WHITEPAPER

Applications of Systems Engineering in Healthcare

Introduction

When it comes to healthcare, time to market is one of the most crucial aspects of success or failure. However, medical product development teams face several challenges that slow product development and in the quest to speed up the process, some teams are turning to systems engineering to improve the process.

In this whitepaper, we'll look at the challenges healthcare development teams face, the difference between market-driven and contract-driven industries, and how the power of simplicity can help healthcare systems engineering teams strike a perfect balance to adapt, innovate, and succeed.

The Challenges of Healthcare Systems Development

To understand how systems engineering can help, it's important to first look at the challenges development teams face.

First, teams must balance time demands with the need to launch products that are both safe and effective. **Today, the time to define requirements has increased by 29%, and unplanned requirements churn has increased 81%, resulting in about 70% of medical products being delivered late**.

The shifting regulatory landscape presents more challenges, including the increased cost of adherence to such regulations as **Software as a Medical Device** (SaMD), **Software in a Medical Device** (SiMD), **Medical Device Regulation** (MDR), and **In Vitro Diagnostic Regulation** (IVDR). At one of the top medical device development firms, for example, their product developers had to monitor approximately 8,000 regulations. Ensuring that products meet quality, safety, and performance standards represents significant financial impact; getting it wrong can cost billions of dollars. Across the industry, non-routine quality events cost between $2.5 and $5 billion per year.

In addition to increasing design complexity, there is an increase in process complexity. Software development teams have gone from between 20 and 40 people to hundreds of people. **Artificial intelligence** (AI), **machine learning** (ML), and other new technologies represent complexity inside devices. Organizations are getting more complex as well, with a heavy focus on acquisition, which means constantly integrating new teams and cultures, sometimes dispersed across the globe.

Systems engineering can help product developers in healthcare manage these complexities and streamline development to keep them competitive in a rapidly changing market.

Market-Driven vs. Contract-Driven

To understand how systems engineering can improve speed to market, it's important to first understand the difference between a "market-driven" and a "contract-driven" industry.

In a market-driven industry, the first mover tends to get the lion's share of the profits. Market-driven industries have many customers, and the stakeholders are internal to the business. Budget, time, and requirements are negotiated within the organization.

In a contract-driven industry, success means satisfying the contract. Budget and time are fixed by the contract with one (or very few) customers. In this scenario, requirements are a key commitment negotiated within formal design control.

The two different industry models present very differentt equirements challenges. In a market-driven industry, requirements are an internal business tool that helps communicate across business functions. They must be validated, but the development team decides on timing and features. If a team member develops a new, innovative feature, everyone can agree to take extra time to develop it. In a contract-driven industry, that likely wouldn't be possible given the constraints of the contract.

Systems engineering can help the market-driven industry turn ambiguous needs into clear and feasible solutions to be implemented by hardware and software teams.

Systems Engineering: From Needs to Solutions

Product developers in a market-driven industry receive a lot of input from the various stakeholders within the organization. Their task is to turn that input into marketable products that work seamlessly on day one, day fifty, and years later. The key value produced is seamless integration of those products into every customer's workflow and work systems. Every installation, every service event must produce a uniform, high-quality, high-performing product.

Within those constraints, developers need to optimize the business value. When there are multiple options, marketing will inform the team of the customer value of these options. The implementation teams will pass on the delivery and product cost of those functions. The role of systems engineering is to make trade-offs between those and optimize the business impact based on the cost of implementing. Associated with that is managing technical risks and scaling cost by risk.

The key value of systems engineering is making sure design decisions are identified and closed predictably with one voice across the team. Decisions are framed, the options are agreed to, the decision criteria are agreed to, and the final decision is closed, and stays closed even as stakeholders change. Once the team has a frozen design, integration or quality problems can be found and resolved prior to moving to the next phase. By creating time to react, teams allow themselves space to adjust design early in the program rather than rushing to fix quality issues before shipping.

Winning products happen when systems thinkers are effective.

When everyone across the program engages in systems thinking, the team will maximize the creativity of the entire program.

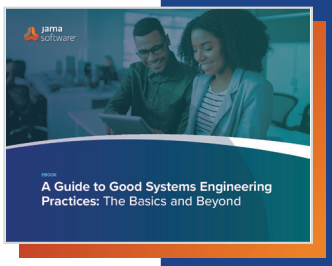
What is Systems Engineering in Healthcare?

As a process example, at one leading US-based medical device development company, engineering teams start with the end customer's performance requirements, such as delivering excellent image quality in their imaging products or proper humidity and temperature for neonatal products. As part of delivering that essential performance, teams must ensure safety and regulatory compliance.

Their product teams also put a high emphasis on usability — ensuring that their products are easy to use and delight the customer. The teams define the right implementation requirements and reliability strategy, and they ensure that their products can be installed and serviced properly.

While there is tremendous diversity in products and programs across most medical device and life sciences companies, there are several commonalities across the product teams as well. Teams have common program milestones and a common systems' lifecycle based on the V-model with iteration and Agile built in.

What differs in product teams are the levels of safety hazards and FDA risk. Teams develop everything from anesthesia technology, which could easily kill a patient, to ultrasound, which is non-ionizing equipment operated with light, handheld probes. To accommodate these different levels of risk, teams adjust the process rigor so that higher risk modalities have higher process rigor.



WHITEPAPER

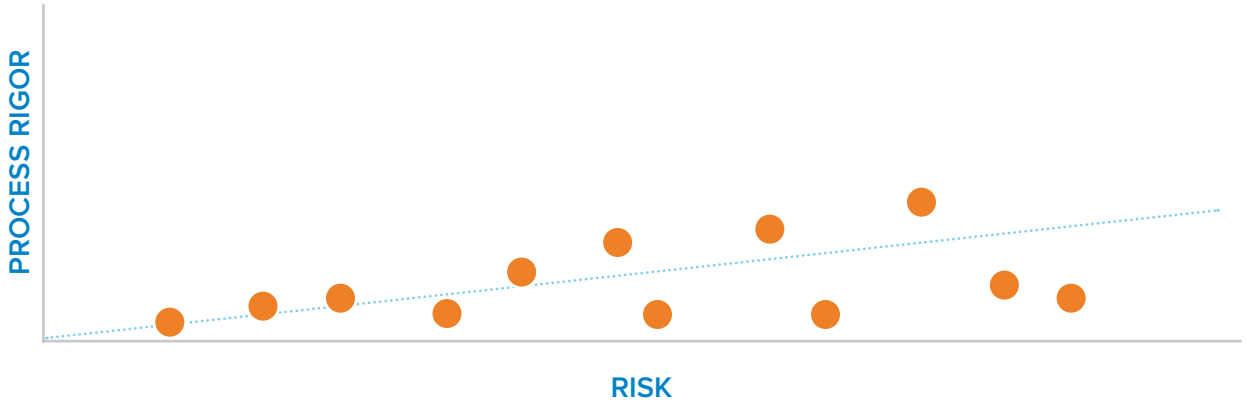
To learn more about ISO 14971, download The Complete Guide to the Systems Engineering Body of Knowledge (SEBoK).

In this whitepaper, we cover the main clauses of ISO 14971 — the FDA's mandatory standard for risk assessment in medical devices — and share how Jama Connect® gives you a comprehensive way to manage risk and requirements throughout development.

Download it here »

Additionally, systems engineering teams can look very different across the world. Many organizations operate in different locations with different cultures and different organization sizes. Systems engineering teams can vary from fewer than ten engineers to over one hundred engineers. The scale of the programs can be just a few engineers over a few months to many hundreds of engineers applied to a program that might last three years and is based on technology developed over the prior decade. (Even in that research phase, teams should apply some systems engineering thinking.) Organizations can be product-centralized or decentralized within an organization.

DIFFERING RISK PROFILE

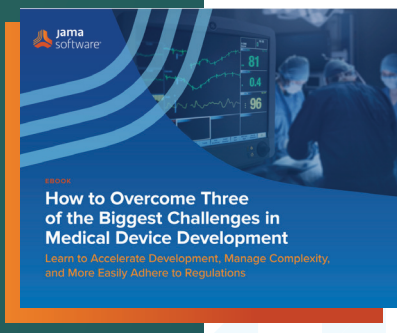


EBOOK

To combat the mounting complexities of innovative, connected technologies, teams must get a stronger handle on their process and avoid gaps in development.

Download this eBook to learn about the three main challenges modern medical device companies face and how to adjust your processes to rise and meet them:

How to Overcome Three of the Biggest Challenges in Medical Device Development »



How to Customize Systems Engineering in Healthcare

Given the various factors that can impact product development and systems engineering in healthcare, here are five attributes to consider to help customize systems engineering for the best product and program outcomes.

1. Technical Risk

Teams can measure technical risk by conducting a thorough hazard analysis. For development projects that present a higher level of risk, teams can customize by providing a higher level of functional excellence, more process documentation, more process compliance, higher rigor of the technical design reviews, and possibly more independent reviewers.

2. Team Experience

A study from Carnegie Mellon showed that the value of systems engineering increases with program complexity, but it decreases with a more experienced team. In other words, a small team with a great deal of experience with the technology and the application can get by with less process rigor. While systems engineering excellence delivers some value, it delivers less value in this scenario. For example, in the ultrasound division of a major medical device and life sciences company, a small team with a dedicated function that had worked together for some time and knew the application could have less rigor and a less formal sign-off..

3. Globally Distributed Team

Globally distributed teams present logistical challenges for systems engineering. To address these challenges, first determine the number of sites and the maximum time differential. If a team has sites in the US, Europe, Middle East, and Asia, the maximum time differential presents real challenges in getting everyone a single phone call or videoconference. The more challenging the communication, the more rigor and detail will be necessary for the program communication plan. The level of attention to communication is critical, and it's vital to have more review of that plan to make sure everybody truly understands it.

4. Team Size

Very large programs may have well over a hundred people organized into twelve to fifteen teams of eight to ten people each. Devote extra attention to the level and rigor of communication and level of review.

5. Product and Technology Maturity

Finally, consider the product and technology maturity. Is it a program to develop new technology or is it cost out? A cost-out program can be challenging, but the team is comfortable with the user base and the application. That kind of program is probably a lower technical risk, and the team probably understands the challenges better. For this kind of program, it's possible to be a little bit lighter in the level of process and documentation rigor. If it's possible to reuse much of the documentation, then senior engineers can move away from spending time doing tedious repetitive work and put more time on hi-quality, hi-level tasks such as acting as reviewers.

Cost out refers to the practice of eliminating cost in the supply chain through the review of actual costs and the adoption of new processes or behaviors to eliminate cost.

ATTRIBUTE	COMPLEXITY MEASURE	EXAMPLE CUSTOMIZATION
Technical Risk	▪ Hazard Analysis	▪ Level of functional excellence rigor ▪ Rigor of technical design reviews
Team Experience	▪ Subjective... ▪ Local senior engineers	▪ Local level of SE functional rigor + independent technical reviews ▪ Level of signoff
Globally Distributed Team	▪ Number of sites ▪ Max time defferential	▪ Rigor and detail in the program communication plan; level of review
Team Size	▪ Number of engineers	▪ Rigor and detail in the program communication plan; level of review
Product and Technology Maturity	▪ New technology vs. cost out	▪ Level of ease of use/'quality' required ▪ Documentation rigor ▪ Senior engineer allocation

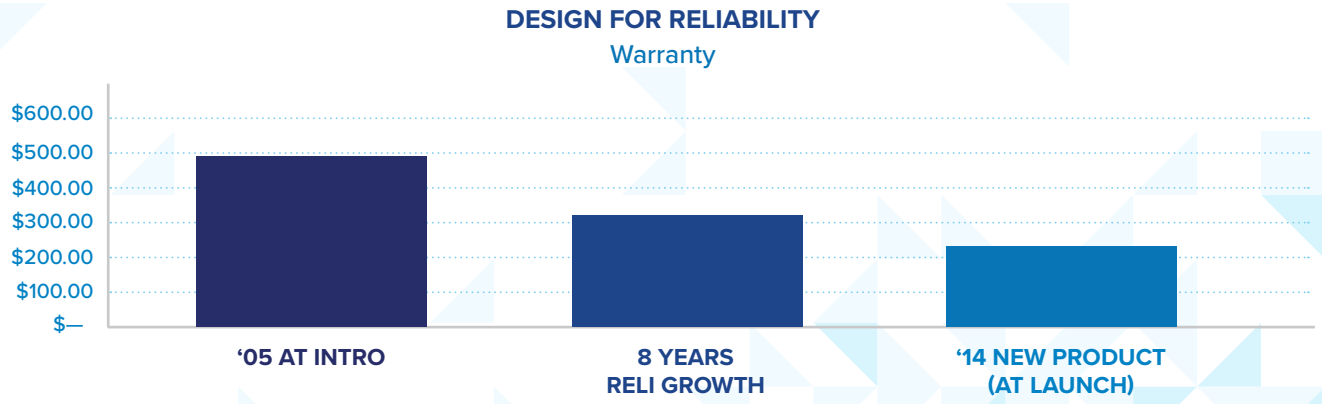
With these foundational ideas established, it's time to get back to the basics — in other words, doing simple things very, very well. Getting back to basics in systems engineering will help set up product teams for success and ensure better rigor and better speed to market. Here are five basics to focus on:

1. Start with the Customer

Start by making sure that the products you deliver integrate into the customers' workflows and work systems, and make sure the software they have works reliably. First, focus on usability "Work Instruction" for compliance with the FDA regulation and 63266. Undergo formative and summative testing; do exploratory testing up front to learn the innovation that new customers might bring to products. Consider things like expected user abuses, such as transport. Implement central resources for coaching people, sharing best practices, and doing independent reviews. Finally, incorporate creative new ideas that will delight the customer.

Next, focus on design for reliability. For example, at a leading global healthcare company, the team uses a formal six-step process and reliability practitioner certification. One product released in 2005 came with a warranty that paid for any service event in the first year. This support costs about $500. Over eight years of constant improvements, the service cost dropped to about $170, about a threefold reduction. Nine years after introducing that platform, they launched a new platform, and the service cost dropped even lower.

With reliability built into the new product development process, all new products deliver high quality at initial release.



2. Scope Management

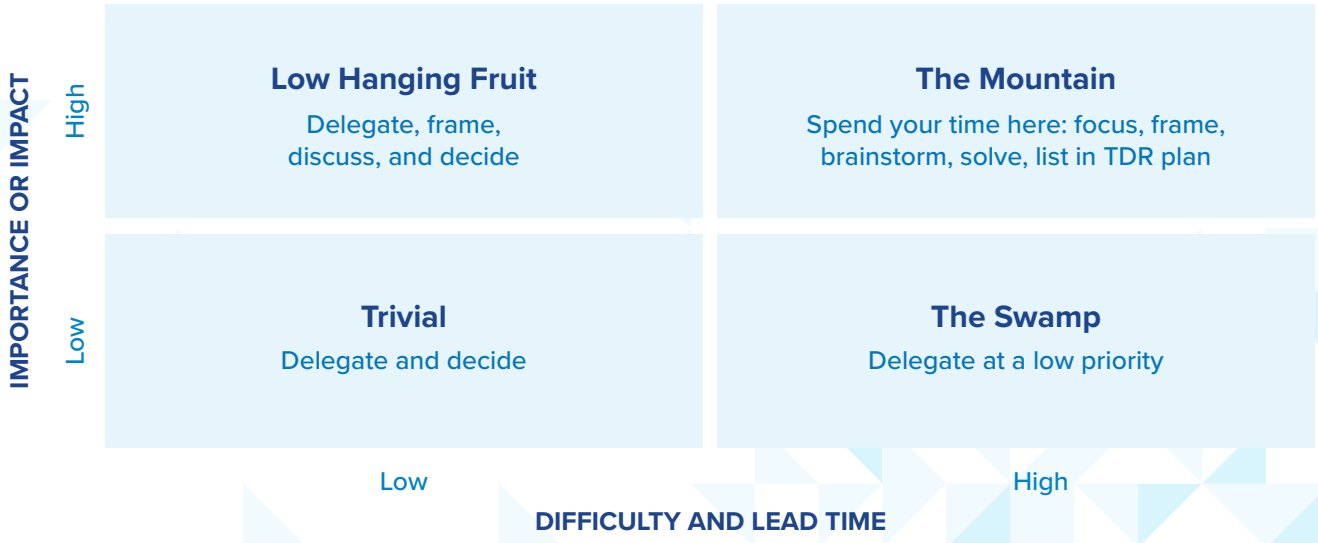
Make sure the amount of work the team undertakes is feasible. The number of resources can be a stretch, but there shouldn't be two or three times too much work for the team to complete with the resources allotted. That will create churn and inefficiencies.

A simple tool used by this leading healthcare company was a spreadsheet that listed critical requirements, stretch requirements, and next generation requirements. The team could prioritize features and assign resources in a way that would give the best chance of completion by the delivery dates. With one mammography project at that healthcare company, the team realized it had about twice as much content as was feasible with the time and resources available. It could simply cut out all the stretch requirements and focus on critical requirements — in this case, patient improvements. The result was that the new product had a two- to three-point price improvement and a three- to five-point share improvement. The spreadsheet helped produce useful conversations about how to get rid of content and still maintain high business impact.

3. Decision Management

Some key values from systems engineering are managing the decision and the trade-offs, understanding the scope, and understanding the options and the decision criteria used.

The same leading medical device company used a decision tool based on a simple four-block grid of importance vs. difficulty.



For items that had high importance but low difficulty, the team simply delegated to the best engineer in that area. For low importance, low difficulty items, the team assigned tasks members who could accomplish them.

One danger for teams is spending too much time in "The Swamp" — items with high difficulty, lead time, or interdependence, but low importance or impact. These items should be delegated or assigned, but with low priority. Teams should spend most of their time in "The Mountain" quadrant. These are the items that will have the most impact on the product. Prioritize, focus, frame, brainstorm, and solve for these items to have the best impact on product development.

4. Technical Risk Management

Make sure the technical risks are handled at the right stage and the right level. Several risks can be delegated to the implementation teams, but if there is a risk that is a showstopper, that risk should be managed at the systems level, even if the subsystem team is the one doing the risk retirement.

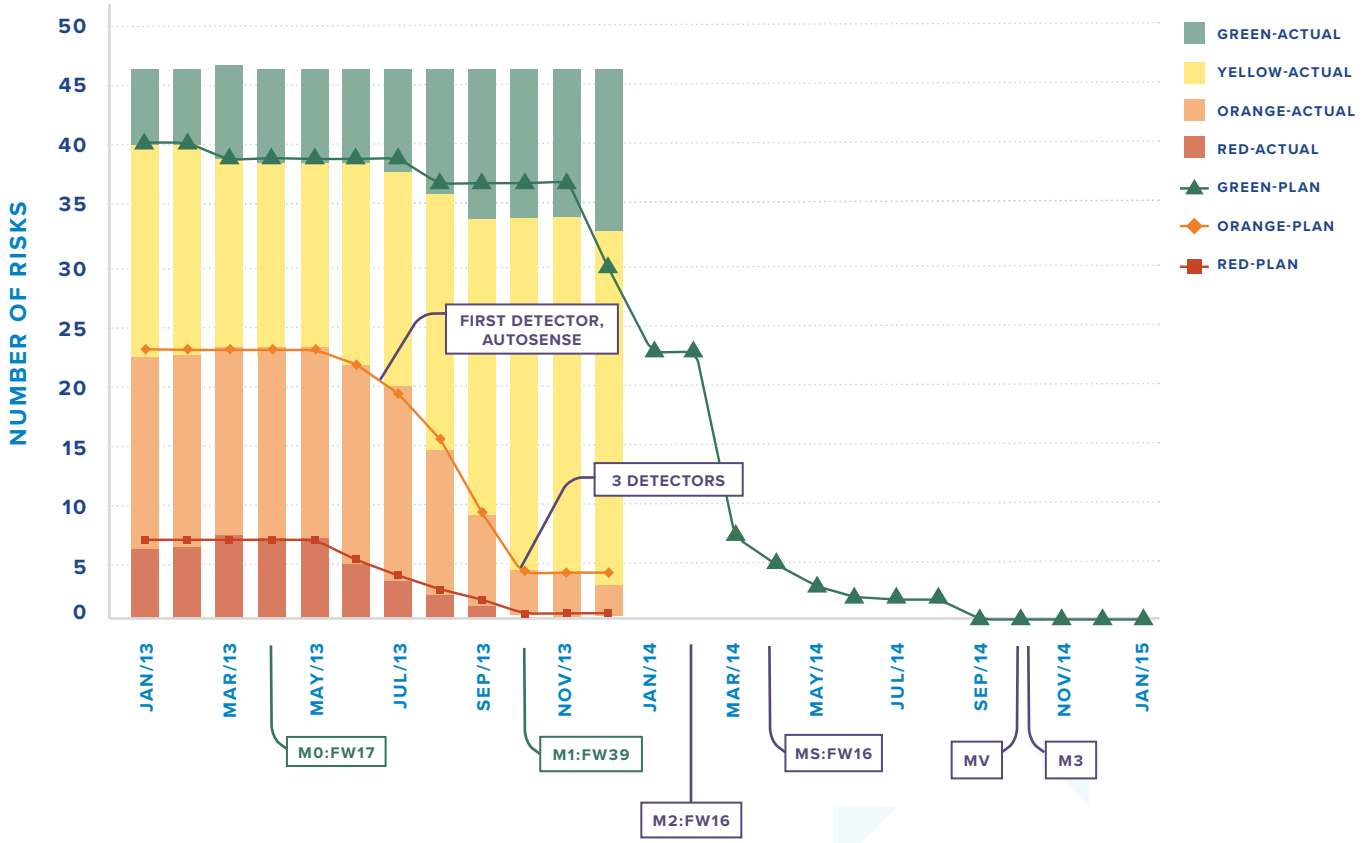
A useful technique is to assess risk classes. These classes are similar to an risk priority number RPN score, but they assign colors to each level to help the team see which items put the program at risk. Red items are program killers. Orange items put the program at risk. Yellow may impact the business case, and green items have little to no effect.

PROBABILITY IMPACT	5 HIGH	4 SIGNIFICANT	3 MODERATE	2 MINOR	1 LOW
5 HIGH	25	20	15	10	5
4 SIGNIFICANT	20	16	12	8	4
3 MODERATE	15	12	9	6	3
2 MINOR	10	8	6	4	2
1 LOW	5	4	3	2	1

The team can then track the risk waterfall — how many of each class exist? The goal is to eliminate most red risks and all but a handful of orange risks before committing to a delivery date.

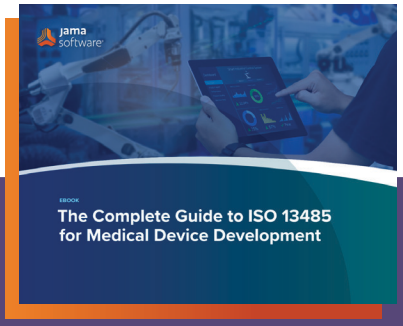
ANNOTATED RISK WATERFALL

Risk Retirement • Actual vs. Plan



5. Active Integration

Active integration helps identify the integration and quality risks up front. The team at the leading medical device company aimed to finish each integration step with 95% confidence — very confident that there would be very little rework and very little quality risk. Focusing on quality in the first part of the program may mean falling behind by the middle of the program, but much of the schedule can be recovered in the second half of the program because most quality issues have already been resolved. By the time the team gets halfway through, they have something they are very confident in.



EBOOK

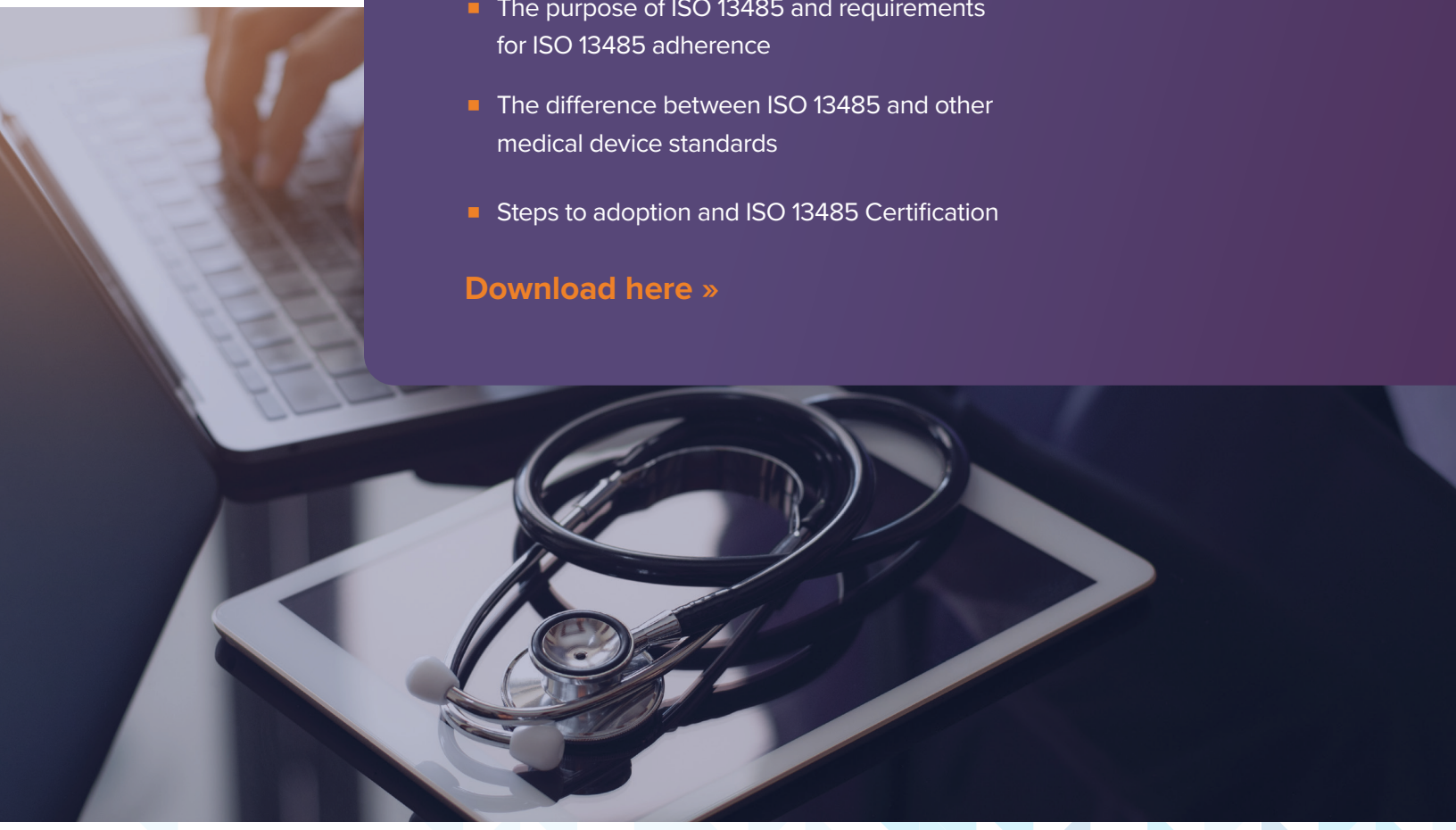
Adopting ISO 13485 can help standardize and systematize the medical device development process.

Though it may look daunting at first, once adopted, ISO 13485 can streamline processes and position organizations for a better outcome with regulatory requirements. This guide will untangle everything your medical device development team needs to know about ISO 13485.

Download this eBook to learn more about:

- The purpose of ISO 13485 and requirements for ISO 13485 adherence
- The difference between ISO 13485 and other medical device standards
- Steps to adoption and ISO 13485 Certification

Download here »



Jama Connect is the Only Solution to Enable Live Traceability™

Executing these basic functions of the V-model at world-class levels is not easy, but it does generate high returns. In a market-driven business, it means focusing on competitive value creation and customer use cases and saving requirements for the next step down.

Systems engineering is a key role and function in medical organization product development and processes. Whether architecting the latest innovation, managing complexity across teams, or making trade-off decisions, modern tools can support these innovative processes by streamlining and digitizing work activities.

Jama Connect® for Medical Device and Life Sciences Development is a best-of-breed platform specifically catered to the medical device and life sciences industry. With Live Traceability™, this solution helps manage the three main pillars of product development: requirements, verification and validation, and risk management. The out-of-the-box medical device configuration aligns with key governing regulations like ISO 13445, 14971, and 62304, but also with some recent updates which also included additional frameworks to align with more life sciences and research organizations.

Many organizations adopt different models based on their organizational processes, markets, regulations, or preferences. Whether it's the FDA waterfall systems V, the Agile method, or a hybrid, Jama Connect can support flexible configurations to accommodate your organization's specific process. Procedures and configuration guides are provided to help organizations implement solutions quickly and

effectively. With additional features such as a collaborative Review Center, reuse and sync for platform development, and Live Traceability to see changes throughout the development lifecycle in real time, Jama Connect can help with your system engineering activities.

Jama Connect also offers out-of-the-box traceability of key elements from intended use, user needs to system requirements and verification or from hazards to risk evaluations, which is compliant to 14971 and their subsequent risk controls. Traceability is a native artifact from just using Jama Connect, and the platform integrates with other best-of-breed tools such as Jira®, Microsoft® Excel®, and others.

Jama Software® updated its medical solution dataset in mid-2023 to line up with industry best practices. Some key highlights include the new Lookup Matrix and a new Industry Standards Traceability use case where you can actually trace industry standards such as ISO standard 60601 to requirements. In addition, the updated dataset includes new reports and templates to improve design traceability and risk management.

Finally, the updated dataset includes solutions such as Software as Medical Device (SaMD), a DHF or design history file project structure for a document-based approach within Jama, and a Research Use Only solution for the less-regulated development processes.

Jama Software is focused on continually updating the Jama Connect platform to meet the rapidly changing healthcare solution industry.

Ready to uplevel your development process?

To learn more about Jama Connect or to arrange a demo, please contact us.

ABOUT THE AUTHOR

Chris Unger is the retired Chief Systems Engineer for GE Healthcare. He was responsible for the definition and improvement of systems engineering process globally. He is a member of the Chicagoland chapter of INCOSE, the co-leader of the INCOSE Healthcare WG, and the co-founder of the “Systems Engineering in Healthcare” conference. Chris has worked as a systems engineer in the defense and medical fields for 40 years. He is also a certified Master Black Belt with fifteen issued patents and two patents applied for. He is an INCOSE certified Experienced Systems Engineering Professional. He graduated with a B.S. in Mathematics and B.S. in Philosophy from M.I.T. and a Ph.D. in Physics from Boston University. Chris is the principal at PracticalSE, LLC a systems engineering consulting company.

Jama software®

Jama Software® is focused on maximizing innovation success in multidisciplinary engineering organizations. Numerous firsts for humanity in fields such as fuel cells, electrification, space, software-defined vehicles, surgical robotics, and more all rely on Jama Connect® requirements management software to minimize the risk of defects, rework, cost overruns, and recalls. Using Jama Connect, engineering organizations can now intelligently manage the development process by leveraging Live Traceability™ across best-of-breed tools to measurably improve outcomes. Our rapidly growing customer base spans the automotive, medical device, life sciences, semiconductor, aerospace & defense, industrial manufacturing, consumer electronics, financial services, and insurance industries. To learn more, please visit us at jamasoftware.com.