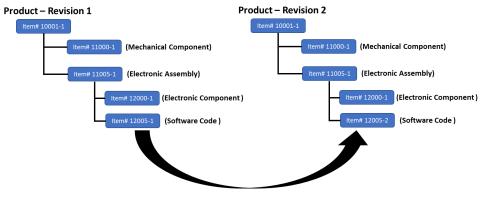## Integrating your Software Release Process in your PLM System

Development of new products in today's environment, typically has a hardware component and nowadays a software component. The acceleration of incorporating electronic controls to replace mechanical controls is happening for good reason. Electronic controls with embedded software provide greater flexibility to mechanical mechanisms, along with providing an overall financial benefit. These software enabled controls is the key product attribute that provides the platform for incorporating the Internet of Things (IoT) functionality and capabilities.

However, these two product development environments come from different origins and different methodologies when it comes to developing products. However, both the mechanical and electronic/software components are needed to complete the product design and functionality. Typically, mechanical design is configured and managed within an organization's "Product Lifecycle Management" (PLM) arena. Physical electronic components as well, are managed and configured within the same PLM arena. Software code development is typically stored in software repositories such as SVN and GitHub just to name a couple. The development of this software is typically performed in Agile toolsets, such as Jira. Jira is used to manage Sprint development of the software, capture bugs, etc.

These components and the toolsets used to manage their development and to store and configure the artifacts needed (e.g. CAD Files, Python software code, etc.) are two distinct siloed repositories. So why does this matter? What harm can this cause? To answer this question, you need to ask yourself this: "How does an organization communicate the product definition and the configuration of the product?" The answer is a configured bill of material. In this bill of material, the product needs to include all of its components, sub-assemblies and assemblies. This also needs to include the electronic components and their software controls. This is where these disparate repositories need to be integrated to include all the details needed to produce the product.

A method that can be utilized is to manage the software component within the bill of material with item numbers dedicated to software code components. These item numbers are released with the rest of the items in the product's bill of material. This now provides a complete configuration of all components - mechanical, electronic hardware and embedded software.

Keep in mind, that you need to be able to cross-reference and control the software component in GitHub configuration to the item number in the bill of material. You could provide different artifacts within the bill of material item number, such as a zip file of the actual software code or a configuration file that depicts the cross reference from the software code repository. The key is to provide a complete bill of material to the organization that includes all components and a single method at the bill of material to configure and manage changes.

We have the expertise and experience to help your organization architect and deliver this type of solution leveraging your PLM investment. If you'd like to discuss this further give us a call.

Next Month: Key Aspects of integrating Google & Alexa Home with your Connected Product Offering