Leveraging PTC's Integrity Platform for IEC 62304 Compliance

SPK and Associates, LLC

Our medical device customers report that up to 70% of new product innovation is being created in the software domain. This represents an increase in both volume and complexity.

In addition, the FDA has intensified the review process of medical device companies with a strong focus on software quality in order to comply with regulations. Our medical device customers routinely follow the EN/IEC 62304 standard for software design of medical products as their compliance playbook. Conformance with this standard provides evidence of having a software development process in place, and fulfills the requirements of 21CFR820 as well as the Medical Device Directive 93/42/EEC.

The challenge presented is how best to set up your development environment to achieve compliance with 62304 while also maintaining engineering efficiency. This article explores how we've leverage PTC's Integrity platform to accomplish this.

A Quick Review of 62304

IEC 62304 starts with building the software development plan. The first part of that is establishing the safety class of the software being created for the device.

"The MANUFACTURER shall assign to each SOFTWARE SYSTEM a software safety class (A, B, or C) according to the possible effects on the patient, operator, or other people resulting from a HAZARD to which the SOFTWARE SYSTEM can contribute."

The three classes are further defined as:

- Class A: No injury or damage to health is possible
- Class B: Nonserious injury is possible
- Class C: Death or serious injury is possible

And the standard describes "serious injury" as:

Injury or illness that directly or indirectly

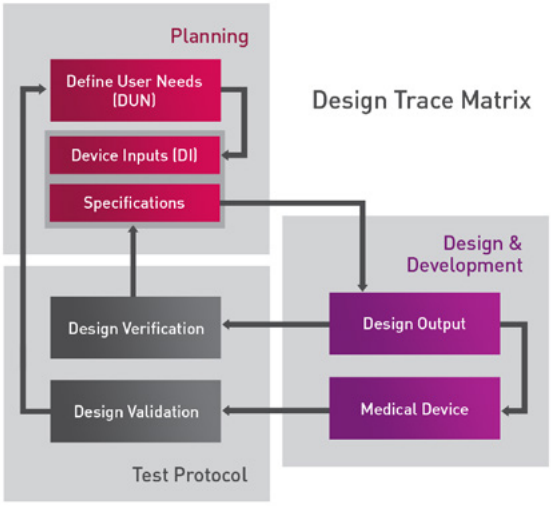
- is life threatening,
- results in permanent impairment of a body function or permanent damage to a body structure, or
- necessitates medical or surgical intervention to prevent permanent impairment of a body function or permanent damage to a body structure

The classification drives what activities are required (Table from Standard):

Software Documentation	Class A	Class B	Class C
Software Development Plan	Must contain contents to sections 5.1 IEC 62304:2006. The plan's content list increases as the class increases, but a plan is required for all classes.		
Software Requirements Specification	Software requirements specification conforming to 5.2 IEC 62304:2006. The content list for the software requirements specification increases as the class increases, but a document is required for all classes.		
Software Architecture	Not required	Software architecture to 5.3 IEC 62304:2006. Refined to software unit level for Class C.	
Software Detailed Design	Not required		Document detailed design for software units. (5.4)
Software Unit Implementation	All units are implemented, documented and source controlled (5.5.1).		
Software Unit Verification	Not required	Define process, tests and acceptance criteria (5.5.2, 5.5.3). Carry out verification (5.5.5)	Define additional tests and acceptance criteria (5.5.2, 5.5.3, 5.5.4). Carry out verification (5.5.5).
Software Integration and Integration Testing	Not required	Integration testing to 5.6 IEC 62304:2006.	
Software System Testing	Not required	System testing to 5.7 IEC 62304:2006.	
Software Release	Document the version of the software product that is being released (5.8.4).	List of remaining software anomalies, annotated with an explanation of the impact on safety or effectiveness, including operator usage and human factors.	

The development plan also needs to address how you're going to manage the potential issues for your device along with prescribed mitigations (Risk Management). And you will need to include a defect tracking system, a configuration management system, and a defined problem resolution process.

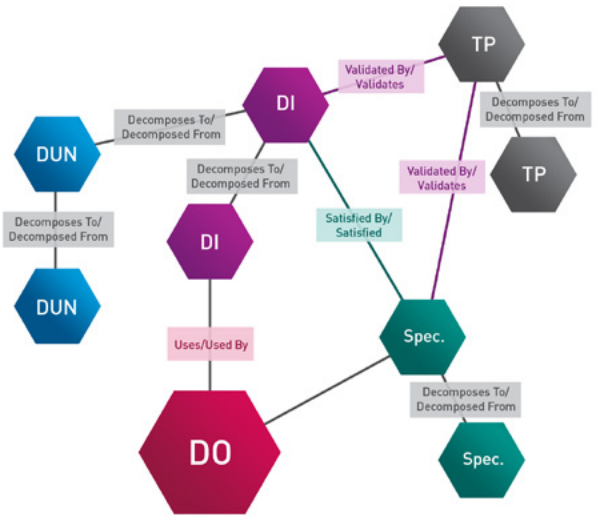
Traceability between requirements, implementation and testing must be provided as well as traces to risk control measures and mitigations. For more information on 62304, purchase it from the IEC webstore.



PTC Integrity Platform

Many of our medical device customers start off with a home grown collection of point tools which may include an SCM system, a defect tracking system, a requirements management system, a testing environment and some PM/reporting tool. We have found that the overhead of maintaining these separate tools, as well as the often poor integration between them, creates an inefficient development environment.

PTC's Integrity platform takes a systematic process approach. It is a solution which shares a common data model and allows you to manage artifacts and change across the entire product cycle for both software and hardware. Integrity's requirements management system, testing environment, defect tracking and SCM system share a common database and UI, which leads to a seamless and efficient workflow between development and test activities.



PTC Integrity – Establishing Requirements MGT and Reports

We begin by leveraging Integrity's configurable document domain architecture to align it with the medical device customer's existing requirement trace relationship. As an example, one of our customers had a model, which can be seen in the right-hand column.

- The planning phase focused on the creation of Defined User Needs (DUNs). These are the business level requirements.
- These DUN requirements then traced to Device/Design Inputs or (DI's) and Specifications (such as an SRS or HRS).
- The DUNs were validated (to make sure the right product was being built) via a Test Case in a Test Protocol.

- And the DI's and Specifications were verified (to make sure we're building the product right), also a Test Case in a Test Protocol.
- Both Plan and V&V items were also traced to Design Outputs. Design Outputs (DO) include all your artifacts associated with building your device. They contain things like plans, drawings, flow charts, part numbers, etc. All design outputs are tracked as part of your design history file (DHF). Our example customer included some of these items as traceable items in a DO document.

The configured PTC Integrity document domain structure then looked like the diagram to the right.

There was a PTC Integrity document domain established for each area (DUN, DI, Specification, DO and Test Protocol). And, specific trace relationships were created between the document domains. I.E.

- A DUN "Decomposes To" a DI
- A DI is "Decomposed From" a DUN
- A DI is "Satisfied By" a Specification
- A Specification "Satisfies" a DI
- A DI is "Validated By or Verified By" a Test Protocol
- A Test Protocol "Validates or Verifies" A DI
- A DI "Uses" a DO
- A DO is "Used By" a DI

Generate Our Reports:

These were the two reports that we needed to create. Medical companies often spend considerable resources on fine tuning their reports. It usually involves several folks including Engineering, QA, and RA (Regulatory Affairs Officers). I've seen other ALM solutions fall flat at this point because they underestimate the importance of this. You have to produce the reports the way the Med Device Company wants it. PTC's Integrity shines here in the flexibility of its report generation capabilities.

The first report was known internally as a DIOVV (Design Input Output V&V). It connected the DUNs with DIs.

DUN #	Intended Use / User Need	DI #	Design Input	Acceptance Criteria	Acceptance Criteria Justification	Validation Protocol(s)	Validation Report(s)
1 Device Specific User Needs [DSUN]							
1.1	User Needs resulting from the Voice of the Customer (Pre-design controls market investigation)	1.1.A					
		1.1.B					
1.2	Consider entries that address critical performance, strength, and/or dimensional criteria that must be met for surgeon acceptance and acceptable surgical outcome.	1.2.A					
1.3	Consider entries that address critical performance criteria that must be met for acceptable surgical outcome and clinical effectiveness.	1.3.A					

The second report was called the DIOTV (Design Input OutPut Trace Verification). It connected the Design Outputs and Test Protocols.

DI #	Design Input	Acceptance Criteria	DO #	Design Outputs & Verification Reference							
				EO (Yes/No)	Design (Part Number/ Feature)	Title / Description	Design Verification Protocol(s)	Design Verification Report(s)	Process (description and / or doc #s)	Process Verification / Validation Protocol(s)	Process Verification / Validation Report(s)
1.1.A			1.1.A.1								
			1.1.A.2								
			1.1.B.1								
			1.1.B.2								

Both reports needed to be output as Excel documents.

Utilizing Integrity's Report Engine, we were able to select a document (or documents) in the DUN domain, pick which trace relationship we wish to traverse (in the example, "Decompose To") and which fields we wished to present (User Need, Acceptance Criteria, etc) and generate the desired DIOVV Excel output document. And the same process was followed, starting from a DI document, to generate the DIOTV report.

PTC Integrity – Establishing Test Environment

PTC Integrity has an integrated test harness for manual testing and an interface capability for automated test tools such as HP Quality center.

The test architecture has the following components:

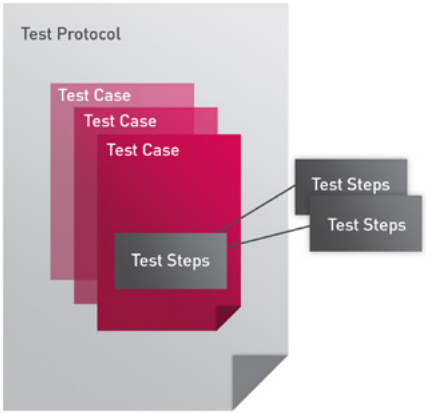
- **Test Plan**: This is your overall test task and metric container. It describes the scope, approach, resources and schedule of the intended test activities.
- **Test Objective**: The Test Plan testing effort is broken down into smaller chunks, perhaps by software functionality or by team/manager, or by specific goals (performance, regression, and unit), etc.
- **Test Protocol**: The test protocol is a document containing test cases.
- **Test Step**: Your test cases can be broken down into one or more specific test step operations performed as part of executing a test.
- **Test Session**: Your test session is a run of a set one or more test case(s)/steps.

Test verdict metrics are captured during a test session. These roll up to the test objective and test plan to provide overall test pass/fail data.

PTC Integrity also easily allows you to see which requirements have no test cases, or which have test cases which have not passed, etc. All of this can be presented in a dashboard with charts and graphs.

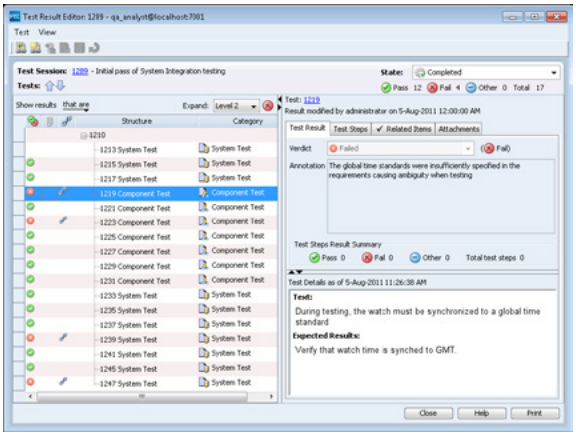
In our customer example, we migrated their existing test environment into PTC Integrity (several thousand test cases and test steps).

Looking into a test protocol document, we now had test cases (which could be traced back to requirements) with defined expected results at each test step.



Generation/Management of Defects

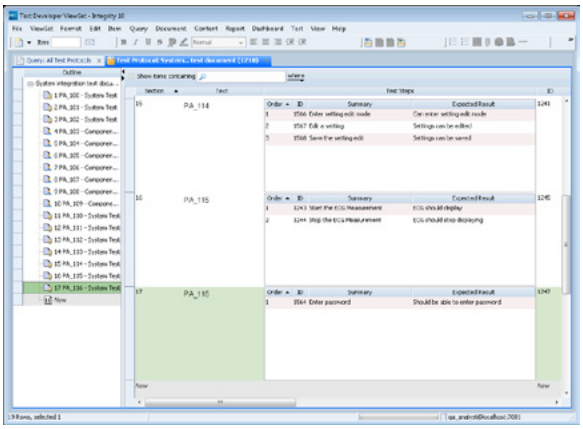
We were able to satisfy 63204's defect system requirement by leveraging Integrity's built-in defect tracking system. It was easy to configure PTC Integrity's workflow and fields to match the customer's existing external defect system and then migrate the data into Integrity. We were then able to initiate defects from within test sessions along with screen capture attachments, and then associate them back to requirements or SCM artifacts.



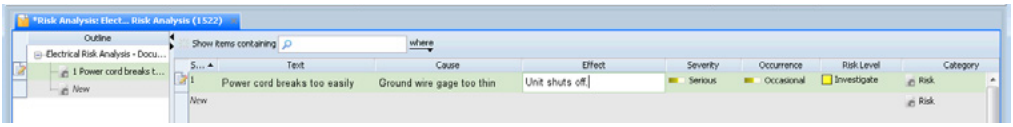
Risk/Hazard Management

PTC Integrity provides an out-of-the box solution for Risk and Hazard management.

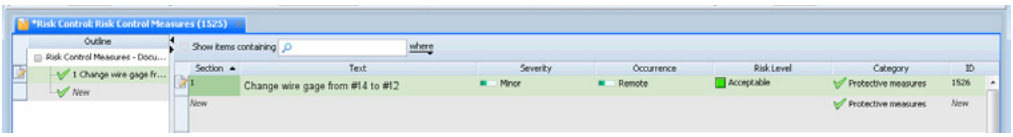
In our customer example we focused on the Risk identification and mitigation. PTC Integrity has a "Risk Analysis" domain. We were able to create risks, with its associated severity and occurrences.



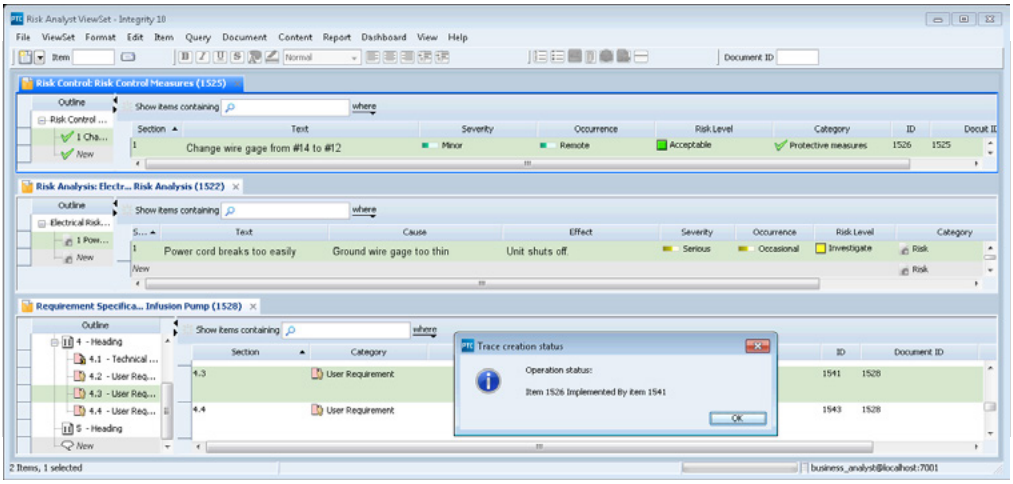
The Risk Priority Numbers (RPN) were automatically generated.



We then were able to leverage Integrity's "Risk Control" domain to create another document containing mitigation requirements. Residual RPNs were automatically generated based on our updated severity and occurrences.



Next we traced the Risk Analysis items to the Risk Control items which created for us "mitigates" trace relationships. And finally we were able to trace this all the way back to our DI requirements which generated "Implemented By" trace relationships.



In Summary

PTC's Integrity platform provides complete end-to-end requirements traceability, tests management, defect tracking, risk and hazard management, SCM and cross activity metrics for your projects. This collection of capabilities under a unified architecture help Medical Device companies develop software efficiently while achieving IEC 62304 compliance.

About SPK and Associates

SPK and Associates helps Engineering groups design and build their products better and faster. We provide exceptional technology implementation and support, specializing in the processes, applications, systems, and networks required to design, develop, test, and release products to market. We are focused on the Medical Device and High Tech vertical markets. As such, we are able to bring this domain expertise into each of our customer engagements. Contact us at 1-888-310-4540 for more information.

SPK and Associates, LLC
www.spkaa.com
Carlos Almeida
VP, Engineering

J2459–SPKandAssociates–EN–0713