

## Component Management



Modified by on 12-Oct-2016

### Related Articles

[Vault-Based Components](#)

[Choosing Physical Components through Part Choices](#)

Parent article: [Design Data Management](#)

One of the fundamental principles to Altium's Design Data Management system, and its way of approaching the design of electronics, is just how important it is to model the objects used in the design process. Not just model them but model them to a very high level of quality. This applies to all objects used in the design process, from simple off the shelf components like resistors, through to complete PCB assemblies, enclosures and even the final product.

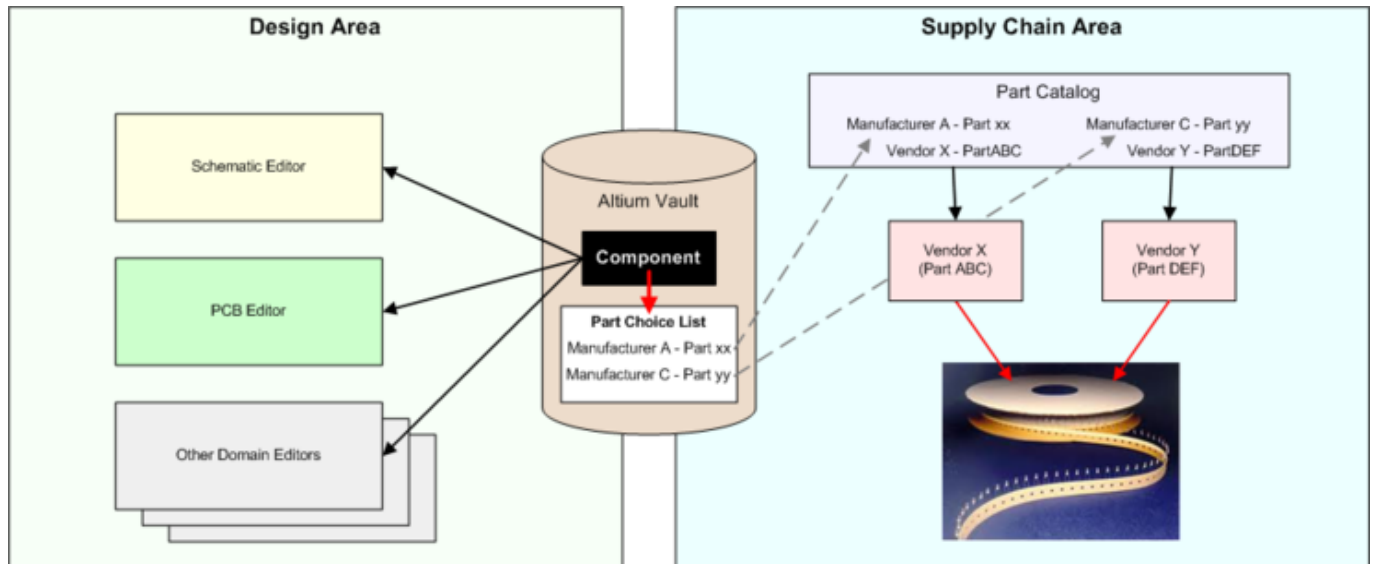
Altium Designer, with its unified design approach, has traditionally used a component model that extends across all aspects of the electronics design process. However, to seamlessly fit the process of electronics design into the encapsulating product development process as a whole, this model needs to evolve - extending to cover other aspects including other design processes, in particular MCAD and Industrial Design, as well as business processes - such as procurement and manufacturing - that intersect with the product development process.

This evolved object model is known as the *Unified Component Model*.

This 'next-generation' component model effectively maps the concept of a design component - in the traditional electronics design arena - to the component as seen by the rest of the organization in the bigger 'product arena'. A truly 'Unified Component' model that not only represents the component in the different design domains (Schematic Capture, 2D/3D PCB Layout, Simulation, Signal Integrity) but also facilitates choices of the desired physical components - real-world manufactured parts - at design-time, offering a significant improvement in terms of procurement cost and time, when manufacturing the assembled product.

Under this modeling paradigm, the design component, as seen by the designer, is separated from the Manufacturer and/or Vendor parts. This information is not defined as part of the component. Instead, a separate vault Item - a Part Choice List Item - is used to map the design component to one or more Manufacturer Parts, listed in a *Part Catalog*, which in turn can be mapped to one or more Vendor parts, allowing the designer to state up-front, what real parts can be used for any given design component used in a design.

These components, along with their part choices, are stored in a target Altium Vault.



The term 'Unified Component' depicts the extension of a design component into the bigger product arena through dedicated part choices, that map that design component to the real-world manufactured parts that have to be sourced by the procurement team. This gives the procurement team a beneficial and timely 'heads-up' on what those parts should be.

## Key Benefits

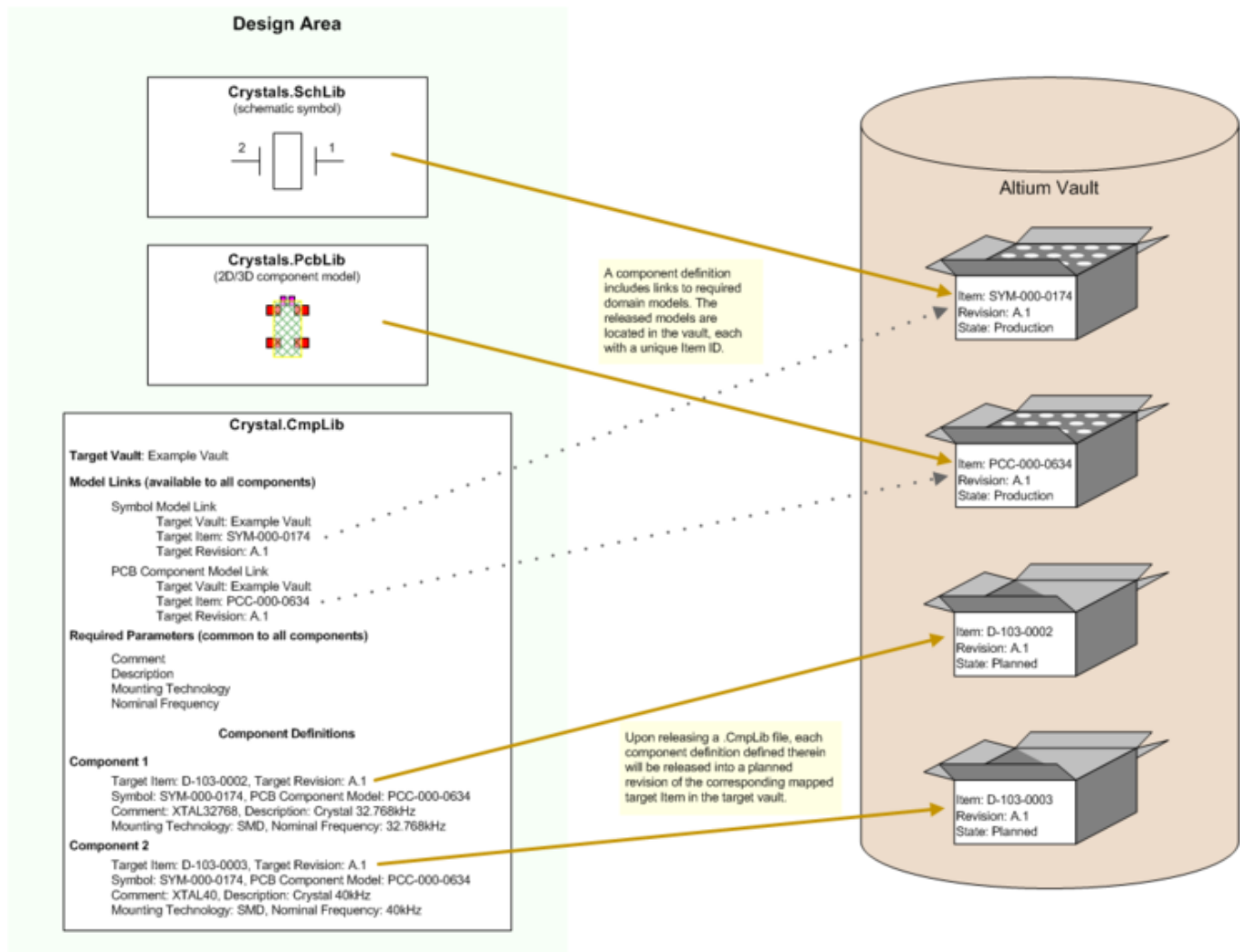
- Company-certified design components. Components are formally modeled as file-based definitions that can be released into a vault for re-instantiation into a design project. Revision-controlled and lifecycle-managed, a company can authorize the 'set' of components that can be formally used by their designers.
- Design-time choice of physical components. For any given released Component Item, you can choose which Manufacturer Parts can be used to implement that component when assembling the board. A perfect 'heads-up' to the procurement team, these 'choices' are specified in a revision-controlled and lifecycle-managed Part Choice List Item, which essentially stores links between the Component Item and chosen Manufacturer Parts. The manufactured parts themselves are referenced in a dedicated *Part Catalog*, complete with links to Vendor Parts - the purchasable components themselves.
- Real-time supply-chain information - fed back from the supplier's web services, to let the designer know the current costing and availability of the chosen parts, and from all vendors that sell those chosen parts (as defined in the Part Catalog).

## Creation and Release

On the design side, each design component released to a vault is specified using a source *Component Definition*. A component definition is simply just that - a definition of a particular design component. A definition that ties together the required models and parameters for that component in a clean and ordered fashion. Each component definition on the design-side maps to an Item - a Component Item - in the target vault. To put this another way, you are defining the source definitions that will, when released, provide a set of components which you can re-use again and again in your designs.

In terms of storage, component definitions are created and managed within a dedicated Component Library file (\*.CmpLib). A single Component Library file can be used to create (and therefore map to) one or more unique Component Items, by entering one or more component definitions. Each component definition will have a common set of parameters and links to required domain models.

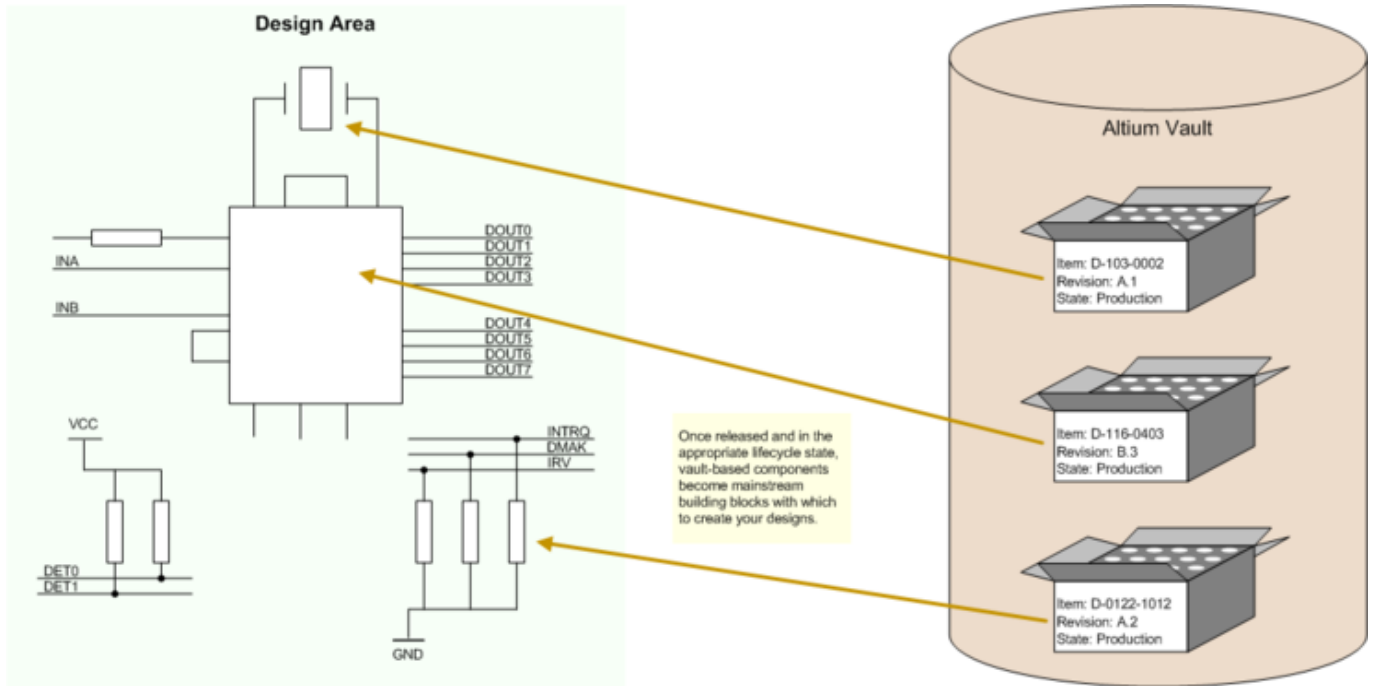
Having defined the component definition(s) as required in the Component Library file, you simply release it. The release process is simply the act of generating a new revision of each targeted Item. What's more, the Items are created on-the-fly as part of the release process – define-and-go as it were.



Definition of a component. All data required for a component is specified on the design-side within a dedicated Component Library file, including links to the lower-level released domain models and parametric information.

## Re-use in a Design

Once released and available in the vault, components can be re-instantiated into any new design project, manufactured into prototypes, or used for production runs. In addition, the concept of component certification is made possible as the components are formally revised and lifecycle-managed. This allows the organization to specify the state of its components and what they can be used for (design, prototype, production, etc). From a design perspective, this results in the creation of vault-based libraries, containing a formal collection of components that have been company-approved for use in each new design project embarked upon within that company.



Design using components that have been certified for use!

The beauty of using certified components in your designs is that when it becomes time to change the lifecycle state of your board design, the integrity of the design becomes greater still, since a design can only be released to "Prototype" or "Production" provided the components it uses are also in a corresponding state. Put another way, you wouldn't start to produce that assembled board if the components are only at a "Design" stage!

And, if we take this to the finest level of granularity in the component management arena itself, the system will not allow you to promote the lifecycle state of a component in the vault until all of its referenced domain models are in a corresponding correct state to be able to do so. In other words, a parent component cannot be further in its lifecycle than its child models.

The highly relational structure of the released data in an Altium Vault lends itself to powerful 'where used' capabilities. At any time, you are able to see where a particular child Item is used, in terms of parent Items in the vault. So for a given domain model, you can quickly identify which component Items reference it. For a given component Item, you can see which designs it has been used in, which managed schematic sheets, and so on.



If a component is deprecated or made obsolete for any reason, you can quickly identify and pull-up the schematic sheets and board-level designs in which it is placed, and update and re-release those Items as new revisions using another, equivalent approved component. And why re-invent that proverbial wheel, when you can find a component that you need to use and see which designs it has already been used in. If those designs are at a production state, that component has been proven already and is therefore good-to-go. It can be re-used in your next design with confidence.

In summary, the unified nature of a vault-based component, through the chosen part choices made for it, ultimately create a link from that component, all the way through chosen Manufacturer Part(s), and on to the Vendor (Supplier) Parts that each itself references. From the designer's perspective, the component is hooked directly into the supply chain. This allows real-time data to be made available - fed back from the Supplier's web services - to let the designer know the current costing and

availability of the chosen parts, and from all Vendors that sell those chosen parts.

---

**Source URL:** <http://techdocs.altium.com/display/DMAN/Component+Management>