capabilities, and enabling technologies that are summarized in Table 2.

Table 2: The Best-in-Class PACE Framework

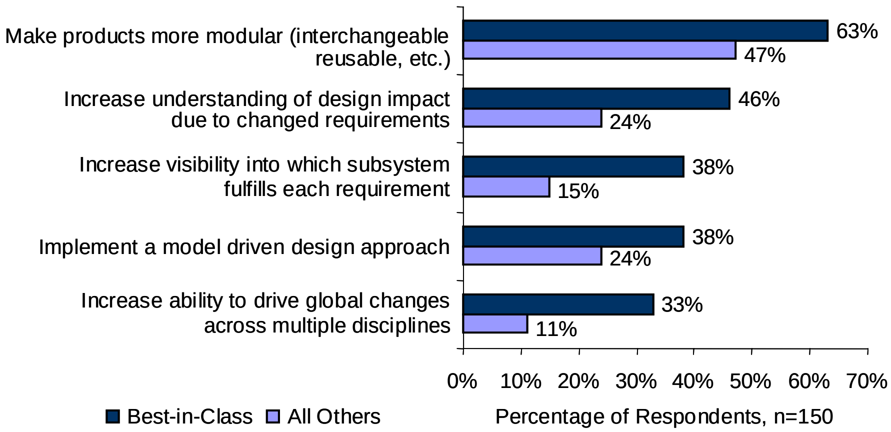
Pressures	Actions	Capabilities	Enablers
▪ Market demand for lower cost products	▪ Increase understanding of changed requirements on design ▪ Increase visibility into which subsystem fulfills each requirement	▪ System functions mapped to system requirements ▪ Trade-off studies investigate architecture alternatives ▪ Requirements verified using models ▪ Customer need linked to requirements ▪ Requirement verification status centrally managed ▪ Owner defined for overseeing each requirement ▪ System engineer provides input into verification tests ▪ Design alternatives evaluated on multiple criteria ▪ Product performance criteria defined prior to development work ▪ Performance criteria defined within system requirements	▪ Requirements management solution ▪ Simulation tools ▪ Document Management ▪ Integrated Product Data and Requirements Management ▪ Product Data Management (PDM) ▪ Product Lifecycle Management (PLM)

Source: Aberdeen Group, October 2009

Best-in-Class Strategies

Given the higher levels of profitability the Best-in-Class enjoy, they are clearly doing a better job of addressing the pressures driving improvements in system engineering and its associated challenges. They are doing this by improving system engineering and this is reflected in the strategies they are deploying to support their system engineering initiatives.

Figure 4: Best-in-Class Strategies to Improve System Engineering



Source: Aberdeen Group, October 2009

"Modular design supports system design by enabling reuse which can reduce development costs and timelines."

~ Maryane Chapman, Director, Integrated Systems Engineering, Pitney Bowes

The Best-in-Class are far more likely than their competitors to deploy a number of strategies. All of these strategies support their ability to simplify the complexity of system design. This helps them manage the development process and requirements better so that they can be sure they deliver the products that customers want that will offer the revenue opportunities they are seeking. This also helps them adapt to changes as well so that they are implemented correctly and do not cause downstream problems later on when everything was not changed as required.

The most popular strategy for the Best-in-Class is to make products more modular. By making them more modular, it is easier to reuse components. Not only does this save development time, but it means proven subsystems can be used again which reduces the risk of finding system level performance problems later on.

The Best-in-Class put a lot of focus on planning the system. They are 2.5 times more likely than their competitors to ensure they have visibility into which subsystem will fulfill each requirement. This ensures each requirement has been mapped to a subsystem, helping them address the challenge of ensuring design requirements are met in the final product. This also helps them understand how changes to the requirements affect the design. This allows them to better evaluate the impact of a change and ensure that the change is implemented correctly.

Another way the Best-in-Class simplify the complexity of the design is by implementing a model driven approach. This provides a visual reference for the system design that makes it easier for all engineers on the team to understand the system definition, which makes it easier for the different engineering disciplines to overcome the lack of cross functional knowledge, one of the top challenges of system engineering.

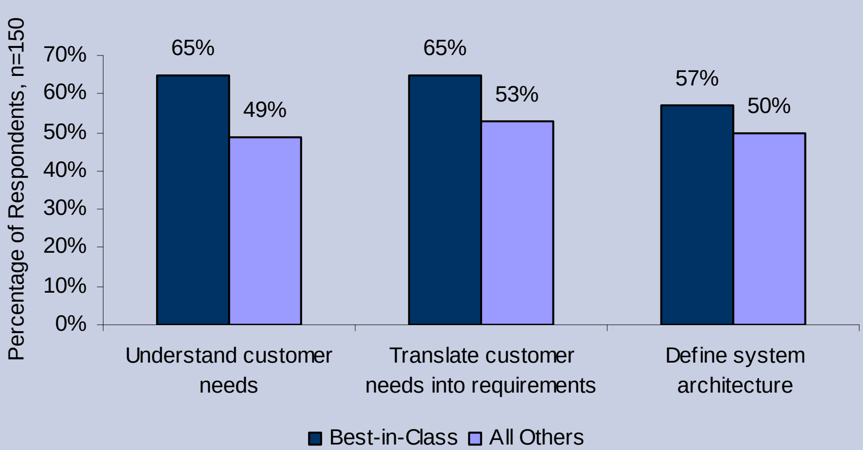
Changes are inevitable in the development process. To successfully implement a change requires understanding its full impact and changing everything that is affected. This is very difficult when making changes to a system consisting of interconnected components that cross multiple engineering disciplines. Often this results in changes that are only partially implemented which leads to problems found later on in the development process, another top challenge of system engineering. To address this, the Best-in-Class are three times more likely to increase their ability to drive global changes across multiple engineering disciplines. The other strategies they have implemented to identify how the requirements map to the subsystems also support their ability to execute this strategy.

Aberdeen Insights — Strategy

The role of a systems engineer has become increasingly important in the development of modern products as they evolve into integrated systems of mechanical components, electronics, and embedded software. However, it is not easy to find systems engineers. More and more colleges are offering Systems Engineering as a degree, but it will take time to address the increased demand for them. Even still, it takes time for them to become experienced enough to run a complete project. Survey responses show that systems engineers are more likely to be mechanical or electrical engineers or even project mangers with the cross functional experience that has enabled then to become systems engineers.

Once a system engineer is hired, what is the best way for the engineering executive to leverage his or her expertise to justify the investment in the position? Overall, most companies use a systems engineer to define the system architecture (Figure 5).

Figure 5: Responsibilities of a System Engineer



Source: Aberdeen Group, October 2009

continued

"Focus on the product requirements through-out the development and don't let scope creep happen."

~ Engineering Manager, Industrial Equipment Manufacturer

Aberdeen Insights — Strategy

They also use them to manage the development project (60%), facilitate collaboration between the engineering disciplines (51%), optimize the design (51%), and define requirements (50%). However, something the Best-in-Class are 33% more likely to do than their competitors is have the systems engineer focus on understanding customer needs.

As reflected in the higher profit margins they enjoy, the Best-in-Class are far more successful at meeting the needs of their customers. One of the keys to this is having system engineers really understand what the customer wants. This allows them to do a much better job translating customer needs into requirements. It also puts them in a better position to make the right decisions about the system architecture. With everything tied to what customers want, it is easier for the engineers to understand how the requirements fit into the whole design, which increases the chances of the final design actually meeting the design requirements, a top challenge of system engineering.

By hiring a systems engineer and ensuring they start the design process with a focus on understanding customer needs, the engineering executive can turn this position into a very strategic one that will make sure products are tightly aligned to customer needs, yielding higher profit margins and leading to greater profitability.

In the next chapter, we will see what the top performers are doing to achieve their performance advantages.

Chapter Two: Benchmarking Requirements for Success

Chapter One demonstrated how the pressures and challenges driving improvements in system engineering as well as the business opportunities offered by implementing Best-in-Class system engineering strategies. Chapter Two explores the practices and enabling technologies the Best-in-Class deploy to execute those strategies.

Fast Facts

Compared to all competitors, the Best-in-Class are:

√ **40%** more likely to evaluate design alternatives based on multiple criteria

√ **39%** more likely to use models to verify design requirements have been met

√ **36%** more likely to link customer needs to requirements

Case Study — Danzco Inc.

Danzco Inc. manufactures unique products that solve design problems or offer improved performance over common products for logging, construction, hydraulics, machining, mechanical power transmission, and cold weather starting diesel engines. They specialize in small quantities of these specialized products. They pride themselves with the special attention they give their customers by focusing on their customer's success. It is this focus that they attribute to their own success.

It was this strong alignment to their customer's needs that drove them to incorporate new technologies into their products. Their customers needed products that cost less to operate, were easier to use, and had lower lifecycle costs. "In this economy, it is more important than ever to meet needs no one else is meeting," says Ed Danzer, General Manager at Danzco. "A me-too product won't cut it right now. You need to think out of the box." It was this realization that drove them to look to improve their ability to develop products that contain an integrated system of mechanical components, electronics, and software.

"One of the challenges was that we did not have all the design tools needed for the development of an integrated system or the skill sets," observed Danzer. "We invested in simulation tools to help us manage the complexity of the design and have better visibility into product behavior." As a result, they were able to take on new business and have identified an area for potential expansion – an impressive and exciting accomplishment in this economy.

"While the tools can get you a long way, they are not the silver bullet by themselves," warns Danzer. "The software helps make better products if properly done, but garbage in, equals garbage out. There is still a learning curve for the tools and you have to know what you are doing." The lesson is, when done well, system engineering processes offers new business opportunities, but there are challenges and knowledge barriers that must be considered to achieve success.

Competitive Assessment

Aberdeen Group analyzed the aggregated metrics of surveyed companies to determine whether their performance ranked as Best-in-Class, Industry Average, or Laggard. In addition to having common performance levels, each class also shared characteristics in five key categories: (1) **process** (the system engineering process they use); (2) **organization** (defined ownership of processes); (3) **knowledge management** (how design data is managed and exposed); (4) **technology** (the tools that enhance and support system engineering); and (5) **performance management** (metrics used to assess product performance during development). These characteristics (identified in Table 3) serve as a guideline for best practices, and correlate directly with Best-in-Class performance across the key metrics.

Table 3: The Competitive Framework

	Best-in-Class	Average	Laggards
Process	System functions mapped to system requirements		
	63%	53%	38%
	Trade-off studies investigate architecture alternatives		
	54%	42%	40%
	Requirements verified using models (while design is under development)		
	57%	45%	35%
Organization	System engineer provides input into definition of verification tests		
	74%	67%	53%
Knowledge	Customer need linked to requirements		
	75%	59%	49%
	Requirement verification status centrally managed		
	57%	51%	40%
Performance	Design alternatives evaluated on multiple criteria (i.e. performance, cost, and risk)		
	74%	56%	48%
	Product performance criteria defined prior to development work		
	100%	81%	73%
	Performance criteria defined within system requirements		
	86%	75%	50%
Technology	System Engineering technologies currently in use:		
	▪ 52% requirements management ▪ 67% simulation tools ▪ 74% document management ▪ 70% PDM ▪ 40% integrated product data/ requirements management ▪ 41% PLM	▪ 42% requirements management ▪ 58% simulation tools ▪ 72% document management ▪ 55% PDM ▪ 29% integrated product data/ requirements management ▪ 38% PLM ▪	▪ 38% requirements management ▪ 50% simulation tools ▪ 57% document management ▪ 40% PDM ▪ 28% integrated product data/ requirements management ▪ 18% PLM

Source: Aberdeen Group, October 2009

Capabilities and Enablers

Based on the pressures driving system engineering improvements, the challenges of system engineering, the strategies deployed, the findings of the Competitive Framework and interviews with end users, Aberdeen's analysis of the Best-in-Class reveals that there are four key areas companies must focus on to optimize system engineering for more profitable products that are on schedule and on time:

- Capture and manage the needs of customers to optimize revenue potential
- Make sure the goals of all requirements are clear across all engineering disciplines to support collaboration and ensure requirements are met
- Analyze the system architecture to take cost out of products while still meeting customer needs
- Focus of performance validation throughout the development process to avoid finding expensive end of cycle system level problems that put schedules at risk

Processes, organizational responsibility, knowledge management, technology, and performance management all play a role in supporting these four things.

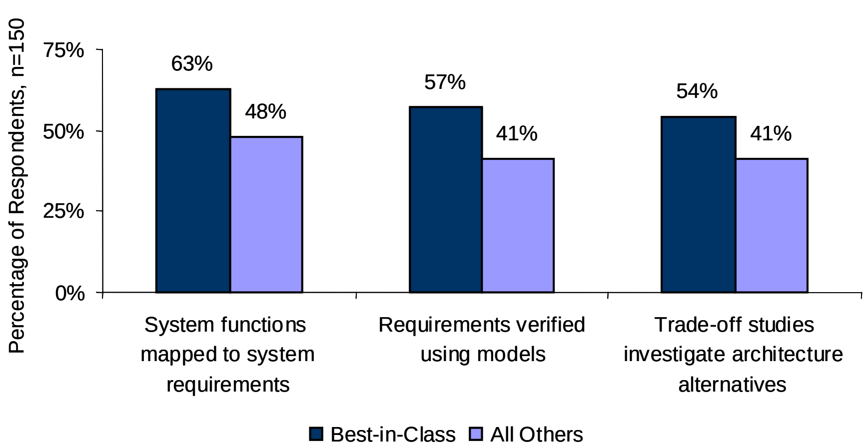
"The introduction of MCAD software has enabled us to significantly reduce our engineering hours for product development by enabling designers to make smarter, more flexible part and product designs. This flexibility and speed allows us to quickly provide designs for new customer applications which also helps to capture new business. We also are able to quickly adapt to changes driven by CAE testing, validation testing, manufacturing and suppliers as well as changing customer requirements.

~ Mark Stirling, Manager, Continental Automotive

Process

The Best-in-Class processes that support system engineering simplify the complexity, catch problems earlier, and improve the ability to make decisions about system architecture (Figure 6).

Figure 6: System Engineering Processes



Source: Aberdeen Group, October 2009

The Best-in-Class are 66% more likely than Laggards to map system functions to system requirements. This clarifies how each requirement fits into the overall function of the system, making it easier for engineers to understand what is expected from their work. Not only does this improve communication across engineering disciplines, but it also help ensure the final system will function as originally intended, a top challenge of system engineering.

To make better decisions about system architecture, the Best-in-Class are 29% more likely than the Industry Average to conduct trade-off studies to investigate architecture alternatives. By investigating alternatives, they can have greater confidence that the final solution is cost effective, a top pressure driving improvements in system engineering, yet have the required performance to meet customer expectations, a top objective for improving system engineering. This is key to reducing product cost while maximizing revenue potential with products that meet the performance expectations of customers. System models help to support this process.

Another process the Best-in-Class are 27% more likely to implement than the Industry Average is to validate the requirements with system models and simulation models throughout the entire development process. This helps them catch system level problems as early as possible during the design process. As a result, they are more like to meet their product launch dates and they avoid last minute changes that drive up product cost. Simulation provides insight into product behavior before a physical prototype exists. While it takes time to set up simulation models, the Best-in-Class have still been able to reduce the length of their development process and are meeting product launch dates better than their competitors, proving that the time invested up front is well worth the time it saves by avoiding significant problems found at the end of development.

"Having a central data base and transparent workflows, provided by tools such as PLM, means less efforts for data/information retrieval. This also ensures that people work on the 'right' version and there are clear states of the project. This leads to easier collaboration and reduced development time which means better results (i.e. products). "

~ Joachim Lentes, Head of Digital Engineering, Fraunhofer IAO

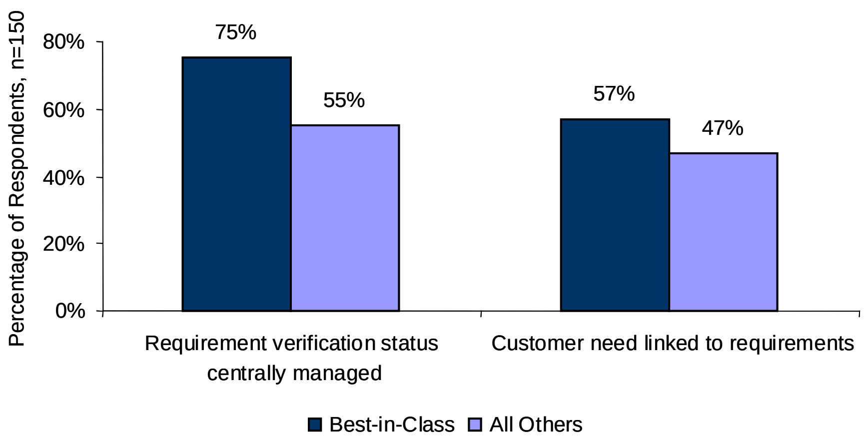
Organization

When dealing with a cross functional team of engineers who are working on an integrated system, it is important that ownership is clearly defined. One of the tasks where responsibilities must be defined is the definition of verification tests. The Best-in-Class are 40% more likely to have a systems engineer provide input into the definition of verification tests. Since the system engineer is defining the system requirements and the system architecture, he or she is also in a good position to define how those requirements should be validated. In addition, by going through the process of defining how a requirement can be validated, it improves the quality of the requirement definition because thought has been given to how the behavior would be measured when it performs correctly. This capability simultaneously addresses the challenges of making sure the requirements are met in the final product and avoiding problems late in the development cycle. With the requirements clearer, and the tests well defined and tied to requirements, it will be much easier to validate performance, verify requirements are met, and identify problems.

Knowledge Management

The information gathered by following Best-in-Class processes needs to be managed and exposed to support team collaboration and improve understanding (Figure 7).

Figure 7: Managing Knowledge for System Engineering



Source: Aberdeen Group, October 2009

"To improve the success of the product, make sure you clearly understand the voice of the customer and meet their requirements."

~ Manager, A&D Company

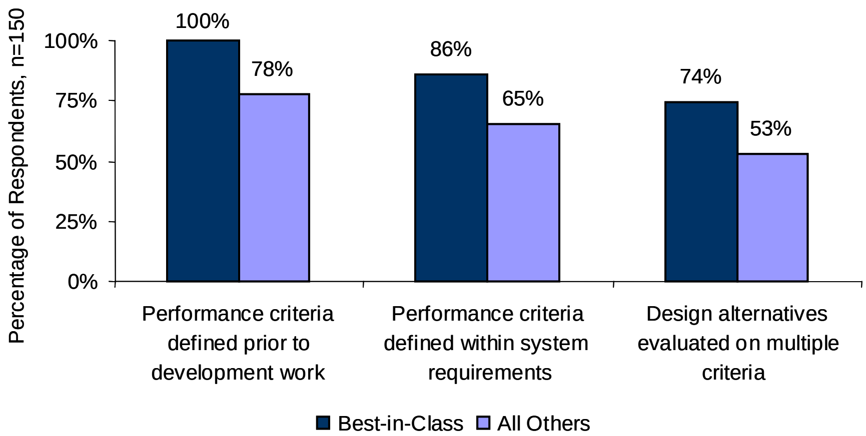
One of the top objectives of improving system engineering processes is to improve the ability to meet customer expectations. This is not possible unless it is clear what customers want and those customer expectations are tied to product requirements. The Best-in-Class are 27% more likely than the Industry Average to link customer needs to requirements. By capturing and managing customer needs and typing that to a product requirement, it is clearer to the engineer what a particular component must do so that the intended requirements are met. It also makes it easier to validate the intended behavior because it is clearer how the customer would be using the product in a real use scenario. There is more demand for products that are aligned to the customer and this higher demand means companies can enjoy higher profit margins on their products.

To support their ability to focus on performance validation, the Best-in-Class are 43% more likely than Laggards to centrally manage which requirements have been verified. This provides more visibility to the status of the project so that other engineers and management know what is working and what isn't. This improves communication and helps the team collaborate better. By understanding what has been validated it is also easier to identify the root cause when a system level integration issue is found.

Performance Management

Defining system performance criteria is extremely important to clarify what is expected out of each requirement. This in turn makes it more likely that the system will perform as intended (Figure 8).

Figure 8: Performance Management for System Engineering



Source: Aberdeen Group, October 2009

"With increased attention on the design specification and applying quality metrics, we have been able to reduce development time uncertainty and increase design reliability."

~ Bob Stout, Scientist/Embedded Systems Architect, Microfirm

By clearly defining the performance criteria, it makes it easier to validate that requirements have been met throughout the entire development cycle. To start things right, 100% of Best-in-Class companies define performance criteria before any development work is started. To make sure this performance criteria is constantly referenced during the development process, the Best-in-Class document it right in the system requirements. Everything starts with the requirements so, by putting in this effort up front with clearly defined requirements, the project has a much greater chance of success.

With performance criteria well defined, it is also easier for the system engineer to make better decisions about the system architecture. While taking cost out of the product is a top pressure, the Best-in-Class are 32% more likely than the Industry Average to look at multiple performance criteria, not just cost, when they define the system architecture. They look at the full picture thus ensuring they will deliver a product that truly meets customer needs in a cost effective way, allowing them to optimize product profit margins.

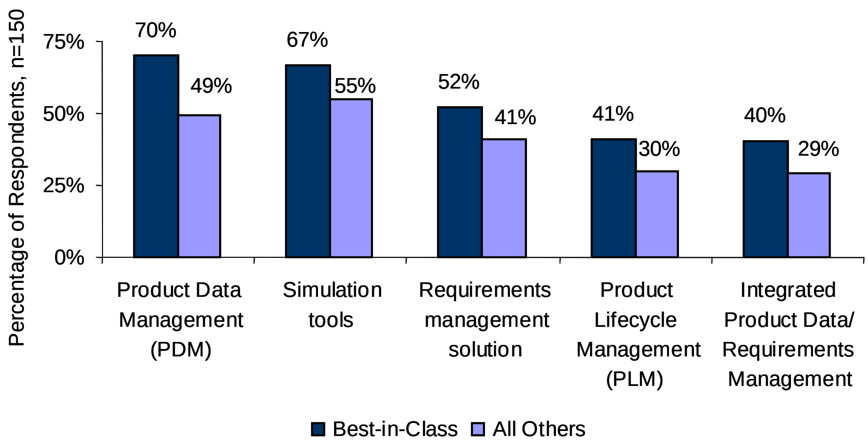
"What has made the biggest difference in our process for developing Mechatronic products is System Engineering and Requirement Management functionalities integrated on a Mechatronics PLM system."

~ Engineering Manger, Instrumentation Controls Manufacturer

Technology

Technology plays a very important role in supporting the Best-in-Class capabilities of system engineering. There are a wide variety of technologies in use to support system engineering. There are a few that are particularly differentiated and more likely to be used by Best-in-Class companies than their competitors, leading to superior performance (Figure 9).

Figure 9: System Engineering Technology



Source: Aberdeen Group, October 2009

"MCAD software coupled with our PLM system permits us to manage product configurations very accurately. Configuration and change management controls implemented in the PDM system enforce best practices for control and release of designs and Product configurations. The PDM system also manages other non-CAD key specification documents for the products and component parts. All product data is related through product structures in the PDM system. Since the system is web based all data is available to users throughout the globe with access rights controlled by user roles."

~ Mark Stirling, Manager, Continental Automotive

One of the most important things for successful system engineering is making sure the requirements are clear and well understood by the entire development team. To support this, the Best-in-Class are 23% more likely than the Industry Average to use a requirements management solution. Requirement management centralizes the requirements so that they are available to the entire development team. When there is a change to the requirements, the requirements management solution ensures that to updated requirement is made available to the entire team. In addition, requirement management solutions have features such as requirements traceability that allow the "life" of a requirement to be tracked. This means that the requirement can be traced from its origin, all the changes it went through, to how it was eventually implemented in the final product. This is especially important when there is a problem with meeting a requirement. In a product as complex as an integrated system, this can be very different to track down. Requirement management solutions also offer another important feature of managing a complex system - visibility to dependencies. This means when there is a change to a requirement, it is easy to know what else has been affected.

The Best-in-Class are more likely to centralize design data. Centralizing design data means everyone has access to the latest version, improving collaboration. Product Data Management (PDM) and Product Lifecycle Management (PLM) are used to support this. The solutions also provide version control so that data is not overwritten and it is clear what data is complete and which is still under development. PLM offers the benefit of PDM but also supports development processes such as change management, project management, configuration management, and collaboration.

The Best-in-Class are 40% more likely to use a solution that integrates requirements management with the storage of the design data. This allows the requirement to be linked to the design or code that met that requirement, improving the ability to confirm a requirement was implemented correctly, in addition to making it easier to manage the relationship between the requirements and design when changes are made, supporting two of the top Best-in-Class strategies for system engineering (Figure 4).

The ability to manage documentation is also important. This includes details such as product specifications and test plans. Centralizing this type of information improves the ability for the team to work together and collaborate because everyone has access to the same information and no one has to waste time trying to track down who has the latest version of a document.

Finally, simulation tools are very important to assess the performance of product behavior before a physical prototype is available. This makes it easier to catch problems as early as possible and also makes it possible to make more well information decisions. Simulation tools are available for each engineering discipline, but there are also some more advanced tools for integrated simulations that represent aspects of the design from multiple engineering disciplines. This provides even better insight into the integrated system behavior and makes it easier for engineers to understand the impact of their design decisions on other engineering disciplines.

"Evaluating different design alternatives using multiple criteria is very helpful. It starts everything right from the beginning. We can generate consensus about the right criteria & weights with a moderated process that advances the design substantially. Additionally the transparency for comparing different alternatives is a big benefit."

~ Joachim Lentes, Head of Digital Engineering, Fraunhofer IAO

Aberdeen Insights — Technology

One of the most important parts of developing systems is managing the requirements. The requirements can be considered the plan for the design. Any project without a good plan has little chance for success. On the other hand, a project with a good plan where it is clear to everyone what they need to accomplish, who will do what, and the interdependencies between the task each person is working on, is far more likely to succeed. This is exactly the effect well managed requirements have on the development team.

Many companies use spreadsheets to manage requirements. This makes sense because it is quick and easy to enter information. However, spreadsheets lack the capabilities that support the ability to execute the Best-in-Class strategies for system engineering. Understanding the impact of changed requirements on the design requires links from the requirements to the design. Without this understanding, the change will not be implemented correctly because system complexity makes it difficult to determine which portions were affected and which engineers need to know about it.

Spreadsheets can easily organize requirements, but are not well suited for managing interdependencies that would provide the required visibility to design relationships. Further, the ability to then drive the change across multiple engineering disciplines is even more complicated, especially without the ability to create role based and function based views of the requirements.

continued

Aberdeen Insights — Technology

With a lot of thought and careful organization a spreadsheet can be set up to increase visibility into which subsystem fulfills each requirement. However, one of the main reasons for doing this is to understand the impact of changes. To make a change to the spreadsheet, it becomes a very manual task of sifting through rows and rows of requirements to make sure each subsystem is updated correctly and even still, it is hard to know if everything that would be impacted was updated.

When dealing with something as complex as system design, solutions that were developed specifically with the challenges of managing requirements in mind should be strongly considered. Requirements management solutions have the capabilities that will support the ability to execute the Best-in-Class strategies for system engineering. These solutions provide traceability across engineering disciplines, enable different role based views of the requirements, and manage dependencies between requirements that are critical for change management. These features will make sure that that well developed plan continues to be valid throughout the life of the project because it will correctly reflect the changes made along the way. As a result, a requirements management solution is a powerful tool for guiding the project to success.

"Project management software has allowed us to better track and monitor progress of the design efforts."

~ Engineering Manager, A&D Company

Chapter Three: Required Actions

Fast Facts

√ Use multiple design criteria to define system architecture and add that criteria to the system requirements

√ Requirements should be linked to higher level system functions as well as to the overall customer need it meets

√ Integrate code and design data with requirements management

Whether a company is trying to move its performance from Laggard to Industry Average, or Industry Average to Best-in-Class, or maintain its Best-in-Class advantage, the following actions will help spur the necessary performance improvements:

Laggard Steps to Success

- **Define performance criteria within system requirements**. Predicting system behavior is a top challenge for system design, but it is virtually impossible to get the behavior right if the performance criterion is not clear. Defining that criteria up front and putting it right in the system requirements makes it clearer to the designer what must be accomplished, thus increasing the chances of success. The Best-in-Class are 72% more likely to do this.
- **Map system functions to system requirements.** One of the reasons making sure the requirements are met in the final product is a top challenge is because the requirements are not understood. By mapping which function the requirement supports, the requirement becomes much clearer to the designer. The Best-in-Class are 66% more likely to do this.
- **Verify requirements with system models throughout the development process**. Identifying system level problems earlier in the development process is a top challenge of system engineering. By using system modeling tools to verify the requirements are met on an ongoing basis throughout the entire development process, many problems can be caught much earlier. The Best-in-Class are 63% more likely to do this.

Industry Average Steps to Success

- **Link customer needs to the requirements.** By aligning products to customer needs, there will be more demand for those products, thus allowing higher profit margins to be realized. The Best-in-Class are 27% more likely to do this.
- **Evaluate multiple performance criteria when defining product architecture.** Cost is a driving pressure for better system engineering, but it is important to also ensure the product meets customer performance expectations as well. By evaluating multiple criteria, more informed decisions can be made about the product architecture, optimizing its profitability. The Best-in-Class are 32% more likely to do this.
- **Integrate code and design data with requirements management.** Many Best-in-Class strategies involve the ability to improve change management. When code and design data are

integrated with requirements, tracing the impact of changes is easier which is needed to implement changes correctly. It is also easier to verify requirements were implemented correctly. The Best-in-Class are 38% more likely to have this level of integration.

Best-in-Class Steps to Success

- **Conduct a gap analysis to ensure all requirements are traceable to a component.** A top challenge is making sure all requirements are met. The Best-in-Class are already doing a variety of things to address this challenge. However, a gap analysis would identify any holes where the requirements will not be fulfilled. This also makes it easier to track down the root cause of why requirements were not met. System modeling is a way to do this. Currently, 21% of the Best-in-Class do this, showing it is on the cutting edge for successful companies.
- **Conduct a post mortem to analyze how many of the initial requirements were changed.** This process will identify areas of improvement for better initial requirements definition to increase the chance of getting them right the first time. Many of the strategies improve change management processes, but this step will help reduce the number of changes in the first place. Currently, 32% of the Best-in-Class are doing this, which is more than competitors, but it will help the Best-in-Class maintain a competitive advantage.
- **Integrate MCAD and ECAD design tools.** One of the top challenges is overcoming the lack of cross functional knowledge. One of the barriers to this is that each engineering discipline has its own design tools. Several solutions are now available that enable the integration of MCAD and ECAD, making it easier for mechanical engineers and electrical engineers to work together. Currently, 30% of the Best-in-Class are using these tools, but considering the impact the lack of collaboration between engineering disciplines has on system design, more should take advantage of this.

"Throughput my career in industrial embedded systems design, failures can cost lives, hard money (many millions of dollars in most cases), and catastrophic environmental damage. Specifications are dictated by industry, legal, and/or international standards. If those aren't met, there simply is no product - period. This imposes a level of discipline in the design process that enforces software engineering rather than simply hacking code. The functional spec is simply a tool to capture the design, which facilitates the design review process. Some of the latest software development and management fads encourage less discipline, which is a mistake."

~ Bob Stout, Scientist/Embedded Systems Architect, Microfirm

Aberdeen Insights — Summary

Good system engineering practices offer a lot of opportunity for greater profitability. However, poor system engineering practices are extremely costly. To be successful with system engineering, companies must focus on the needs to their customers and translate them into design requirements to maximize revenue potential. Requirements must be clear across all engineering disciplines. Understanding what each is responsible for will make collaboration easier and overcome the lack of cross functional knowledge across engineering disciplines. With a thorough trade-off analysis of the system architecture it can be optimized for the most cost effective, highest performing product. Finally, validating system performance throughout the development cycle is critical to catch problems as early as possible during the design process.

Appendix A: Research Methodology

Between July and August 2009, Aberdeen examined the use, the experiences, and the intentions of 170 enterprises in a diverse set of industries. Aberdeen supplemented this online survey effort with interviews with select survey respondents, gathering additional information on their strategies, experiences, and results.

Responding enterprises included the following:

- *Job title:* The research sample included respondents with the following job titles: Executive level manager (11%); VP/Director (20%); Manager (34%); Engineer (27%); and other (8%).
- *Industry:* The research sample included respondents from a wide cross section of industries. The sectors that saw the largest representation in the sample were aerospace and defense (14%), industrial equipment manufacturing (24%); industrial product manufacturing (14%); medical devices (12%), automotive (13%); high tech (16%), and other (7%).
- *Geography:* The majority of respondents (66%) were from North America. Remaining respondents were from Europe (24%), the Asia / Pacific region (8%), and from the rest of the world (2%).
- *Company size:* Twenty-five percent (25%) of respondents were from large enterprises (annual revenues above US $1 billion); 35% were from midsize enterprises (annual revenues between $50 million and $1 billion); and 40% of respondents were from small businesses (annual revenues of $50 million or less).
- *Headcount:* Thirty-seven percent (37%) of respondents were from small enterprises (headcount between 1 and 99 employees); 22% were from midsize enterprises (headcount between 100 and 999 employees); and 41% of respondents were from large businesses (headcount greater than 1,000 employees).

Study Focus

Respondents completed an online survey that included questions designed to determine the following:

√ What is driving companies to improve system engineering

√ The challenges of system engineering

√ The actions these companies are taking to improve system engineering

√ The capabilities and technology enablers they have in place to support system engineering

The study identifies emerging best practices for system engineering and to provide a framework by which readers could assess their own capabilities.

Table 4: The PACE Framework Key

Overview
Aberdeen applies a methodology to benchmark research that evaluates the business pressures, actions, capabilities, and enablers (PACE) that indicate corporate behavior in specific business processes. These terms are defined as follows: **Pressures —** external forces that impact an organization's market position, competitiveness, or business operations (e.g., economic, political and regulatory, technology, changing customer preferences, competitive) **Actions —** the strategic approaches that an organization takes in response to industry pressures (e.g., align the corporate business model to leverage industry opportunities, such as product / service strategy, target markets, financial strategy, go-to-market, and sales strategy) **Capabilities —** the business process competencies required to execute corporate strategy (e.g., skilled people, brand, market positioning, viable products / services, ecosystem partners, financing) **Enablers —** the key functionality of technology solutions required to support the organization's enabling business practices (e.g., development platform, applications, network connectivity, user interface, training and support, partner interfaces, data cleansing, and management)

Source: Aberdeen Group, October 2009

Table 5: The Competitive Framework Key

Overview	
The Aberdeen Competitive Framework defines enterprises as falling into one of the following three levels of practices and performance: **Best-in-Class (20%) —** Practices that are the best currently being employed and are significantly superior to the Industry Average, and result in the top industry performance. **Industry Average (50%) —** Practices that represent the average or norm, and result in average industry performance. **Laggards (30%) —** Practices that are significantly behind the average of the industry, and result in below average performance.	In the following categories: **Process —** What is the scope of process standardization? What is the efficiency and effectiveness of this process? **Organization —** How is your company currently organized to manage and optimize this particular process? **Knowledge —** What visibility do you have into key data and intelligence required to manage this process? **Technology —** What level of automation have you used to support this process? How is this automation integrated and aligned? **Performance —** What do you measure? How frequently? What's your actual performance?

Source: Aberdeen Group, October 2009

Table 6: The Relationship Between PACE and the Competitive Framework

PACE and the Competitive Framework – How They Interact
Aberdeen research indicates that companies that identify the most influential pressures and take the most transformational and effective actions are most likely to achieve superior performance. The level of competitive performance that a company achieves is strongly determined by the PACE choices that they make and how well they execute those decisions.

Source: Aberdeen Group, October 2009

Appendix B: Related Aberdeen Research

Related Aberdeen research that forms a companion or reference to this report includes:

- *Embedded Systems Development: Three Proven Practices for Speed and Agility*; March 2009
- *Engineering Evolved: Getting Mechatronics Performance Right the First Time,* November 2008
- *Engineering Executive Agenda*; June 2008
- *System Design: New Product Development for Mechatronics;* January 2008

Information on these and any other Aberdeen publications can be found at www.aberdeen.com.

Author: Michelle Boucher, Research Analyst, Product Innovation & Engineering Practice, (michelle.boucher@aberdeen.com)

RAL14020-USEN-00