Getting Started with Altium Vaults

Summary

This article provides an introductory overview to Altium Vault technology and how it integrates with the overall design flow.

One of the greatest challenges of developing an electronic product is getting the components correctly specified. It is much more than specifying the right electrical properties – each component must also be available, in the volume required, over the projected life of the product, and at the right price. The foundation of component management is the correct storage and management of *all* of the component data. Beyond the component models and parameters you need visibility into the supply chain to confirm price and availability when the component is needed, and also usage data for every component across all designs. You also need to be able to revise components, and manage the revisions and lifecycle of each component.

This takes us to the question, *just what is a component?* Sure the resistors and ICs are components, but what about that blank board ready to be loaded and tested, or the loaded board, ready to be slotted into the case with the other boards? Well those boards are like components too, and so is the case.

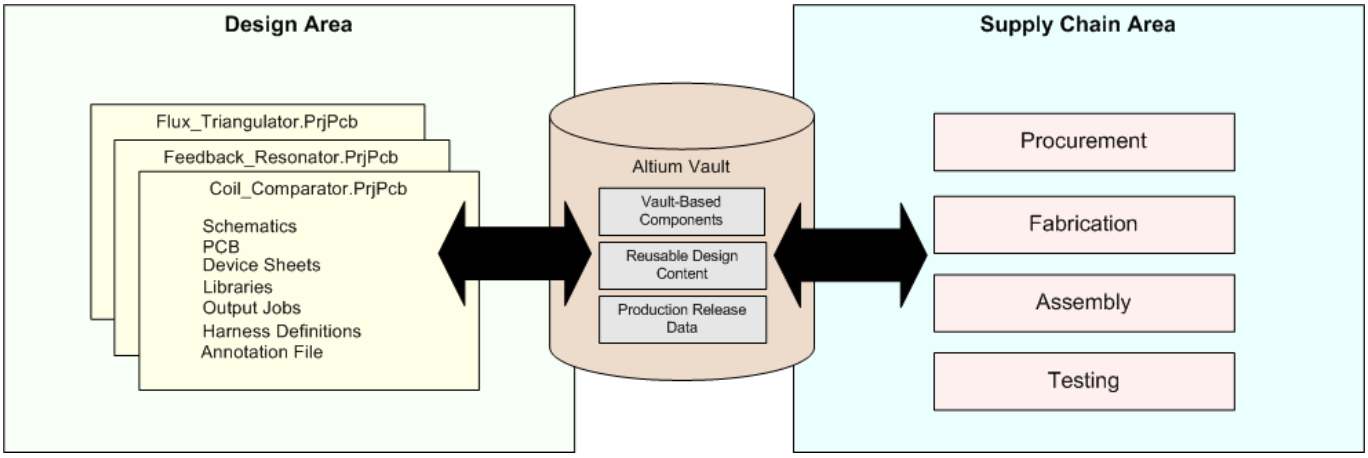
Altium solves this data management challenge with its *Vault Server* technology. An *Altium Vault* is used to store all of your company's qualified, released, ready-to-use design entities, from the components through to the loaded boards themselves.

What is an Altium Vault?

An Altium Vault (or Altium Vault Server to give it its full title) is a server-based engineering content management system. It is the foundation upon which the various sub-systems of Altium's smart data management technologies are built. A distinct design solution in its own right, an Altium Vault Server works in harmony with Altium Designer to provide an elegant answer to the question of handling design data with secured integrity.

An Altium Vault not only provides rock-solid, secure storage of data, but also enables re-release of data as distinctly separate revisions – essentially tracking design changes over time, without overwriting any previously released data. It also caters for the lifecycle of the data to be managed, allowing people that need to use the data to see, at-a-glance, what stage the data have reached in their 'life' and therefore what they can be safely used for.

The vault essentially provides the common ground between the Design and Supply Chain areas, facilitating the secure handling of data with high integrity, while providing both design team and supply chain access to that data as needed. And this access can be controlled, so that only those requiring the information can get at it, and only those that have authority to change the data or their state can do so. Through the vault, the teams in both areas are better informed, have a real-time picture of the progress of the design at the wider product level, and have access to data that are not only guaranteed in their integrity, but also validated and managed in terms of lifecycle – so that only the data that are authorized to be used for a specific application, are used.



An Altium Vault provides the common ground between Design and Supply Chain areas.

What can be Stored in a Vault?

An Altium Vault is designed to store all of your ready-to-use electronic product data – data that completely defines every component, through to all of the data needed to fabricate, load and assemble the boards. The vault is for storing qualified data, as distinct from design data and source files that are still being developed. Source design data files are stored in a separate *Design Repository*.

The various data entities (released from the Design Area or otherwise) are represented in an Altium Vault by unique Items. An Item simply represents a specific object, and is uniquely identified through use of an assigned Item ID – which becomes the identifier of that Item throughout the design process and out into procurement and manufacturing.

An Altium Vault then, is a centralized storage system into which all data, for each Item, are stored. What exactly is represented by, and stored in an Item can vary, depending on what has been mapped to that Item. A design project, for example, can be the source of multiple bare or assembled boards. Each of these is a distinct physical object built by the production team, and so would be represented in the vault as different Items, each with its own unique ID. A design itself might use vault-based components, which themselves are released entities – each a separate Item and each assigned their own unique Item ID. Drill down further and you reach the finest level of granularity, with the schematic symbols and other domain models – used in the definition of those components – also released and represented as uniquely-identifiable Items.

Each Item in the vault is stored as a series of revisions. Each revision contains data that are used to represent, or to build a particular version of that Item. Each time a change is made to the source design data, a new revision of that Item is created in the vault, ready to accept (store) the generated data.

To cater for the storage of these differing types of Item a vault can be divided, conceptually, into three distinct areas, or 'zones' of content:

- **Component Management** – an area of a vault dedicated to the management of released Component Items and the domain models they reference (Schematic Symbol Items, PCB Component Items, etc).
- **Design Content Management** – an area of a vault dedicated to the management of released reusable design content (Managed Sheet Items, Template Items, Reference Design Items).
- **Production Release** – an area of a vault dedicated to the management of released data used in the production of physical board design entities (Blank Board Items and Assembled Board Items).

Vaults use folders, or *zones* to organize the different types of data. Access can be controlled down to the folder level, allowing a single vault to be used for an entire organization.

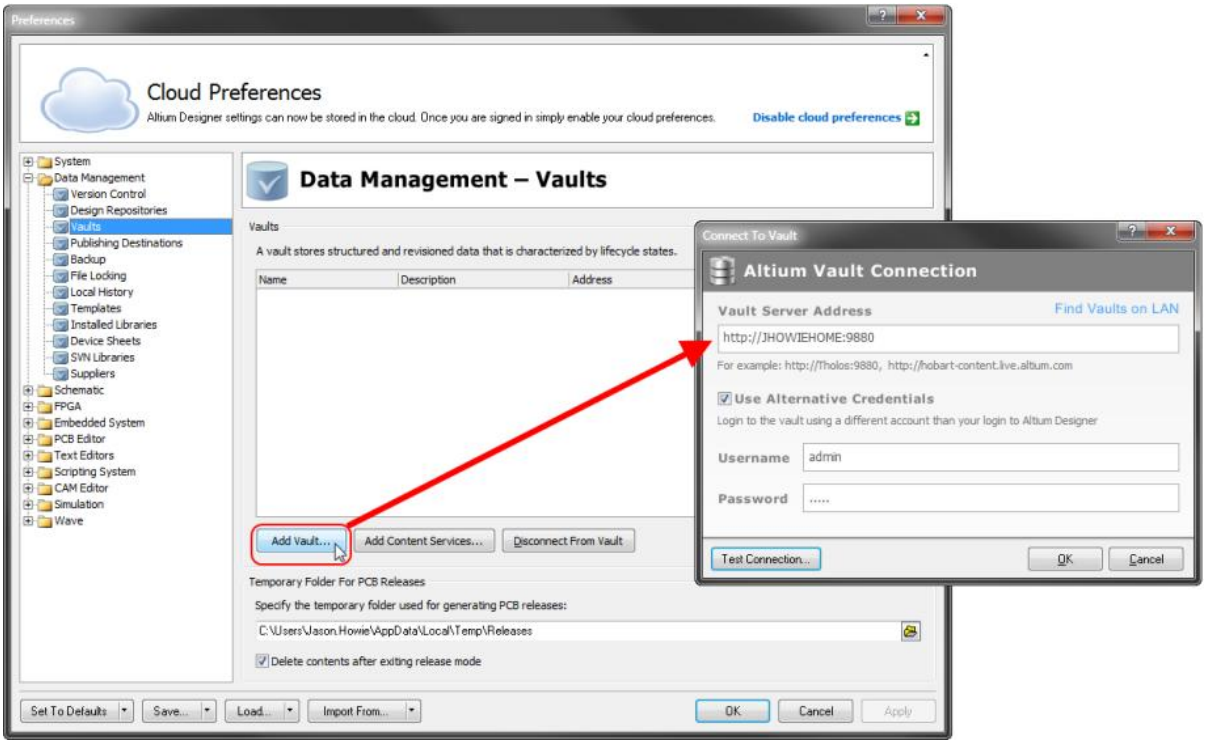
Connecting to a Vault...

Once a Vault Server is installed, you can access it. And because it is delivered as a web server – and is separate to Altium Designer – it can be accessed not only from within Altium Designer, but also from your favorite web browser, anywhere in the world that you can get a connection.

...from Altium Designer

Altium Designer provides a design-side user interface into your Altium Vault. Before a vault can be used from within Altium Designer, a connection must be established from Altium Designer to that vault. Connections are created and configured on the **Data Management – Vaults** page of the *Preferences* dialog (**DXP»Preferences**). Any number of vaults can be connected to.

Click the **Add Vault** button to access the *Connect To Vault* dialog.



Connect to your vault in the Preferences dialog.

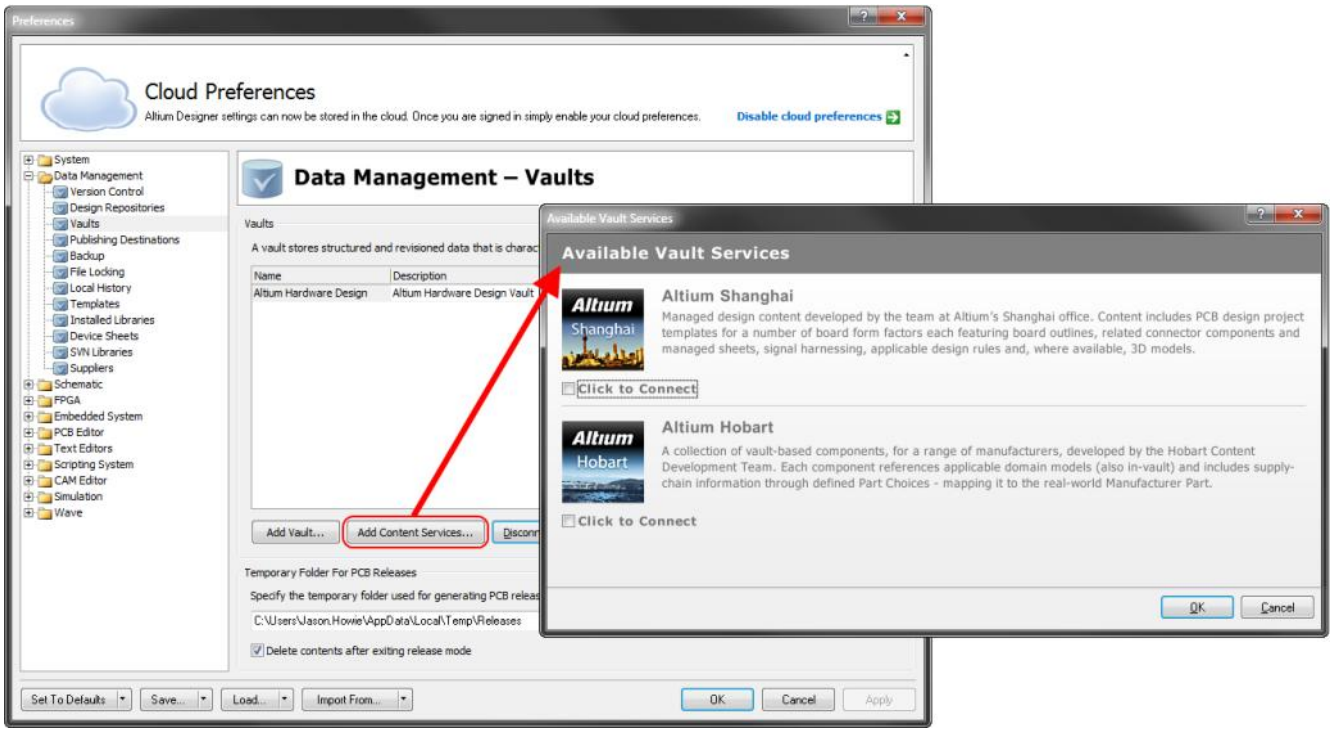
Use the **Vault Server Address** field to enter the location URL of the vault server to which you wish to connect. An Altium Vault that has been installed on your Local Area Network can be quickly discovered by clicking the **Find Vaults on LAN** link.

If you are using an Altium Personal Vault, connections are anonymous so a *username* or *password* is not required. For the Altium Vault Server, a *username* and *password* may be required if that Vault has been configured not to use the Windows Domain for authentication.

Adding Vault Services from AltiumLive

Should you wish to quickly add connections to vaults that are part of the AltiumLive ecosystem, and to which you have authenticated access to, you can do so by clicking the **Add Content Services** button. This will give you access to the *Available Vault Services* dialog. The dialog presents vaults running within AltiumLive.

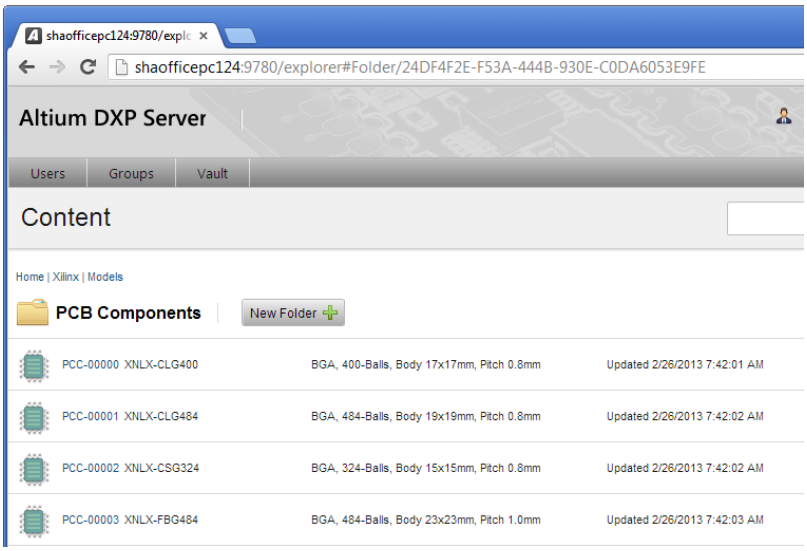
To connect to one of these vaults, simply enable the **Click to Connect** option beneath the vault's icon, which will change to **Connect**. Clicking **OK** will add that vault to the list of vaults you are currently connected to.



Access all vaults available to you as part of AltiumLive Vault Services.

...from a Web Browser

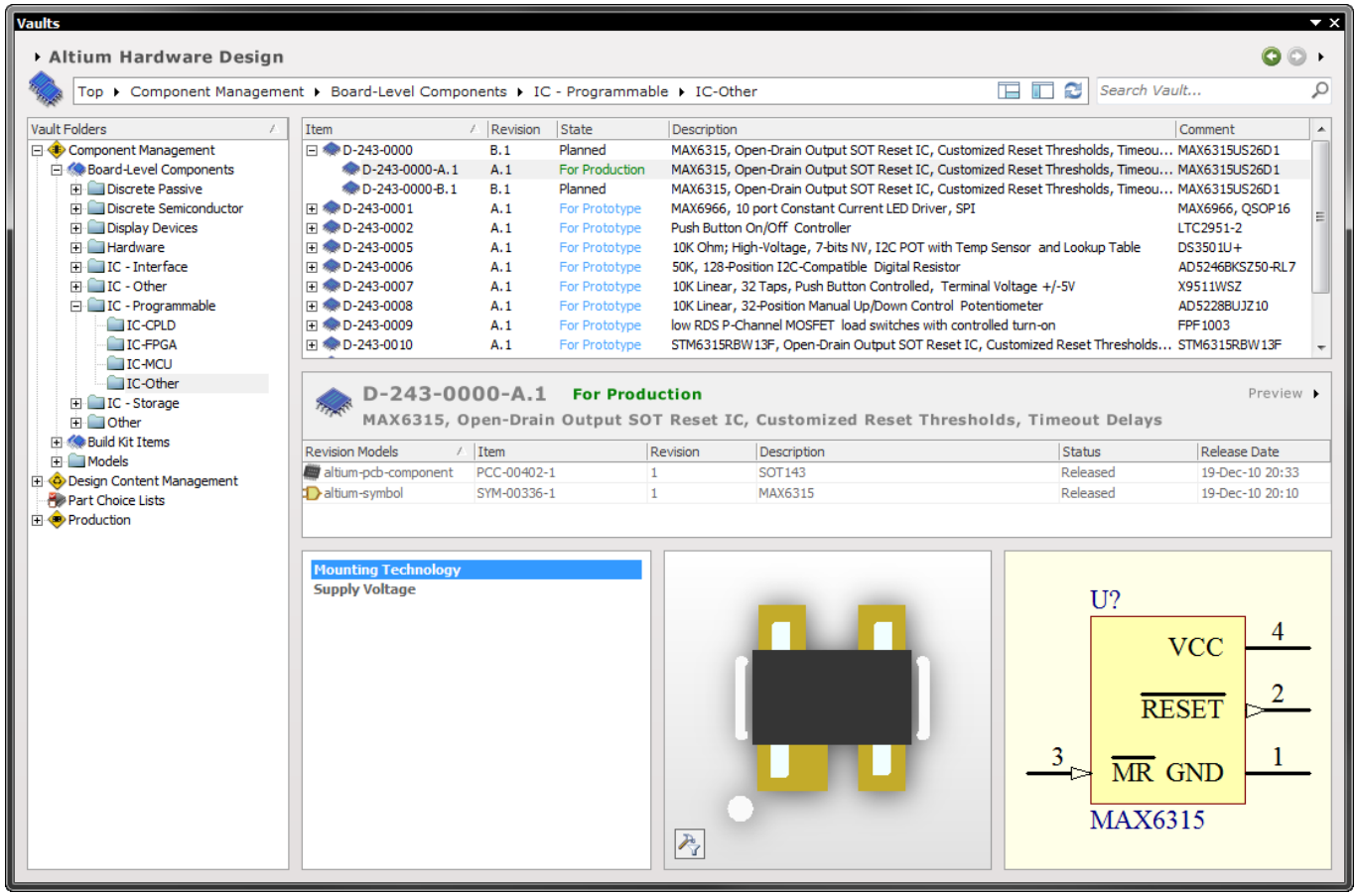
The Altium Vault Server also supports accessing the vault via your preferred web browser, providing you have the URL to that vault. The browser interface allows for administration of vault access, as well as some manipulation of vault content itself, including folder definition. This allows people other than the designer to contribute to the management and maintenance of vault data.



Access the vault via your preferred web browser.

I'm Connected to a Vault – now What?

Once connected to a vault, you can access and work with it. Vault management and usage is performed from Altium Designer's **Vault Explorer** panel. To display the panel, click the **System** button down the bottom right of the Altium Designer workspace, and select **Vaults** from the menu. The **Vault Explorer** panel will display, size and position it to suit your needs.



The Vault Explorer panel gives access to your valuable company data stored in your Altium Vault.

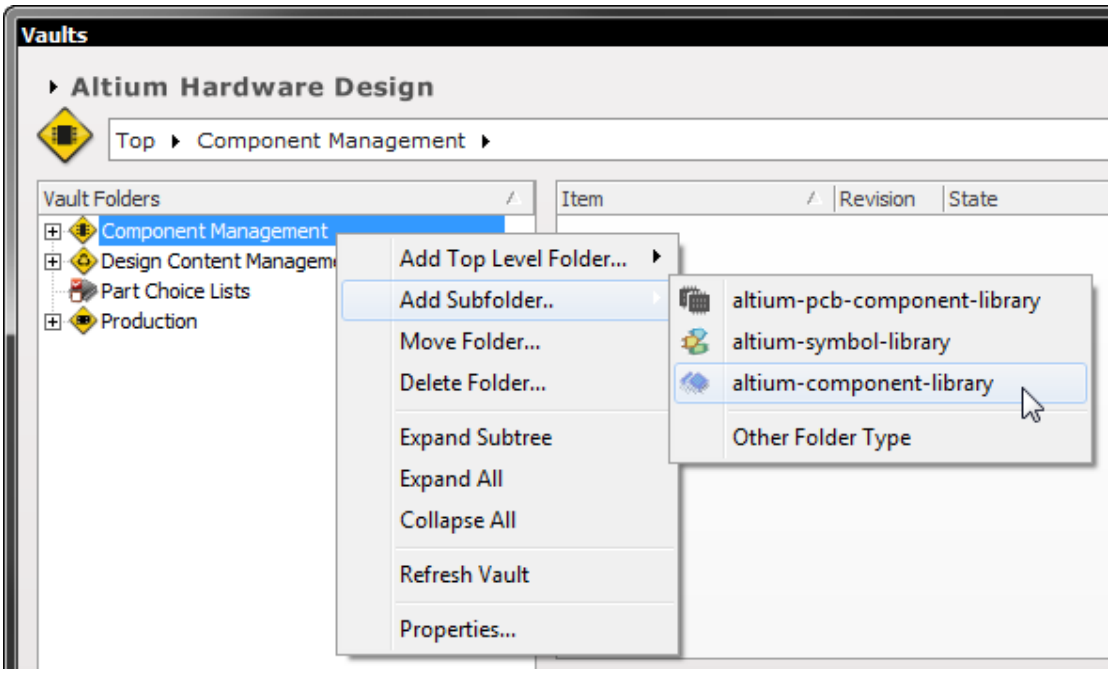
The **Vault Explorer** panel is the primary interface between Altium Designer and a connected Altium Vault, making it an integral part of the *Design Data Management System*. From the **Vault Explorer** panel you create and manage the organizational structure used in the vault and also create any number of Items, each representative of a design-side object. From the **Vault Explorer** panel you also access detailed Item information, where you manage the *revision* and *lifecycle* settings for the Item. The **Vault Explorer** panel also gives access to *Where Used* and *Supply Chain* detail.

Choosing the Working Vault

The **Vault Explorer** panel can only interface to one vault at a time. The field at the top-left of the panel indicates which vault you are currently exploring – the vault whose content you will be able to browse and modify. To select a different vault, to refresh the displayed data for all vaults, or to access the **Data Management – Vaults** page of the *Preferences* dialog, click the icon to the left of the vault name (or on the vault name itself) to access a menu of top-level vault management controls, controls that enable you not only to choose the vault you wish to work with, but also to manage the connections to vaults – both existing and new.

Organizing the Vault

You maintain order within your vault by creating a tree of folders for the various Items stored in the vault. A number of folder types are available, use these to categorize the vault content. Right-click in the **Vault Folders** region of the **Vault Explorer** panel to add folders and define a storage structure of folders and sub-folders that suits your organization.

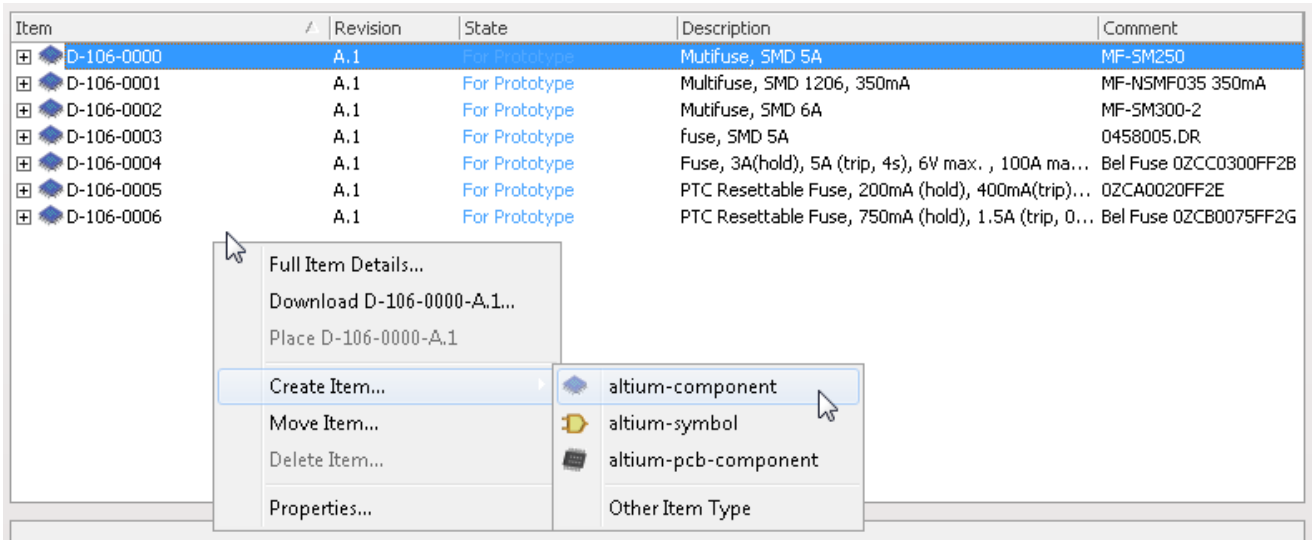


The right-click menu includes commands for defining a folder hierarchy, to bring a sense of logical order to the vault.

Within a vault's folder hierarchy, different folders can be used to store different types of Items. To declare the intended purpose of any given folder – that is the type of content it will be used to contain – the folder's *type* can be specified. This is done using the **Folder Type** property, which can be set when creating the folder, or later by editing the folder properties. The folder type has no bearing on the actual content of the folder. It simply provides a visual 'clue' as to what is stored in a folder and can be beneficial when browsing a vault for particular content.

Vault Items

Each entity that is stored in the vault is called an Item. To support the need to be able to update an Item over time, what is actually stored in the vault is a series of Item Revisions, that is specific revisions of that Item, whose name is based on the chosen *Revision Naming Scheme*. Each new Item is created in a folder, once you have defined the required folders within the vault you can create new Items in the selected folder, in the **Item** region of the panel.



Right-click to create a new Item in the vault.

Different Items are used to store and represent different types of release data. One Item could represent a schematic symbol, another a PCB component model, while another could contain the generated data from a released board design configuration, along with source design snapshot. To declare the type of content an Item (or rather its revisions) will be used to contain, its

Content Type property needs to be specified when creating or editing that Item. To put this another way, you are in essence specifying the *Item Type.*

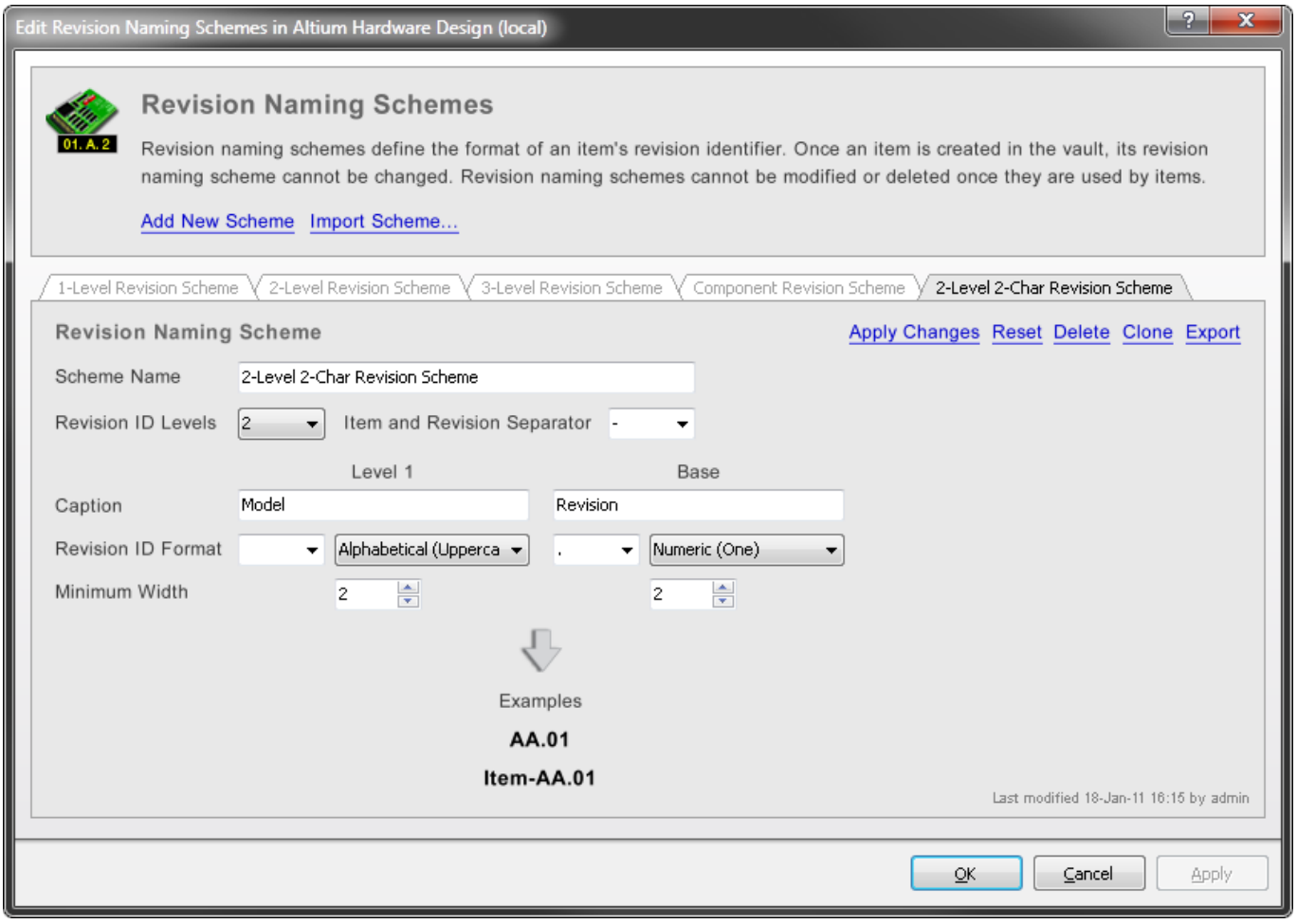
To create an Item, right-click in the **Item** region of the **Vault Explorer** panel and select one of the commands in the **Create Item** sub-menu. The sub-menu will change to display the most applicable content type, based on the type of folder in which the Item is being created.

Items can be created manually and directly in a vault, as described. In some cases, a release process will create Items automatically – on-the-fly as it were.

Revision Naming Schemes

For each Item in the vault there will be at least one revision of that Item. Each time the source data used to create that Item is changed, it is re-released into the vault as the next Item-Revision. Revisions are identified by the *Revision Naming Scheme*, this scheme is selected when the Item is created. Note that different Items in the same vault can have different Revision Naming Schemes assigned to them.

Each vault has a set of pre-defined Revision Naming Schemes, these can be edited and new schemes can also be defined for that vault. The Naming Schemes can be viewed and edited through the *Edit Revision Naming Schemes* dialog. This dialog can be accessed from the **Data Management – Vaults** page of the *Preferences* dialog. Select the required vault, click the **Properties** button, then choose **Edit Revision Naming Schemes** from the drop-down menu.

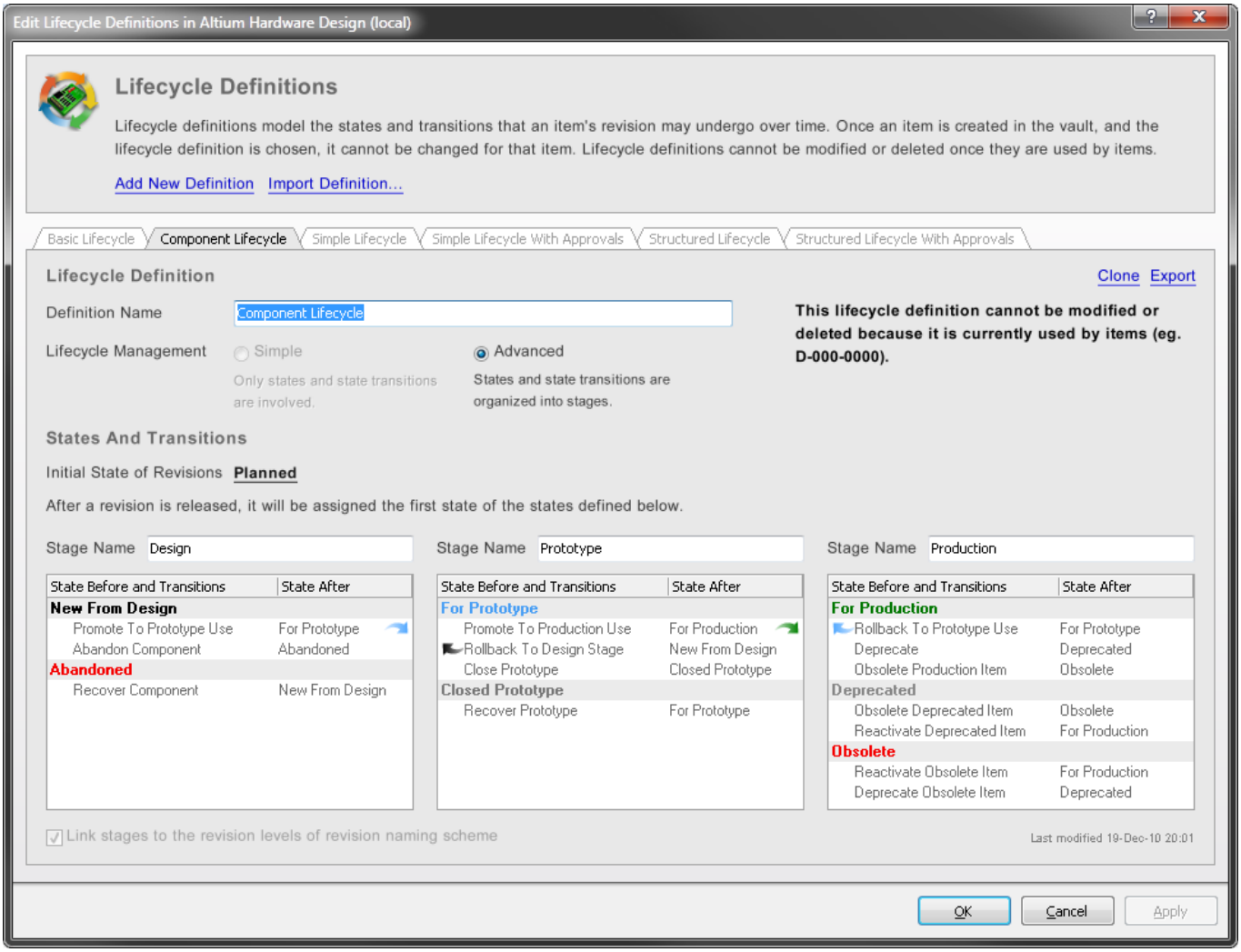


The Edit Revision Naming Schemes dialog presents the default naming schemes, as well as user-defined ones.

Lifecycle Definitions

Each Item in the vault also has a *Lifecycle*. The Lifecycle is used to model the state transitions that an Item Revision may undergo over time. Lifecycle definitions are also defined at the vault level, then chosen and applied to the individual Item when the Item is first created. Each Item can have a different Lifecycle Definition assigned to it.

The Lifecycle Definitions are viewed and edited through the *Edit Lifecycle Definitions* dialog. This dialog can be accessed from the **Data Management – Vaults** page of the *Preferences* dialog. Select the required vault, click the **Properties** button, then choose **Edit Lifecycle Definitions** from the drop-down menu.



Examine and edit the Lifecycle Definitions in the Edit Lifecycle Definitions dialog.

Working with a Vault – the Design Flow

From a design perspective, an Altium Vault serves as both a destination repository and a source repository. For example, it receives the domain models, such as the schematic symbols and PCB 2D/3D component models, as an output of the process of designing and qualifying these models. Once qualified and stored those models are then used as part of the source for a fully described component, when that component is qualified its data are then stored back in the vault. That qualified component is then used in a design, which is also released back into the vault.

The following sections take a look at this 'design flow', including the features and functionality in the software that enables you to release data to a target Altium Vault, place data from that vault back into your designs, and take advantage of other powerful vault-related features as you design your next innovative electronics product.

Creating and Releasing Domain Models to a Vault

From a designer's perspective, a vault-based component gathers together all information needed to represent that component across all design domains, within a single entity. It could therefore be thought of as a container in this respect. A 'bucket' into which all domain models and parametric information is stored. In terms of its representation in the various domains, a vault-based component doesn't contain the domain models themselves, but rather links to these models. These links are specified on the design-side, as part of the source component definition – from which the released Component Item is generated. As such, before you can delve into the process of defining and releasing components, you must first ensure that all the domain models themselves have been created and released.

Preparing the Schematic Symbols and PCB 2D/3D Component Models

A drawn schematic symbol within a Schematic Library document (`*.SchLib`) on the design side is mapped to a Schematic Symbol Item in an Altium Vault. Each release of the library stores the symbol model data into a new revision of that Item. Create a new Schematic Library file (**File»New»Library»Schematic Library**) and add/draw symbols as required, or use an existing library. A single library may contain any number of schematic symbols, with each independently linked to a unique Item in the target vault.

A PCB 2D/3D component model within a PCB Library document (`*.PcbLib`) on the design side is mapped to a PCB Component Item in an Altium Vault. Each release of the library stores the model data into a new revision of that Item. Create a new PCB Library file (**File»New»Library»PCB Library**) and add/draw the 2D footprints as required. Add also additional 3D body information to each footprint if applicable. A library may contain any number of PCB 2D/3D component models, with each independently linked to a unique Item in the target vault.

> Although a single library can contain multiple symbols/models, from a version control perspective it is best practice to have one symbol/model per library file. This allows you to check out and modify just the symbols/models you need to modify, without registering a version change to an entire, single source. A tool to split schematic and PCB source libraries into individual libraries, each containing a single schematic symbol or 2D/3D component model, is available from the *Component Release Manager* – as part of a set of migration tools.

With respect to schematic symbols, one important thing to observe here is that you are creating purely a schematic symbol – the representation of the higher-level vault-based component within the schematic editing domain. It is not a 'schematic component' as defined for use in integrated libraries, where other models and parameters are defined as part of that schematic component. A vault-based component needs only the graphical depiction of the symbol. It will include links to other domain models and parameters as part of its own definition.

> If the symbols are coming from old library sources, you do not need to delete any linked model and parameter entries. The system will simply strip these during the process of releasing the symbols to the target vault. This 'stripping' applies to the released data – the design-side source schematic component definitions are not altered in any way, allowing you to still use these same libraries with other component and library management methodologies supported by Altium Designer.

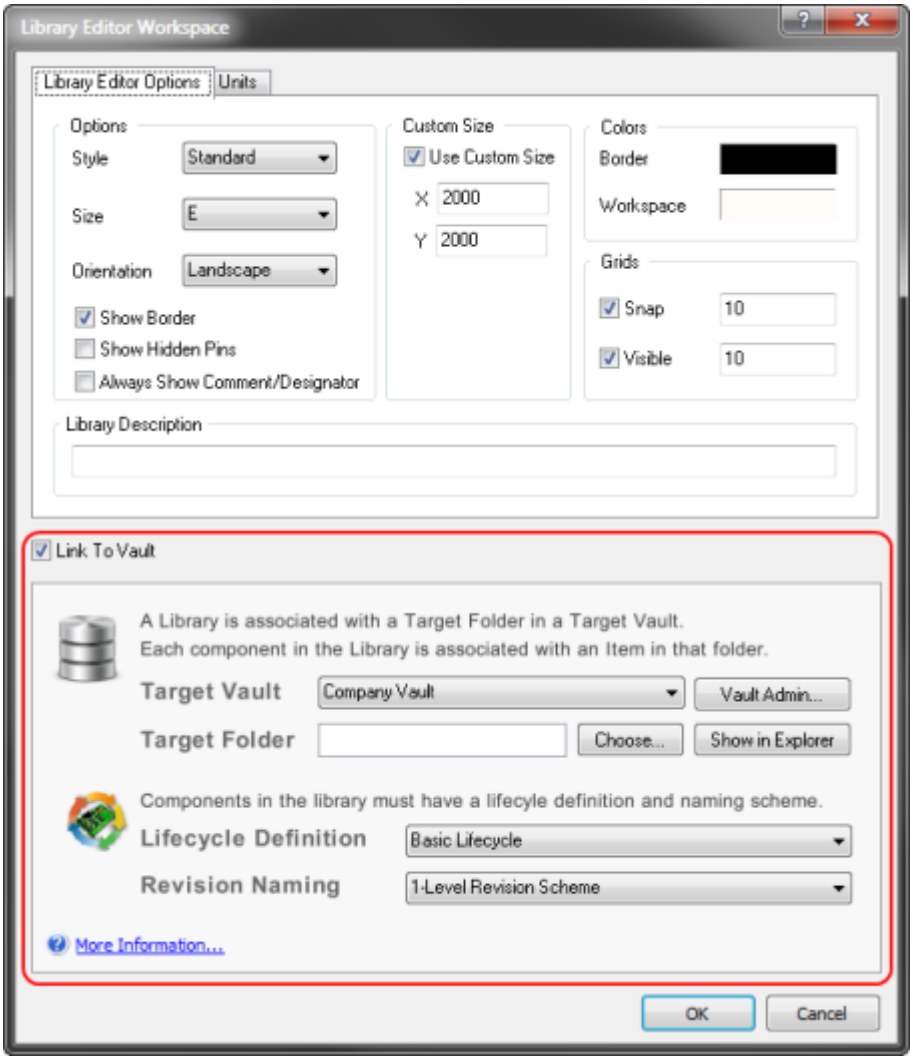
Once the required schematic symbols and 2D/3D component models are ready, you can proceed to release them to a target vault.

Linking and Releasing the Active Library Document

Releasing the active Schematic or PCB source library to a vault is a two-step process:

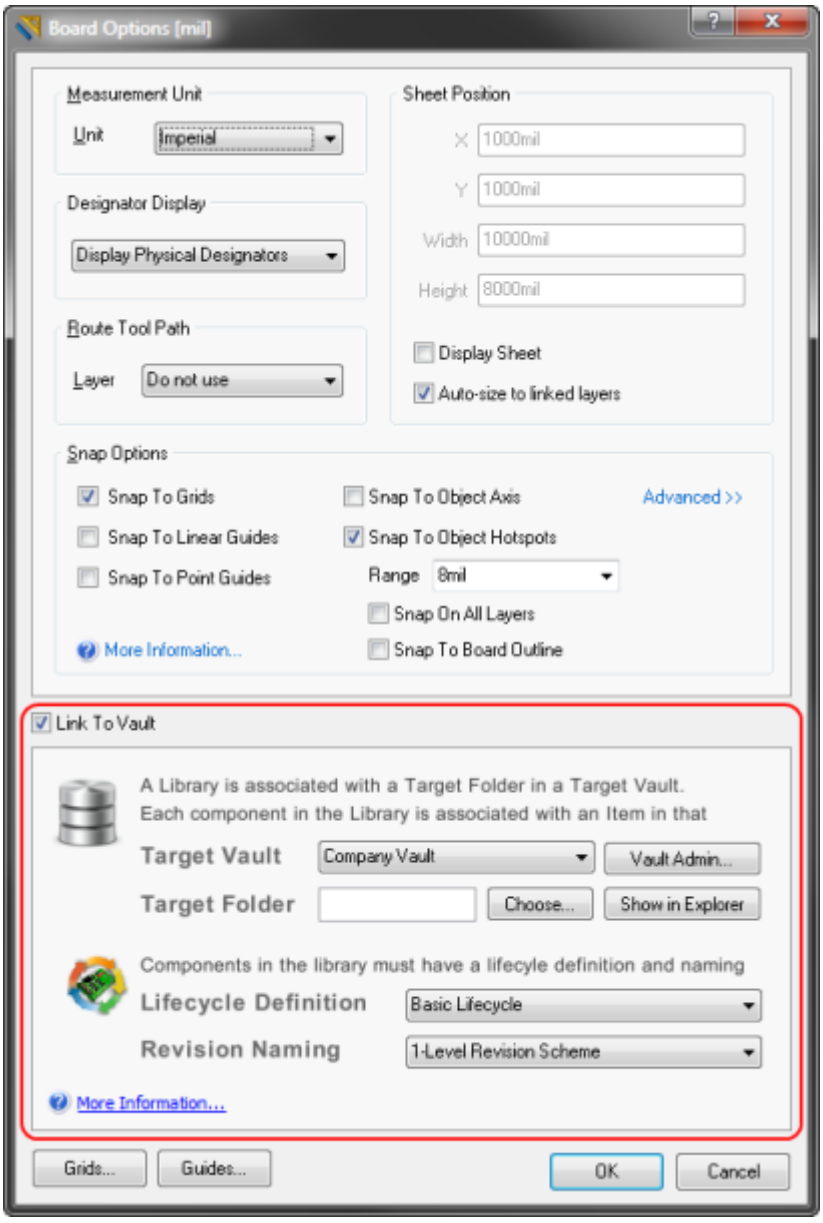
1. Link the library file to the required target vault, and a specific folder within that vault.
2. Release the required symbol(s)/model(s) into the next planned revision(s) of the corresponding target Item(s).

To link a library, run the **File»Link Library to Vault** command. For a Schematic Library, linking is performed in the *Library Editor Workspace* dialog.



Link the active Schematic Library to the required target vault, and folder within that vault.

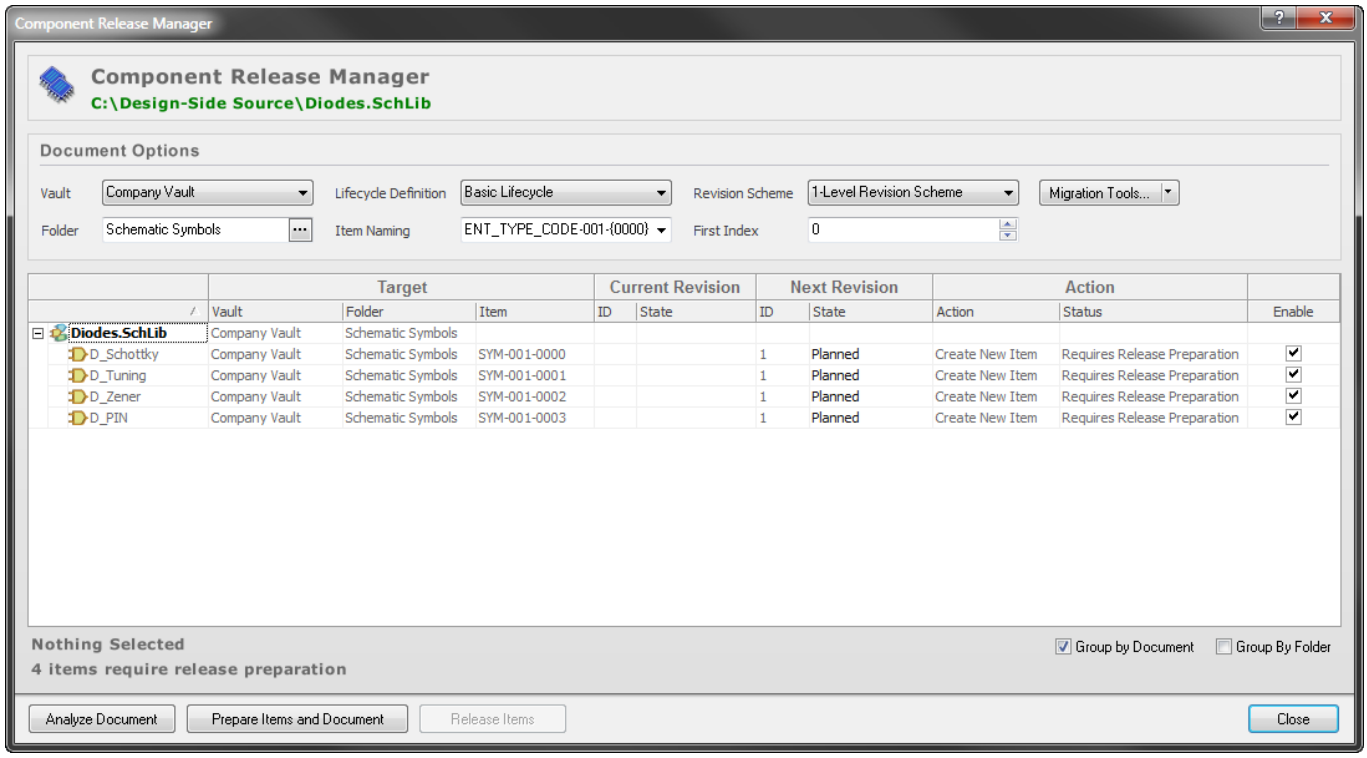
For a PCB Library, linking is performed in the *Board Options* dialog.



Link the active PCB Library to the required target vault, and folder within that vault.

Use the **Target Vault** field to nominate the vault into which you want to store released data for the symbol(s)/model(s) within the library – choose from a list of vaults you currently have an active connection to. Use the **Target Folder** field to nominate the folder into which the corresponding Items for the symbols/models being released will be created. Click the **Choose** button to access the *Choose Folder* dialog. This dialog essentially provides you with a folder-specific view into the target vault. Either browse to, and select an existing folder in the vault, or add a new folder. Use the **Lifecycle Definition** and **Revision Naming** fields to choose the type of lifecycle management to be used for the created Items, and the naming scheme employed for their revisions, respectively.

To release the symbol(s)/model(s) in your library, simply use the **File»Release To <TargetVaultName> Vault** command. The *Component Release Manager* dialog will appear. This is a modified version of the full *Component Release Manager* dialog, the difference being that in this context it is used to release the single, active library document, rather than batch-release multiple libraries in a nominated source folder.

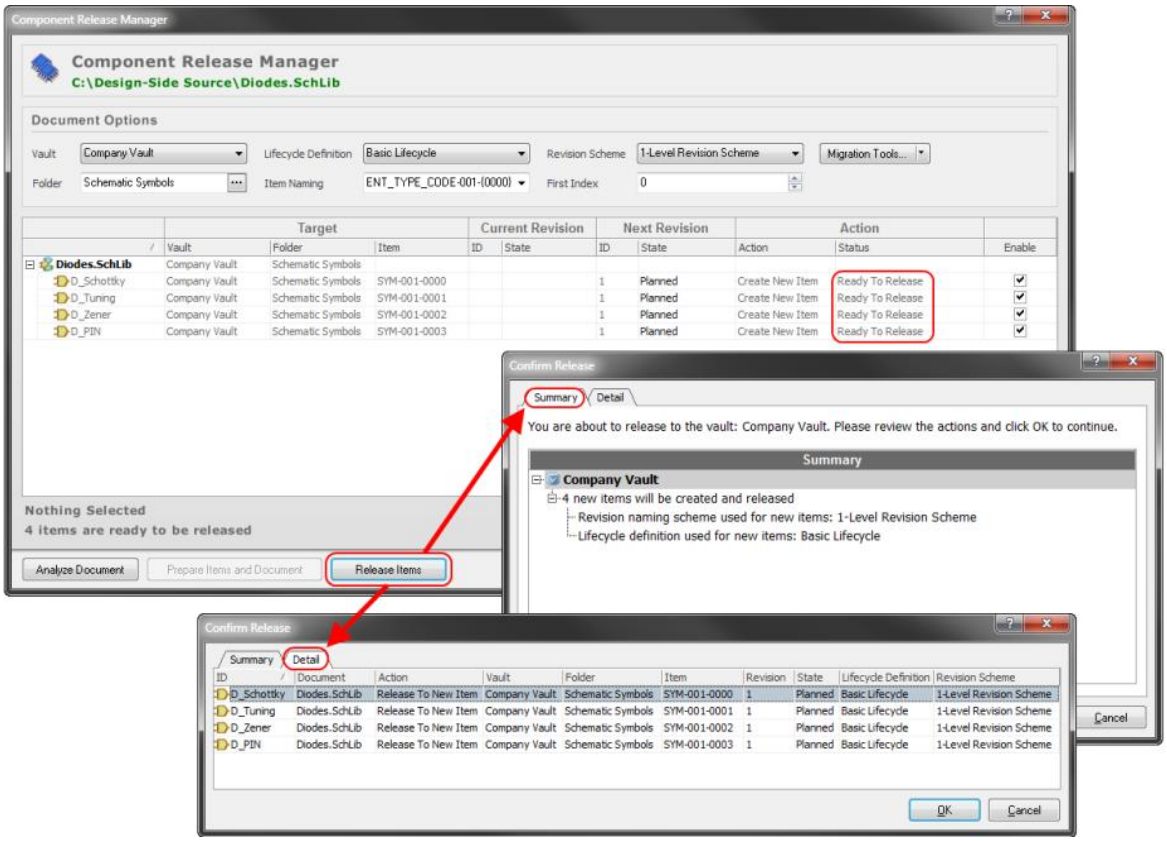


Release content in the active library document using the Component Release Manager.

The main grid in the dialog presents all information relating to the link between a source symbol/model and its vault-based Item, as well as the actions that will be taken if the source entity is included in the release process. If you have split your libraries, then only a single symbol/model entry will be presented. If you haven't, simply choose which of the source symbols/models to include in the release, by toggling the associated **Enable** option accordingly.

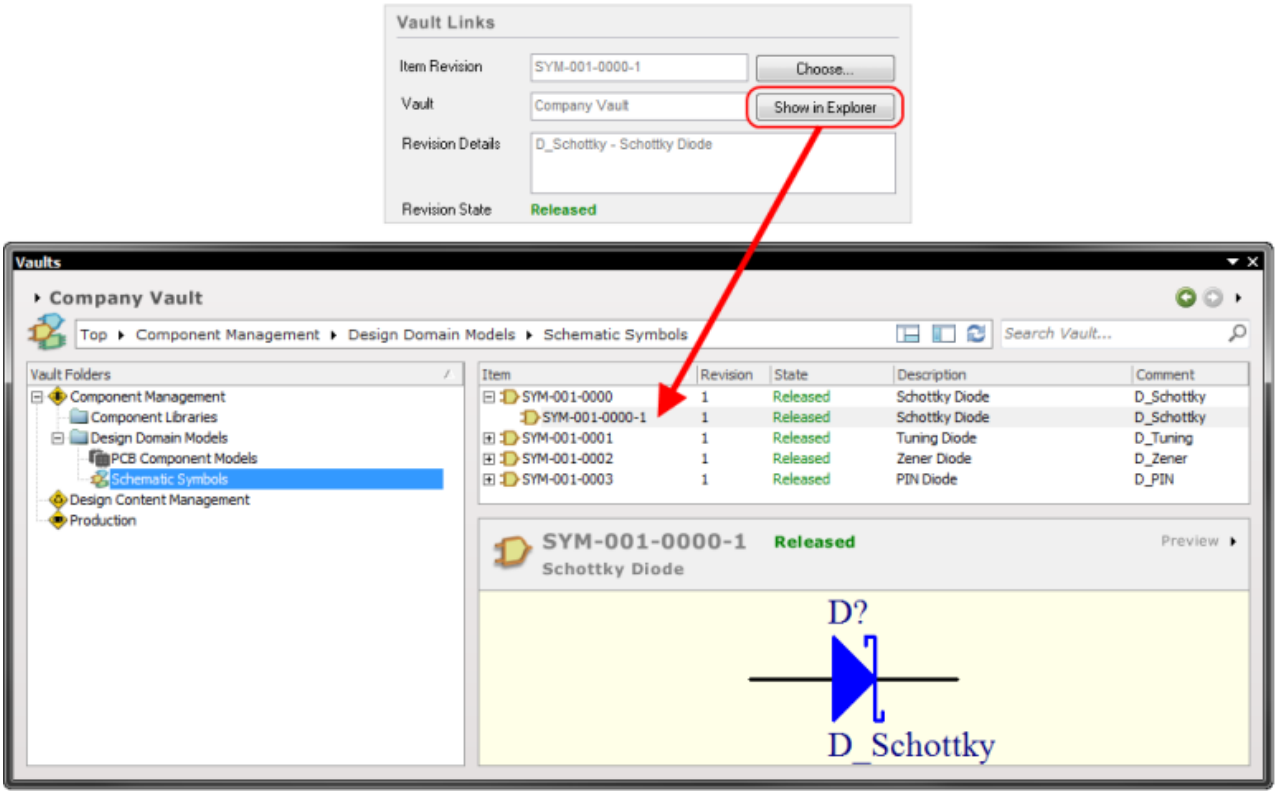
Then click the **Prepare Items and Document** button. On the vault side Items themselves will be created, with initial revisions in the `Planned` state. If re-releasing to an existing Item, the next planned revision will be created. This is important, so that these Items, and the unique IDs assigned to them are essentially locked-down – so no-one else releasing to the vault can create Items with these same IDs, or release to the same revision of an existing Item, before you get a chance to finish your release process! On the document side, the source library document is saved with the link information. This is the link to the target vault and folder, along with the Item naming, lifecycle definition and revision naming schemes. At the lowest level, each entity in a library will store information that links it to the corresponding named-revision of the Item in the vault that will receive and store the release data for it.

Once the preparation phase is complete the **Release Items** button becomes available. Clicking this button will present a confirmation dialog allowing you to review the release actions either at a summary level, or at a more detailed level.



Review and confirm the actions that will be taken by the release process.

Click **OK** to proceed with the release. After successful release, information regarding the Item that a symbol/model is mapped to is displayed in that symbol's/model's associated properties dialog. Click the **Show in Explorer** button to browse the released symbol/model directly in the vault using the **Vault Explorer** panel.



Example showing information about a symbol's mapped Item in the vault. Access the Vault Explorer panel to browse the released Item directly within the vault.

Creating and Releasing Components to a Vault

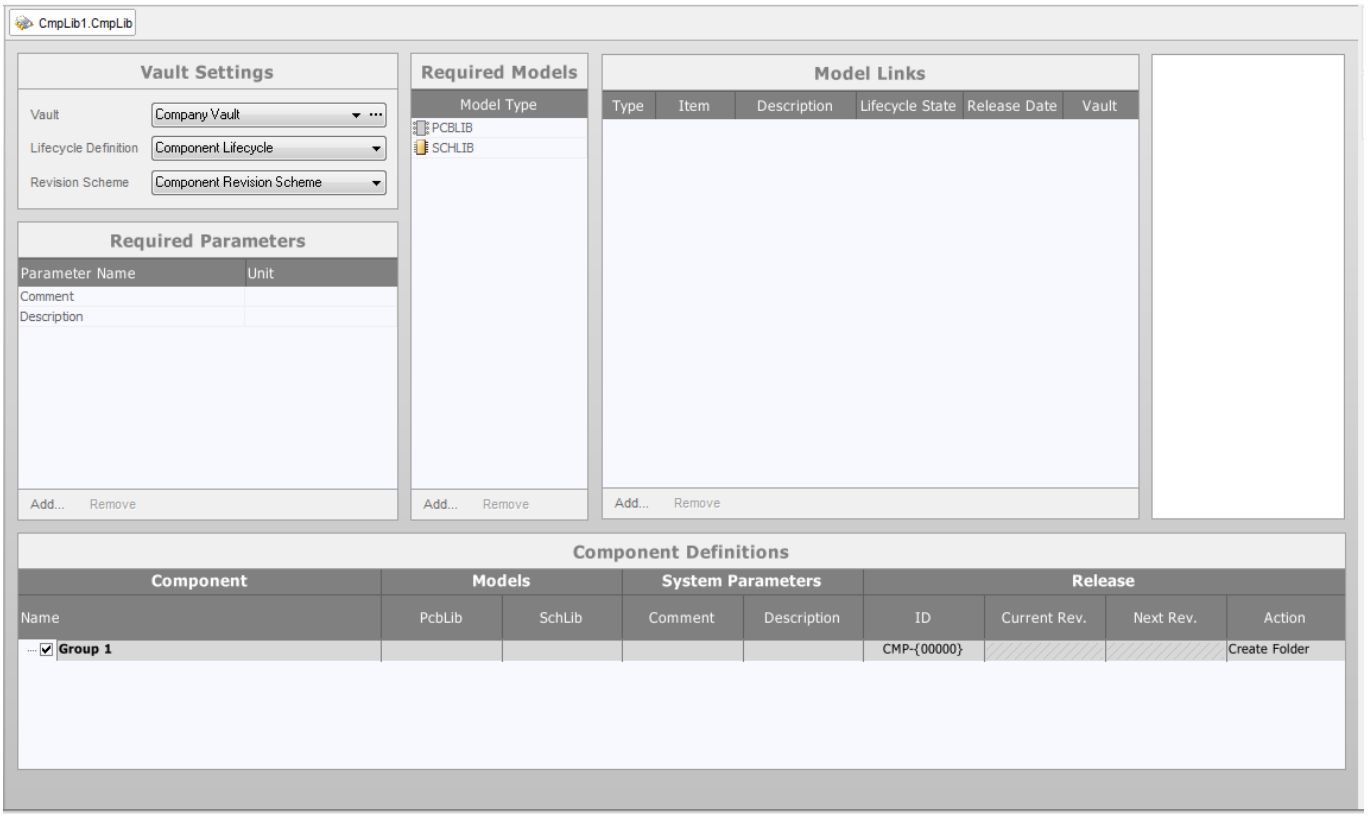
On the design side, each design component released to an Altium Vault is specified using a source *Component Definition*. A component definition is simply just that – a definition of a particular design component. A definition that ties together the required models and parameters for that component in a clean and ordered fashion.

A Component Library file (`*.CmpLib`) is the design-side entity in which to create and manage one or more component definitions. Each component definition in a Component Library 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 vault-based components which you can re-use again and again in your designs.

Preparing the Component Definitions

A new Component Library file can be created using the **File»New»Library»Component Library** command. A single Component Library file can be used to create (and therefore map to) one or more unique Component Items in a target vault, by entering one or more component definitions. Each component definition will have a common set of parameters and links to required domain models.

> In general it is good practice to have one component definition per Component Library but there are exceptions where it makes sense to manage components as a set, such as a set of chip resistors for example. Component Libraries provide for hierarchical factorization of models so when there are large sets of components that share symbols or footprints then sharing these in a single Component Library can facilitate a higher level of data integrity. If a footprint is changed for example, all of the 0603 chip resistors can be easily updated to use the new revision of that footprint, without the risk of missing one.



Create and manage design-side component definitions within a Component Library file.

Definition is simply a case of:

- Specifying the target vault in which to release.
- Specifying the required models (typically schematic symbol and 2D/3D component model for a board design component).
- Adding model links – browse the required vault for the named revision of a Schematic Symbol and PCB Component Items to be made available to the component definitions within the library.
- Define a set of parameters as required – these are applied to all component definitions in the library.
- Add the definitions themselves. Each definition will have fields with which to specify the domain models it should use, values for any parameters, and a unique ID for the Component Item that will be created. A default naming scheme is provided.

The document supports the creation of a hierarchy of component definitions, by allowing you to gather definitions into groups. The top-level group in this hierarchy can be mapped to a folder in the target vault, enabling you to specify an existing location in which to store released component data.

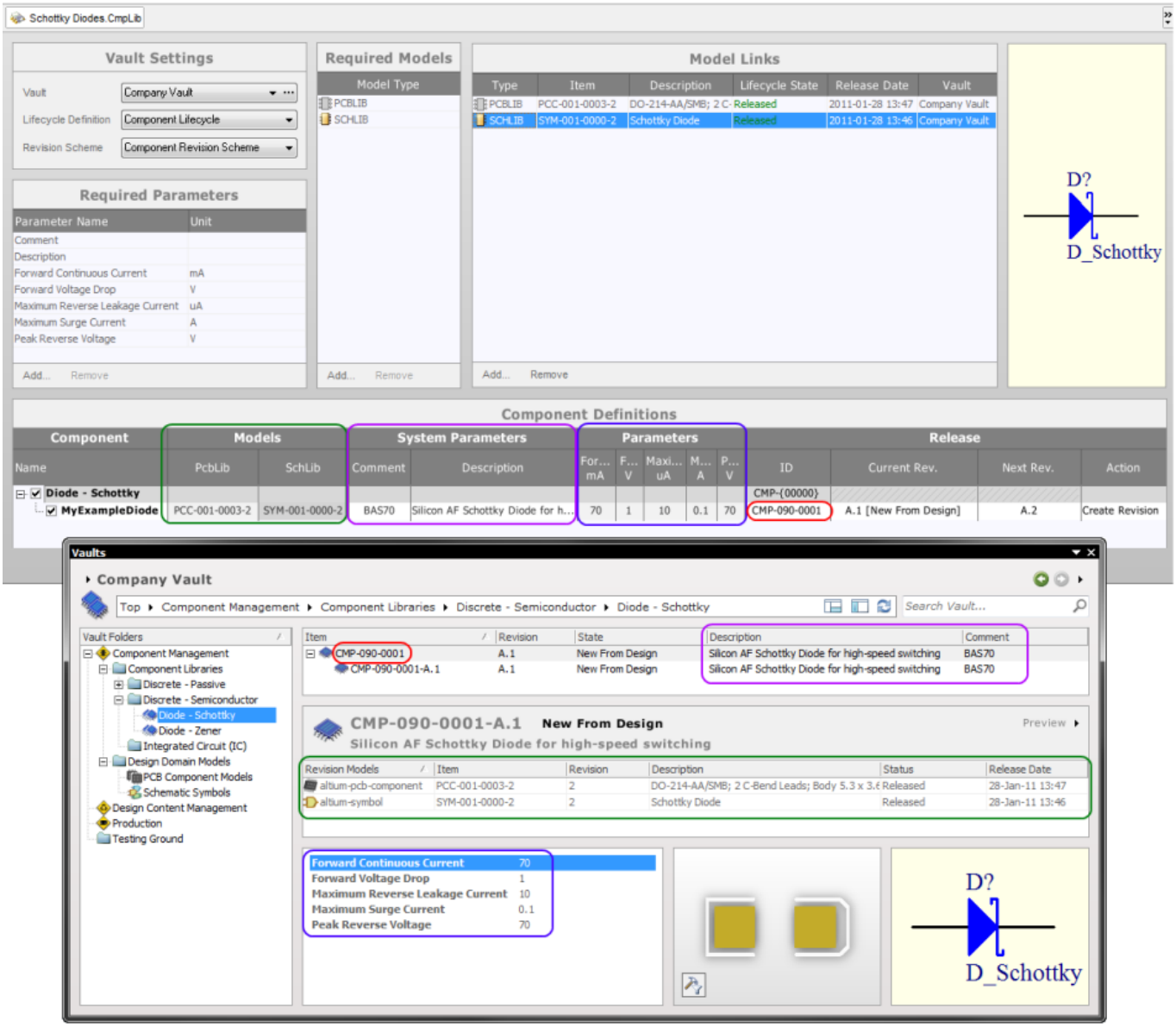
Releasing the Active Component Library File

Having defined the component definition(s) as required in the Component Library file, you can now proceed to release that file. The release process is simply the act of generating a new revision of each targeted Item.

To release the enabled component definition(s), simply use the **File»Release To Vault** command. A confirmation dialog will appear asking you to confirm that you want to release to the chosen target vault. Click **Yes** – the release process will proceed, in batch-fashion, with release data stored for each subsequent component definition in the next (planned) revision of the corresponding target Item in the vault. Where a definition is being released for the first time, a new Component Item will be created in the target folder within the vault, with the specified ID, and linked back to that definition.

The groups specified in the Component Library file link to folders in the target vault. If a vault folder exists with the same name as a specified group – in relation to where the folder hierarchy is to reside in the vault – then it will be used (linked back to that group). If such a folder does not exist, it will be created and linked back to that group.

Once released, you can browse the component(s) directly in the vault using the **Vault Explorer** panel. Jump directly to a Component Item in the vault by right-clicking on its component definition and choosing the **Show in Explorer** command.



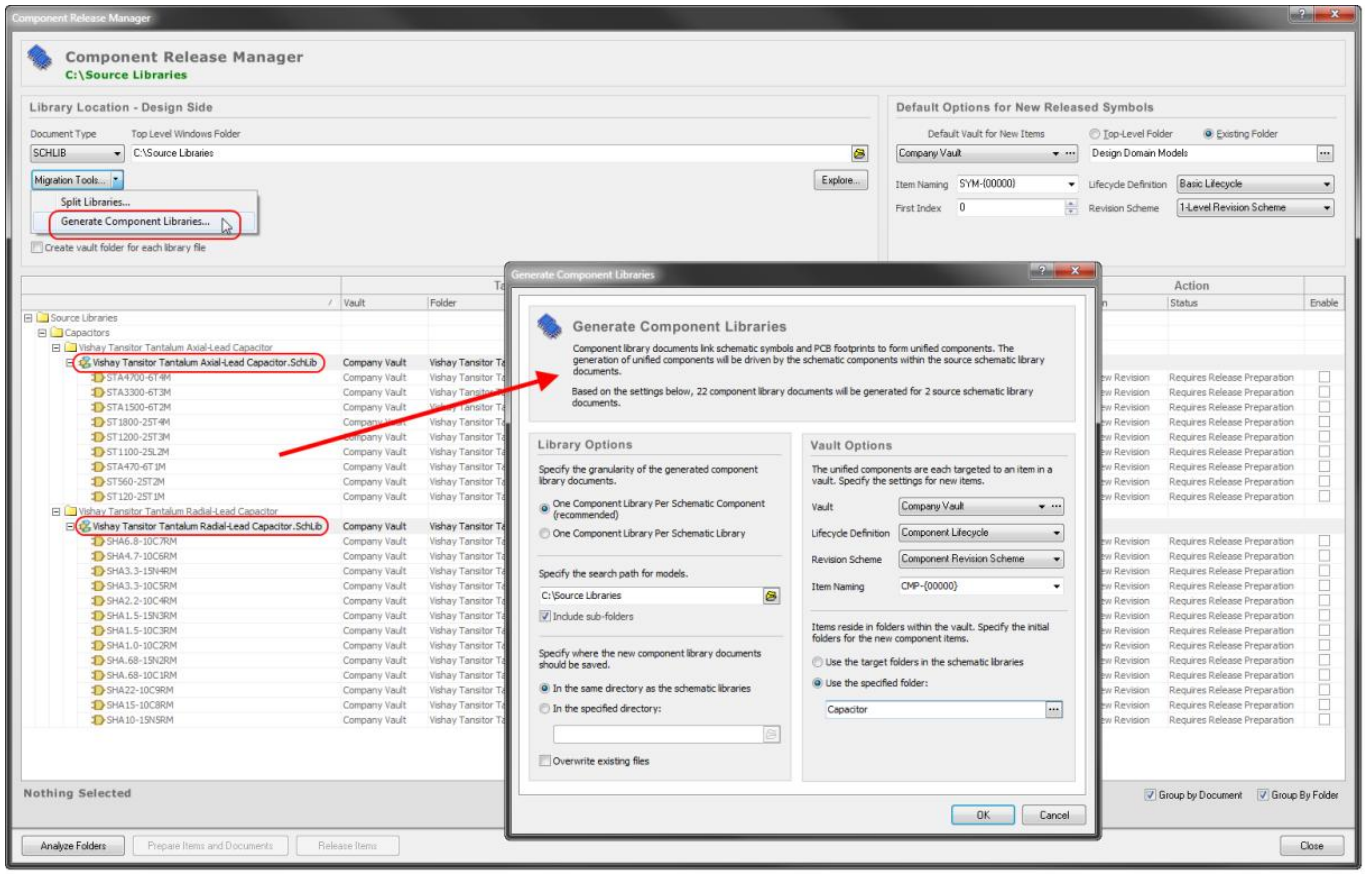
A released Component Item viewed directly from within the target vault, courtesy of the Vault Explorer panel.

Auto-Generating Component Libraries from Existing Schematic Libraries

Migration to another way of doing things. Slow or fast-tracked, we have all done it at some stage. Remember when you migrated to the use of Integrated Libraries, or Database Libraries? For the latter, a set of dedicated tools helped that migration no-end. Just as with migration to earlier component management methodologies, Altium Designer supplies you with various tools to aid in the migration to the next-generation of component management – vault-based components.

One particularly useful tool in the context of Component Libraries is the ability to generate these automatically from source Schematic Libraries. Performed from within the *Component Release Manager* (**File»Component Release Manager**), this tool takes the manual labor and potential for error out of the equation when creating the source component definitions from which the vault-based Component Items are generated.

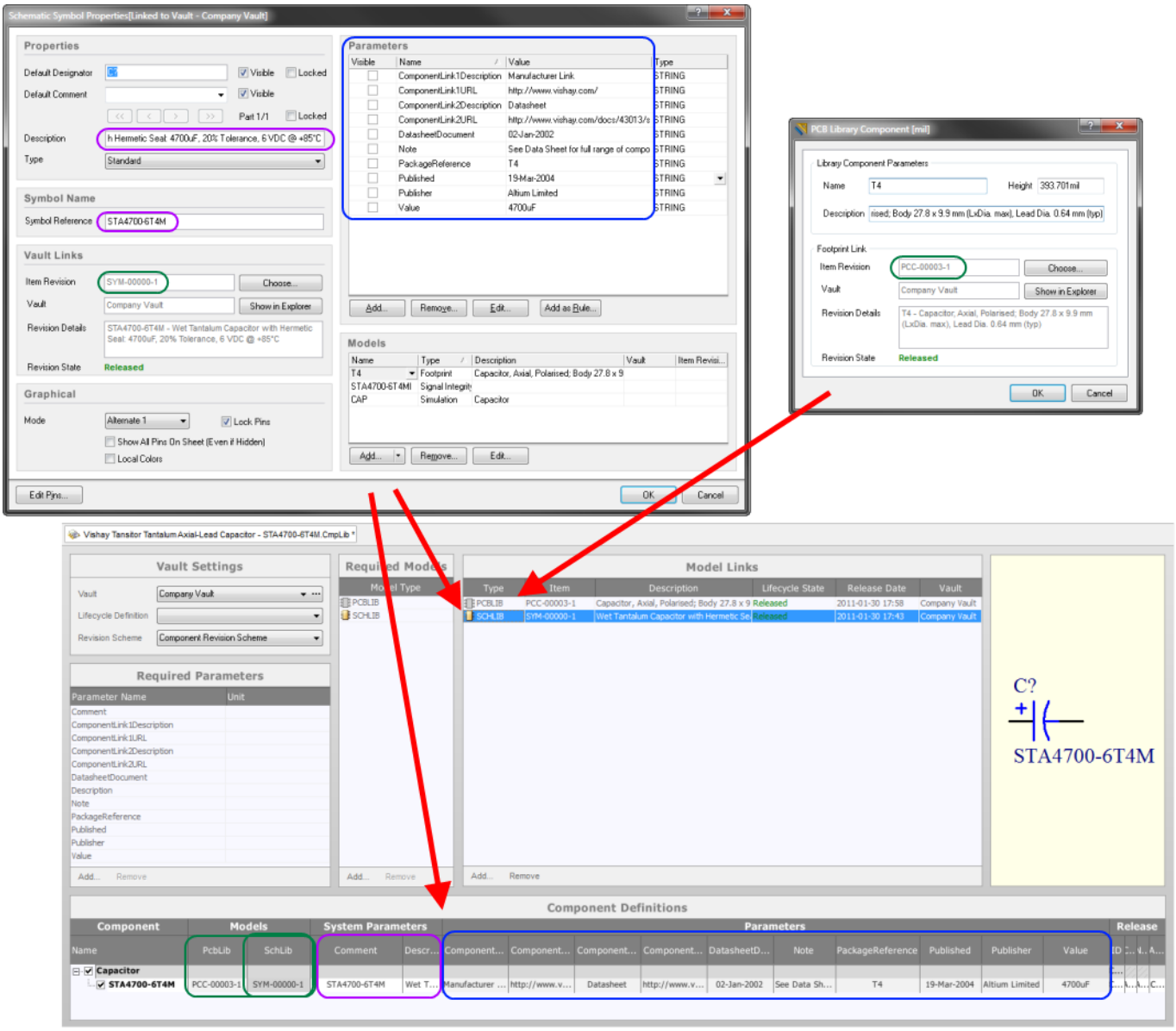
> Generation of Component Libraries in this way first requires that the domain model Items it needs to reference are themselves released first. The source symbol and associated PCB 2D/3D Component model for an original source schematic component must therefore be released to the target vault, before a new component definition can be created from that schematic component.



Automate generation of Component Libraries with the Generate Component Libraries migration tool.

The effort you have already spent in building a collection of components in you source Schematic Libraries is not ignored or discarded and you are not required to duplicate that effort! The migration tool uses the information for each original source schematic component in such a library to essentially fill-in the required fields of the component definition:

- The Items created from the release of schematic symbol and linked PCB 2D/3D Component model are used as the model links for the definition.
- The Symbol Reference and Description parameter values are used as the values for the Comment and Description parameters in the definition.
- All user parameters defined for the schematic component are added as required parameters for the definition.

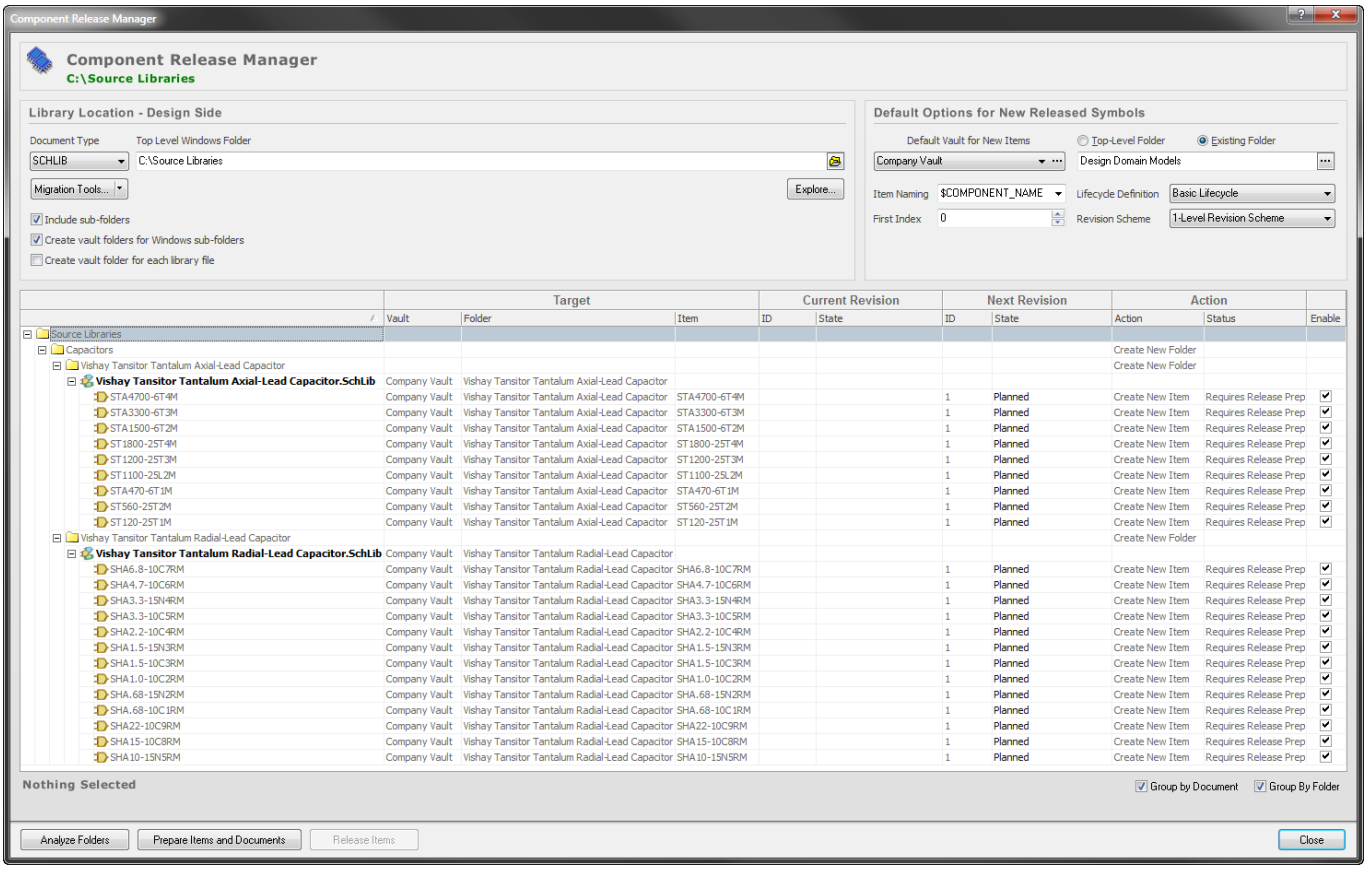


Example of a new Component Library and constituent component definition, generated from an original schematic component in a source schematic library.

Batch Release using the Component Release Manager

The *Component Release Manager* provides a centralized release 'console' with which to release, in-turn, schematic symbols, PCB 2D/3D component models, or component definitions, stored within one or more Schematic, PCB or Component Libraries respectively – facilitating a batch-style release of libraries to a target vault.

The *Component Release Manager* is accessible from any editor within Altium Designer, by running the **File»Component Release Manager** command.

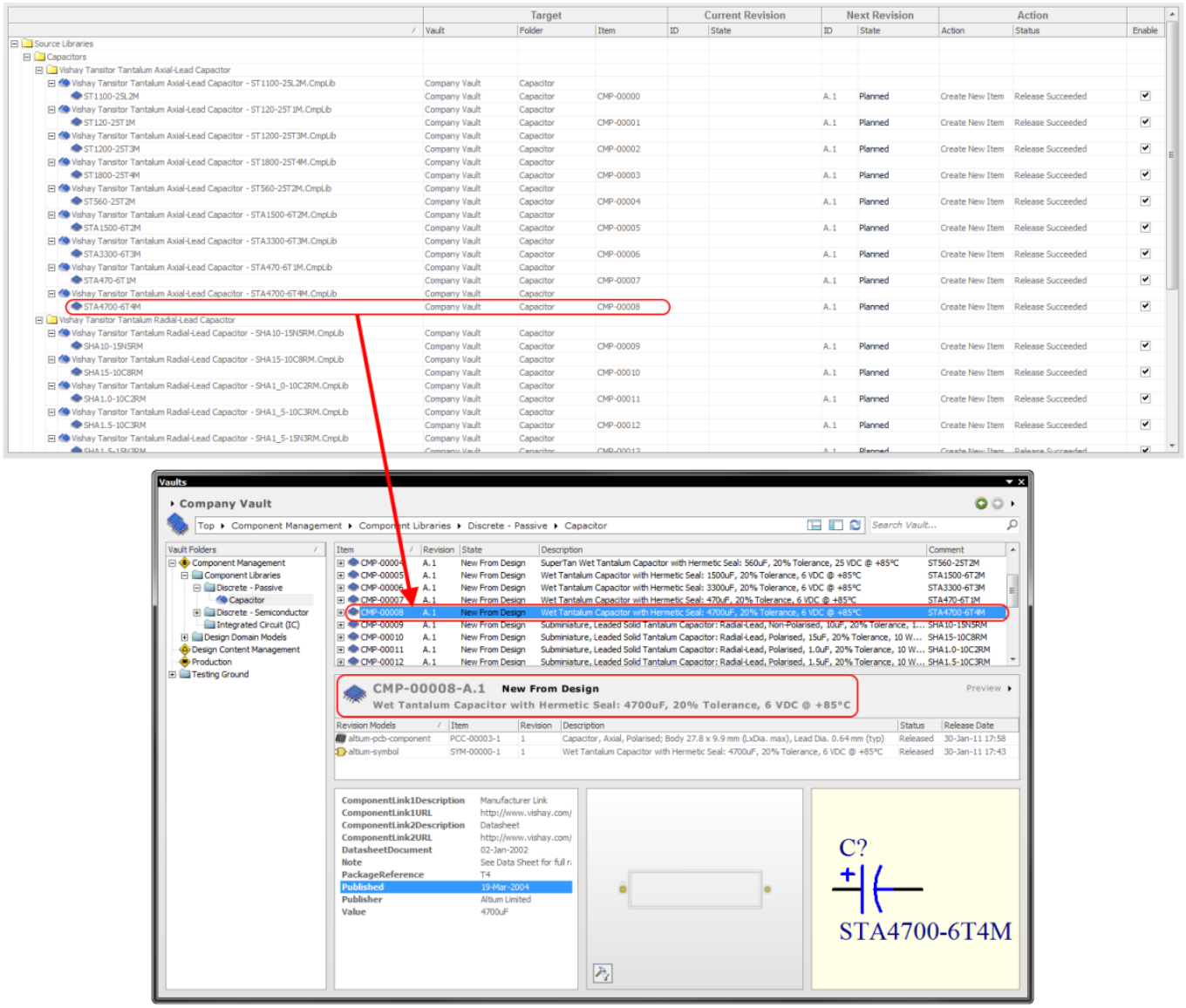


Example of the Component Release Manager in action!

Use of the *Component Release Manager* involves:

- Specifying a Windows folder where the source libraries (`*.SchLibs`, `*.PcbLibs` or `*.CmpLibs`) can be found.
- Specifying what type of source to release – either `SCHLIB`, `PCBLIB`, or `CMPLIB`.
- Specify vault-side options – where newly created Items should be stored, how they should be identified and the schemes used for naming their revisions and lifecycle states.
- Clicking the **Analyze Folders** button – the software scans, or analyzes both the source library folder(s) and the target folder(s) in the vault – to see how it needs to handle that content with respect to releasing and in accordance with the specified options (for example whether to create new Items and/or to re-release to new revisions of existing linked Items).
- Previewing link information and release actions. The main grid presents all information relating to the link between a source symbol, PCB 2D/3D component model, or component definition and its vault-based Item, as well as the actions that will be taken if the source entity is included in the release process.
- Clicking the **Prepare Items and Documents** button. On the vault side, any new folders that need to be created, will be created – in readiness to accept any new Items. Items themselves will be created, with initial revisions in the `Planned` state. If re-releasing to an existing Item, the next planned revision will be created. On the document side, the source libraries are saved with the link information. This is the link to the target vault and folder, along with the Item naming, lifecycle definition and revision naming schemes. At the lowest level, each entity in a library will store information that links it to the corresponding named-revision of the Item in the vault that will receive and store the release data for it.
- Clicking the **Release Items** button – to proceed with the release.

After release, use the **Vault Explorer** panel to browse the released Items.



Released Component Items - vault-based components - in the vault.

Part Choices

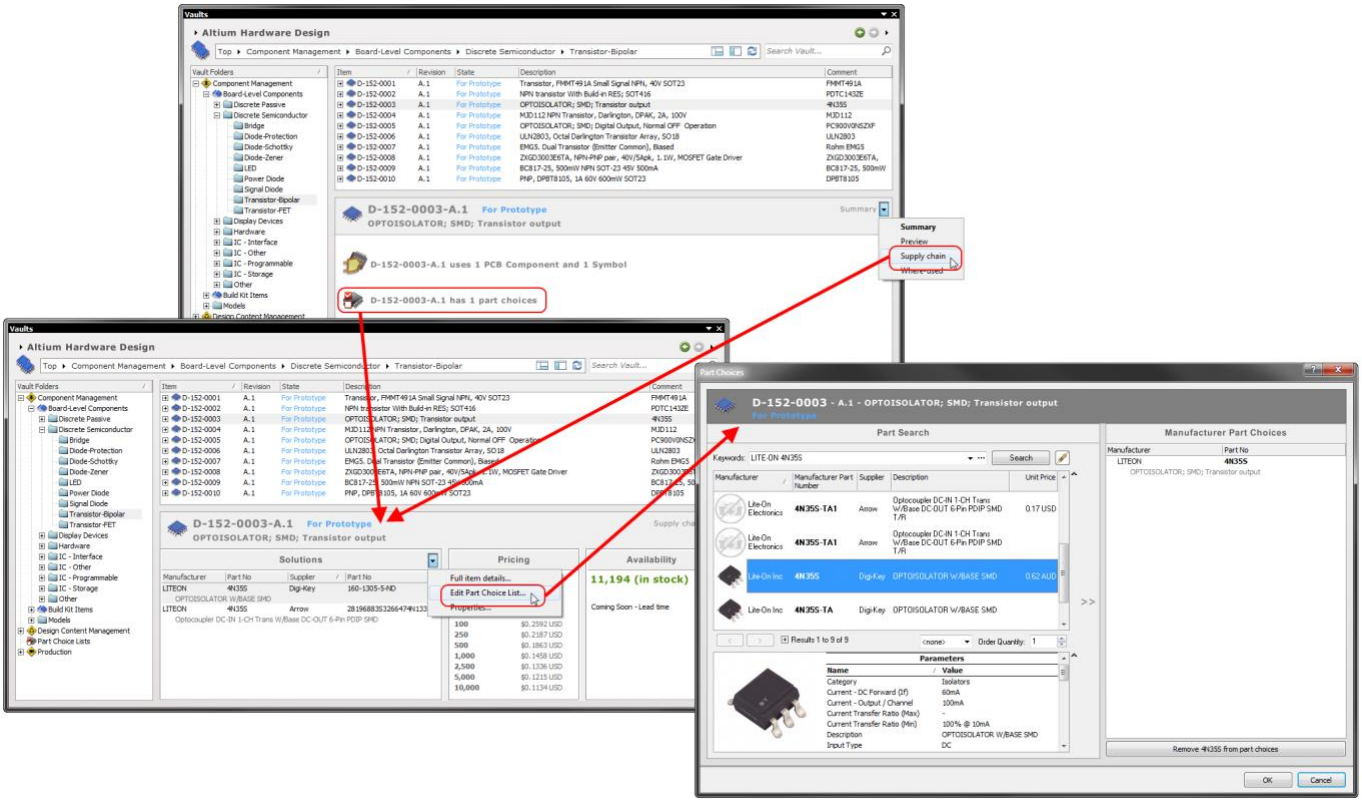
Releasing a component definition results in creation of a Component Item in the target vault, or 'vault-based component'. But at this stage, it is still simply the design (or engineering) view of the component. It means the world to the designer using it in a board design, but is not meaningful outside of the design arena. To become a truly Unified Component, that unites the design and supply chain camps, the Component Item must be mapped to physical, real-world manufactured parts.

Under the Unified Component model, 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 Item. 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 Global 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.

This ultimately creates 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 (as defined in the Global Part Catalog).

In the **Vault Explorer** panel, access to supply chain information for a Component Item is available from the **Supply Chain** view for that item. From the **Solutions** region of the view you can gain access to the *Part Choices* dialog – your interface to the Part Choice List Item. Simply search for a required Manufacturer Part and add it to the list. Remove existing parts from the list if no

longer a viable option in accordance with changes to design requirements – the choice of parts is down to you! When you save your choices a new Part Choice List Item will be created and linked to that Component Item or, if the Part Choice List Item exists already, it will be saved as a new revision.

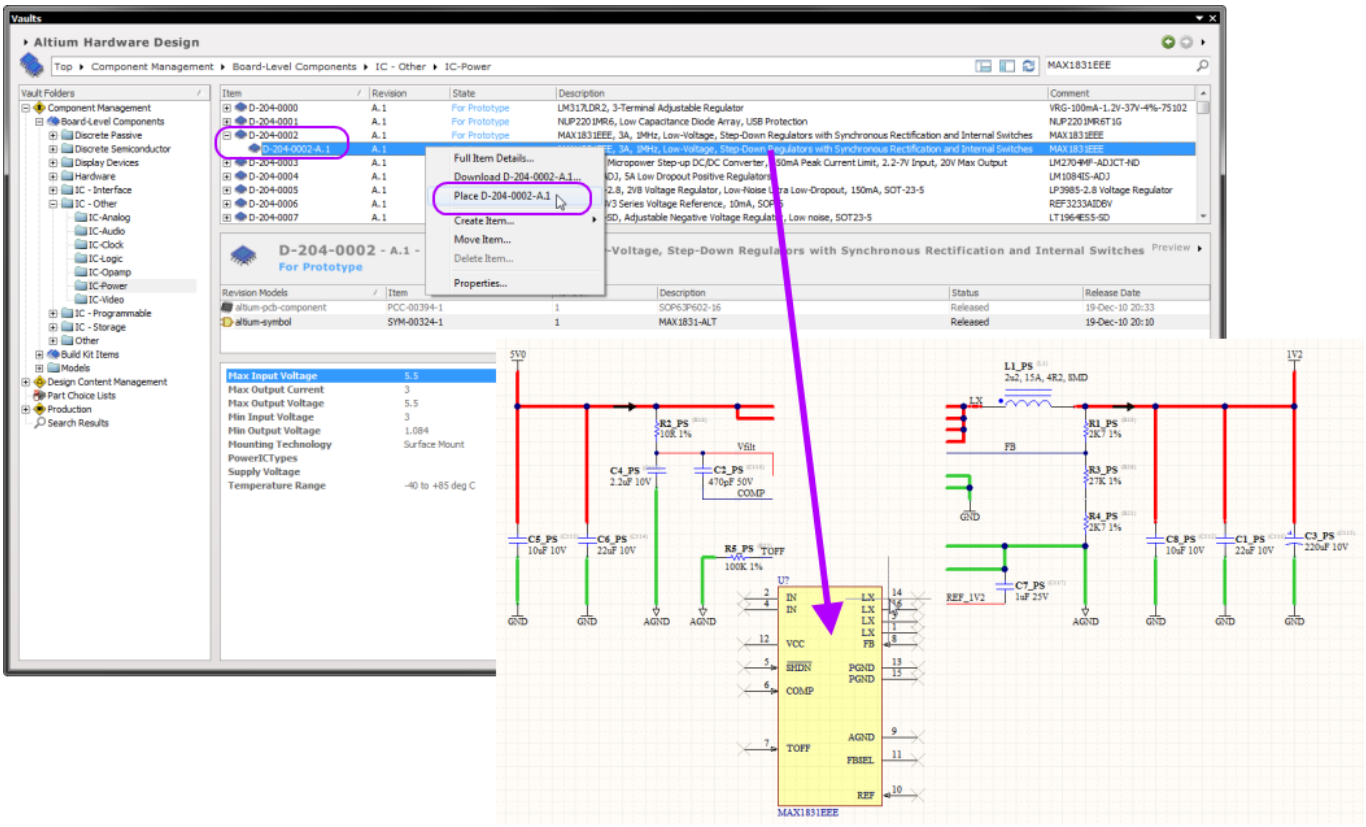


Search for and choose the real-world manufactured parts that can be used to implement a Component Item during assembly. For all Vendors linked to the chosen Manufacturer Parts, real-time pricing and availability information is fed back, directly from those Vendors.

Placing Vault-Based Components into a Board Design

Placing a vault-based component truly is simplicity itself. But before you do anything, first ensure that the schematic sheet that is to receive the component is open in Altium Designer and is the active document.

Now, pop open the **Vault Explorer** panel, browse or search for the Component Item you wish to place, then right-click on the specific revision of that component required (typically the latest, in which case just right-click directly on the top-level Item entry) and choose the **Place** command. As with placement from other library types via the **Libraries** panel, the component will float attached to the cursor – just pick a ball-park spot on the active schematic sheet and click to effect placement. You can fine tune and nudge it into its final location at a later stage!



Place a vault-based component in a matter of clicks!

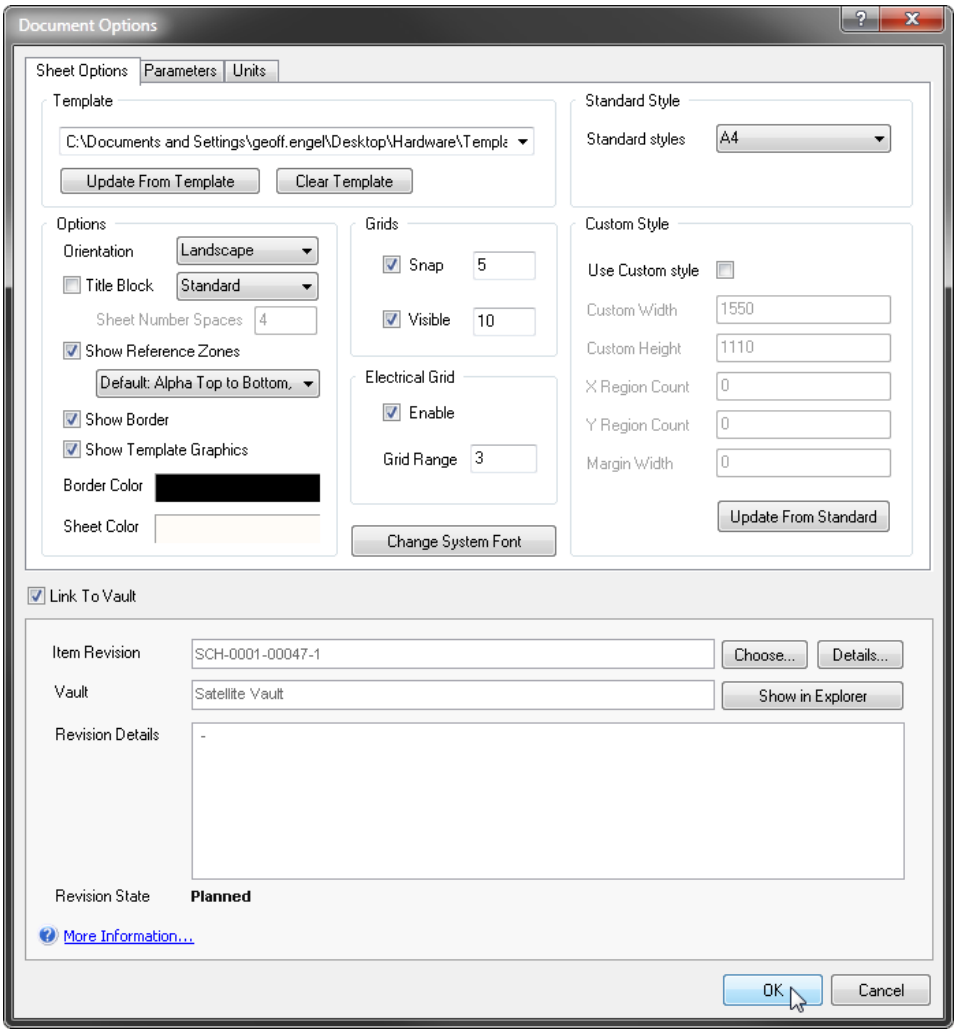
Managed Schematic Sheets

Moving up from the individual component to the 'chunk of design' level, Altium Designer delivers design reuse through what is called a *Device Sheet*. A Device Sheet is actually a standard Altium Designer schematic sheet, it is how it is stored and used in Altium Designer that distinguishes it from a regular schematic sheet. The Device Sheet concept is not limited to a single schematic sheet either, you can place a Device Sheet in your design that is the top of a tree of other Device Sheets.

Like components, Device Sheets can also be stored in the controlled environment of an Altium Vault. When Device Sheets are stored in a vault, they are referred to as *Managed Sheets*. Like vault components, Managed Sheets are placed from the vault into the current design. Like vault components, they can be revised if needed, with full control over the revision process. And like vault components, their lifecycle can also be managed (ready for prototype, for production, and so on), reflecting their real 'usability' state.

Because Managed Sheets are stored in a vault, the components on them should also be stored in the vault. That way, you get the full benefit of the managed content system that the vault provides, including being able to identify and locate all the components used on the Managed Sheet (the Children), and also being able to identify and locate which designs the Managed Sheet has been used in (Where-used).

With vault-based components placed and wired as required on a schematic sheet, you simply link the sheet to a Schematic Sheet Item created in an Altium Vault (and in the Planned state), using the **File»Link Sheet to Vault** command. Linking is performed in the *Document Options* dialog.

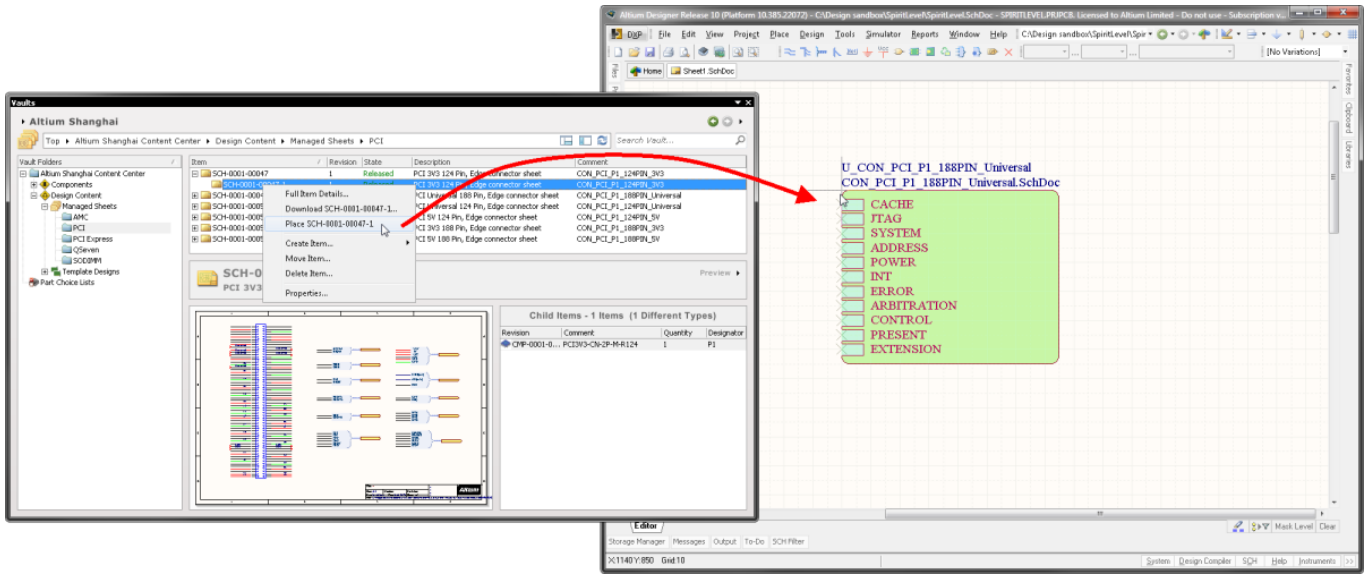


Link the active schematic sheet to the chosen planned revision of a Schematic Sheet Item in a vault.

Once linked, the **File»Release To <TargetVaultName> Vault** command will become available – click this to release the sheet to the target vault.

Placing a managed sheet is a straightforward process. But before you do anything, first ensure that the schematic sheet that is to receive the managed sheet's sheet symbol is open in Altium Designer and is the active document.

Now, pop open the **Vault Explorer** panel, browse or search for the Schematic Sheet Item you wish to place, then right-click on the specific revision of that sheet required (typically the latest, in which case just right-click directly on the top-level Item entry) and choose the **Place** command. As with placement of vault-based components, the managed sheet symbol will float attached to the cursor, ready for you to effect placement at the location required.



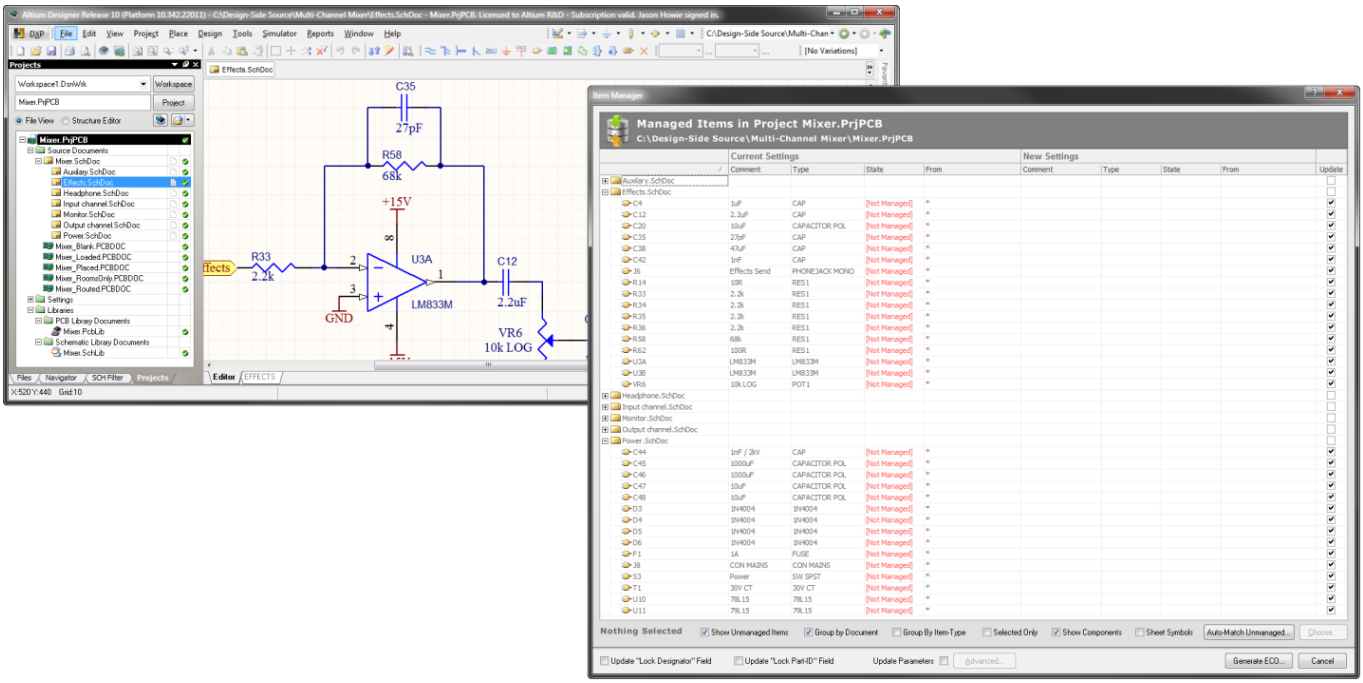
Managed sheets are placed from the Altium Vault in the same way a component is placed.

Replacing Existing Design Components with Vault-Based Ones

Getting your source components and device sheets migrated into a vault is one thing, but once you have your collection of Component Items (referencing various vault-based domain model Items) and Schematic Sheet Items, what then? Your existing board design projects may still be utilizing components that point to 'old world' source libraries. How to quickly update your design so that it uses the Items you have released into a vault?

The *Item Manager* fits the bill and provides the solution required – the final piece of the migratory puzzle as it were – by enabling you to essentially update *unmanaged* design items in a current board design project to use *managed* design Items that reside in one or more Altium Vaults.

Assuming you have already created collections of vault-based components and schematic sheets in a target Altium Vault – through release of the required SCHLIBs, PCBLIBs, CMPLIBs, and schematic sheets – simply open the board design project whose design entities you wish to 'convert'. Then, from an active schematic document in that project, access the *Item Manager* (**Tools»Item Manager**). The *Item Manager* presents all components and/or sheet symbols found in the project's set of schematics. For each entity, information about the current design item used in the sheet/design is given.

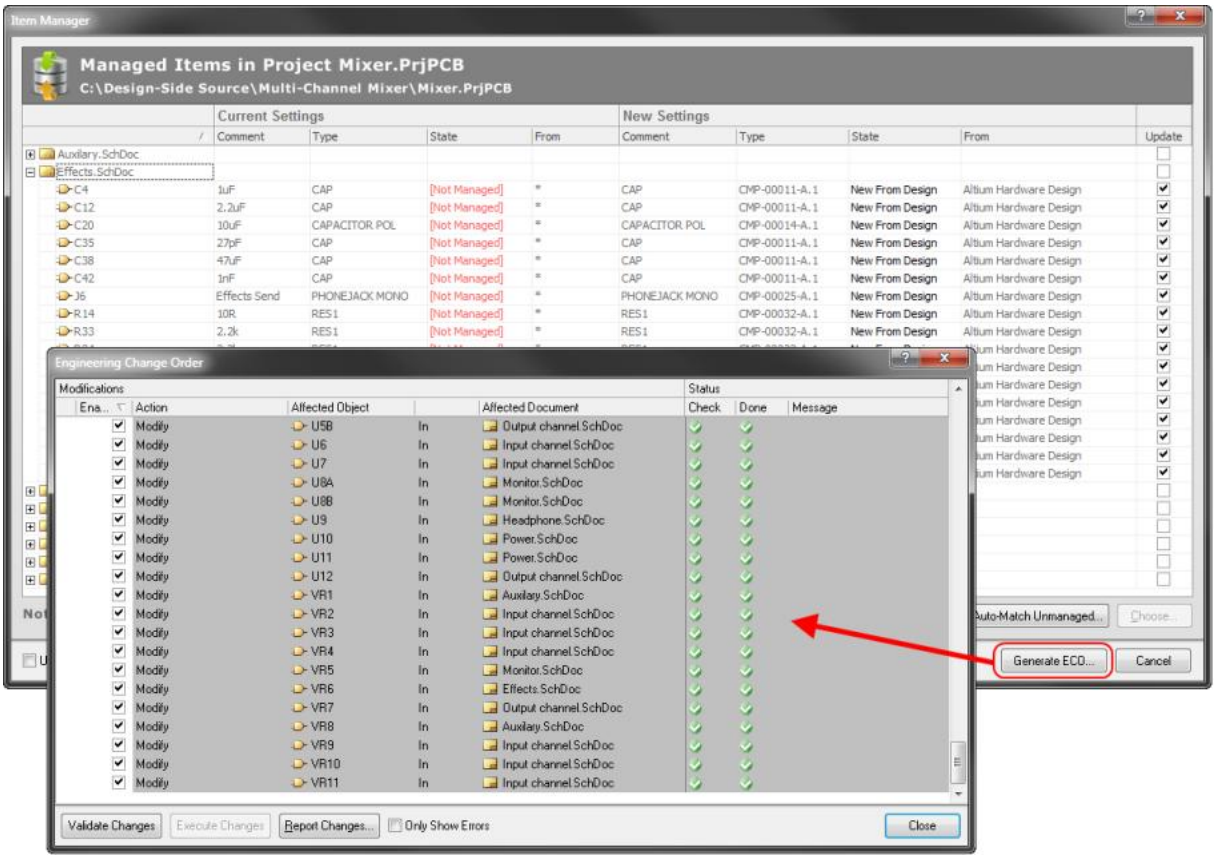


The Item Manager surveys the board design schematics and presents information on the components and/or sheet symbols currently being used.

Select an unmanaged item in the grid (or a group of items of the same type) and click the **Choose** button, accessing a dialog through which to browse within any of your currently connected vaults and select the required Component or Schematic Sheet Item, and specific revision of that Item, to be used in the update.

Alternatively, and in a far more streamlined and expedient fashion, make use of the *Item Manager's* auto-matching functionality – invoked by clicking the **Auto-Match Unmanaged** button. This process takes the unmanaged design items and attempts to match each one with a managed Item in a connected vault. The matching process takes the **Design Item ID** of an unmanaged item and compares it against the **Comment** property of managed Items in a vault. When a match is found, across any connected (and enabled) vault, that managed Item (Component Item or Schematic Sheet Item) will be proposed for the update.

You have full control over which of these design-side entities to update, and how. All proposed changes are reflected back in the manager. Once changes have been set up as required, they are effected using an Engineering Change Order (ECO).



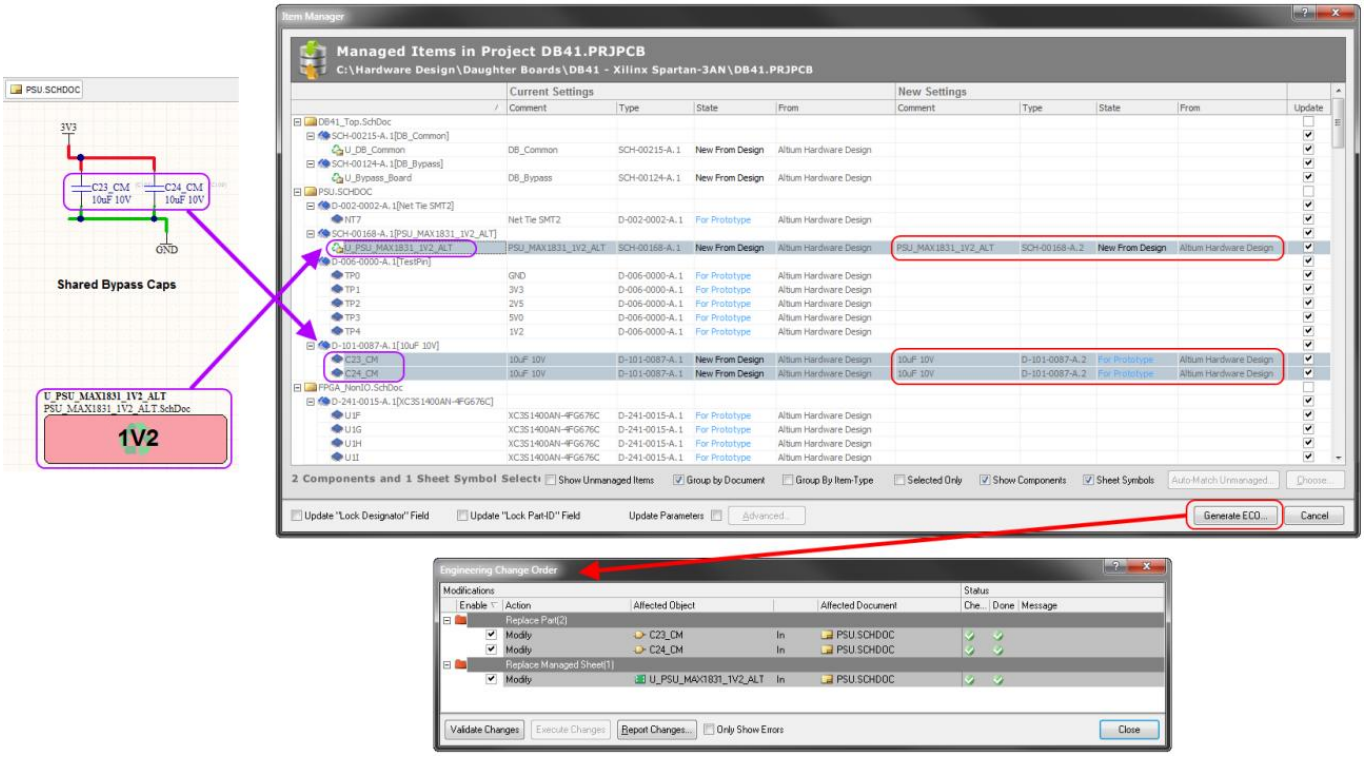
Changes are implemented through use of an Engineering Change Order (ECO).

Keeping your Design Components In-Sync with those in a Vault

Having initially placed managed design Items onto schematic sheets of a board design, or replaced unmanaged entities with managed ones, there now needs to be some way of updating those instances, with any changes to their linked source Items in the vault. In other words, a means by which to keep the design synchronized with the vault entities that it uses. The *Item Manager* steps in to offer a solution which is analogous to the 'Update From Libraries' feature – only this is more 'Update From Vault'!

You might have re-released a component definition in a source Component Library, to take into account a change to a referenced domain model (which will also have been re-released!). This creates a new, later revision of that Component Item in the vault. And that Component Item might be used as a child Item in a managed Schematic Sheet Item. To use this later revision of component requires that the source schematic sheet be updated to use that component and then re-released to obtain a later revision of that Schematic Sheet Item. Then, the board design in which both that Component Item and/or Schematic Sheet Item is used must be updated (if required) to now use the latest revisions of those Items.

Simply use the *Item Manager* to choose the required new revisions of these Items and perform an update as required, implemented through generation and execution of an ECO.



Synchronize the vault-based items placed on your schematics with their linked source vault Items using the powerful update capabilities of the Item Manager.

Accessing Supply Chain Info as you Design

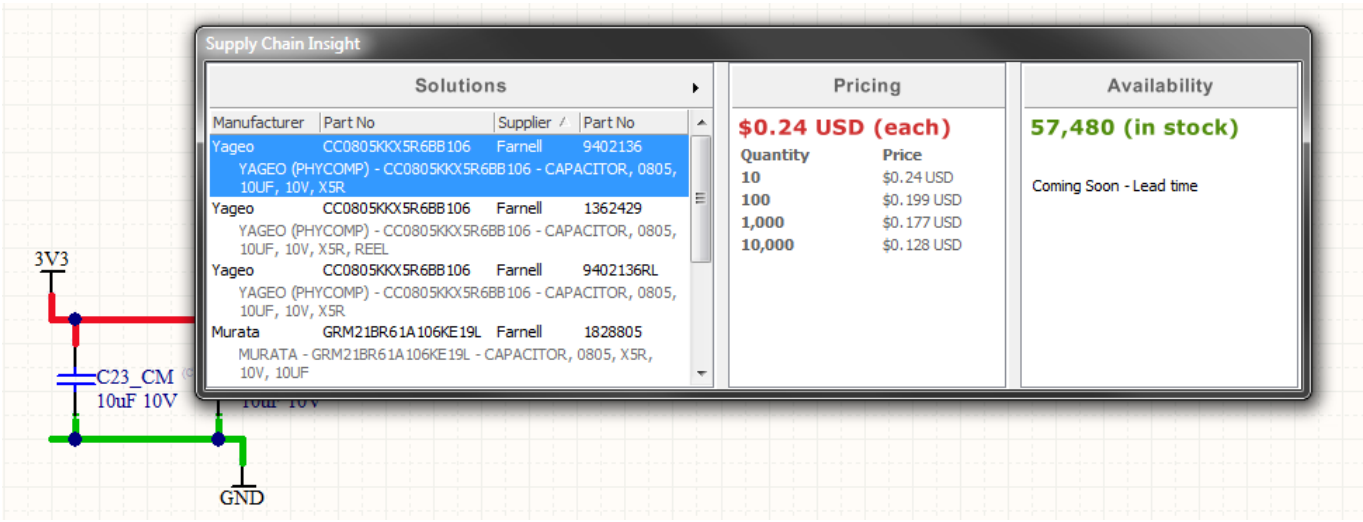
As a designer the ability to specify, at design time, which Manufacturer Parts can be used to implement your design components, gives you greater control and effectively streamlines the procurement process. You don't wait anxiously after design data are sent to manufacturing, wondering if the components procured for assembly will perfectly reflect your design intent. You are in the driving seat, with procurement able to see your choices at an early stage.

But sometimes even the best laid plans can go astray. What happens if you had a single part choice and that part becomes no longer available? Or maybe a price change to a component could cause blow-out costs during manufacturing and assembly. In this case, the ability to make design-time part choices loses its edge if you are unaware of such changes to cost and availability of the very parts you have 'authorized' to be used.

To retain the advantage that design-time part choices provides, Altium Designer delivers real-time display of cost and availability for chosen parts directly within the Schematic Editor as you design. This is facilitated by extending the software's *Design Insight* feature to include *Supply Chain Insight* – enabled and configured from the **System – Design Insight** page of the *Preferences* dialog (**DXP»Preferences**).

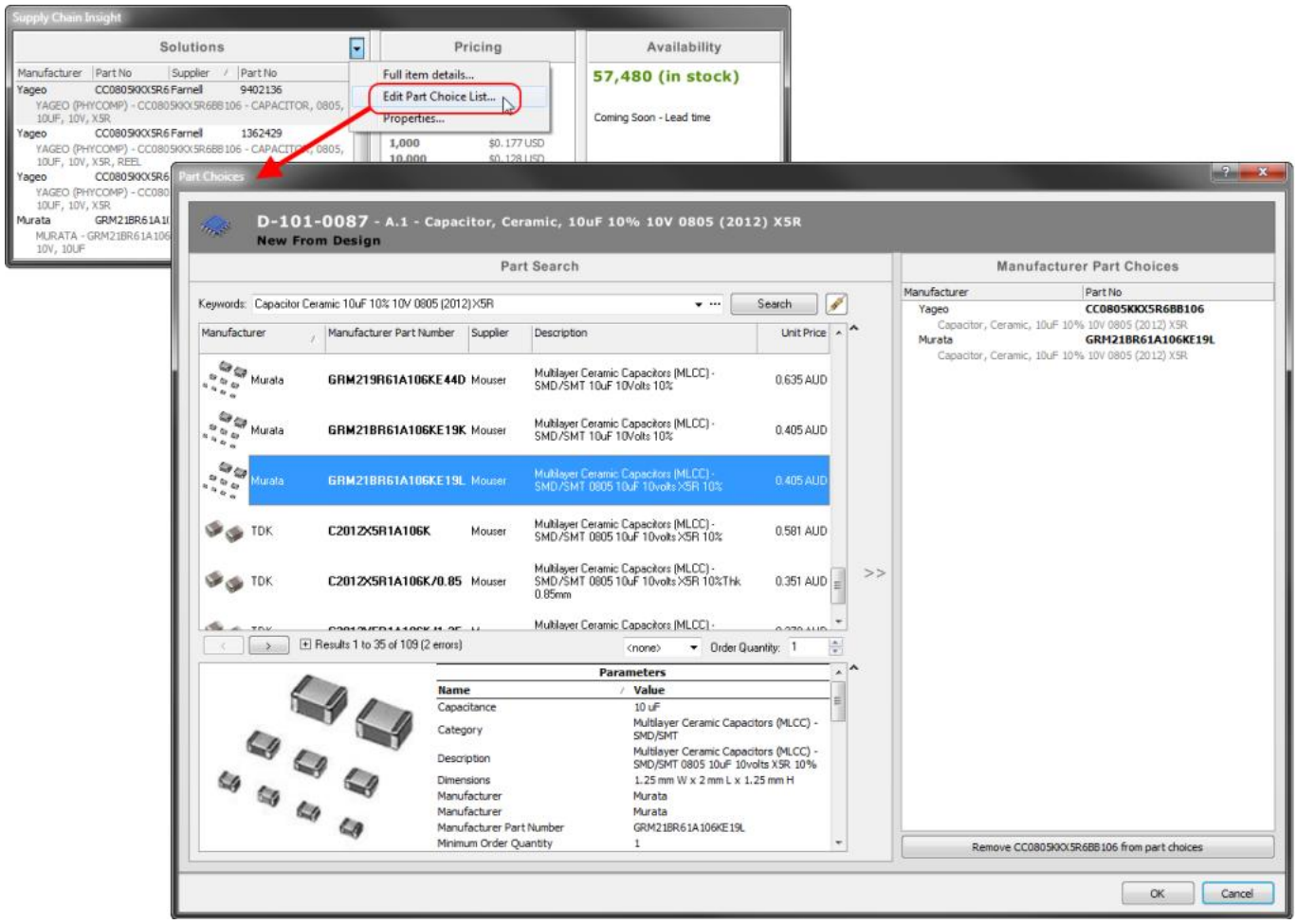
As you hover the mouse over a component on a schematic sheet, supply chain information will be displayed, provided that:

- The component has been placed from an Altium Vault – it is an instance of a named-revision of a Component Item in that vault, and
- The component has part choices made for it – at least one Manufacturer Part has been chosen in the component's associated Part Choice List Item.



Example supply chain information accessed by hovering over a component on the schematic sheet.

Suddenly seen that a chosen part is not available, or that its price no longer justifies its use? No problem. You can access the *Part Choices* dialog and make changes to the list of Manufacturer Parts as required. Choose a cheaper component, or one that is available – the choice is yours, and the choice can be made while you are still designing and *not* after the design has left your hands!

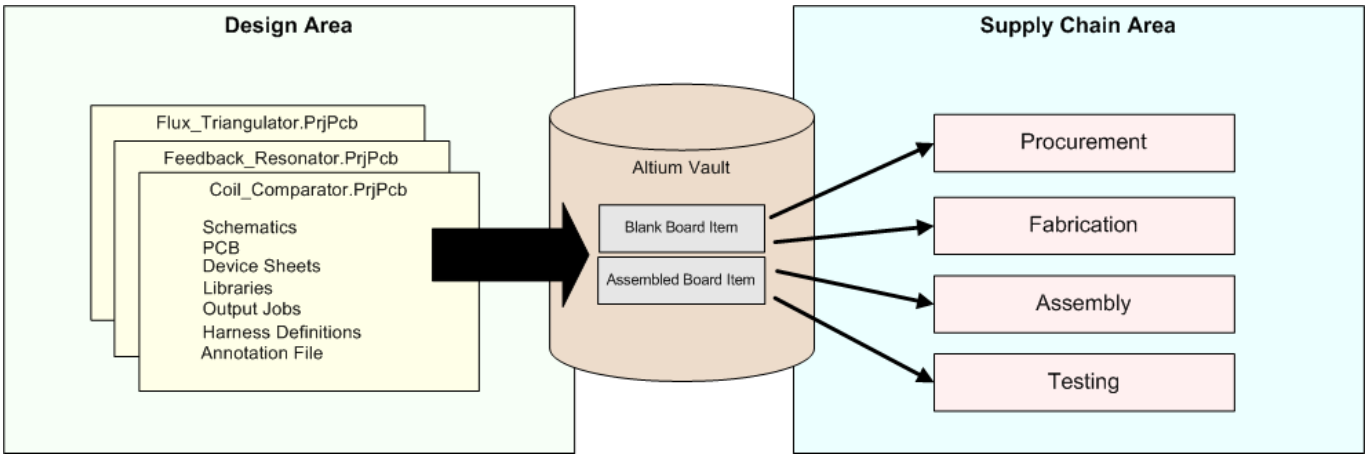


Not happy with current part choices? – Make changes as required on-the-fly as you design!

Releasing a Board Design to a Vault

With your board design ready, and featuring a myriad of managed components and perhaps also managed sheets – placed from one or more Altium Vaults – you now need to release that design itself into a target vault.

Altium Designer answers this call by providing powerful, high-integrity board design release management, as part of Altium's wider Design Data Management System. The board design release process is automated, enabling you to release your board design projects without the risks associated with manual release procedures. When a particular project is released, a snapshot of the design source is taken and archived along with any generated output. Release data are stored as revisions of a specific Item – the entity within a target Altium Vault, and which represents a tangible product that is made from that design project and sold by the company. This can be either a Blank Board (manufactured by the fabrication house), or an Assembled Board (the bare board populated with specified components, in accordance with a Bill of Materials).

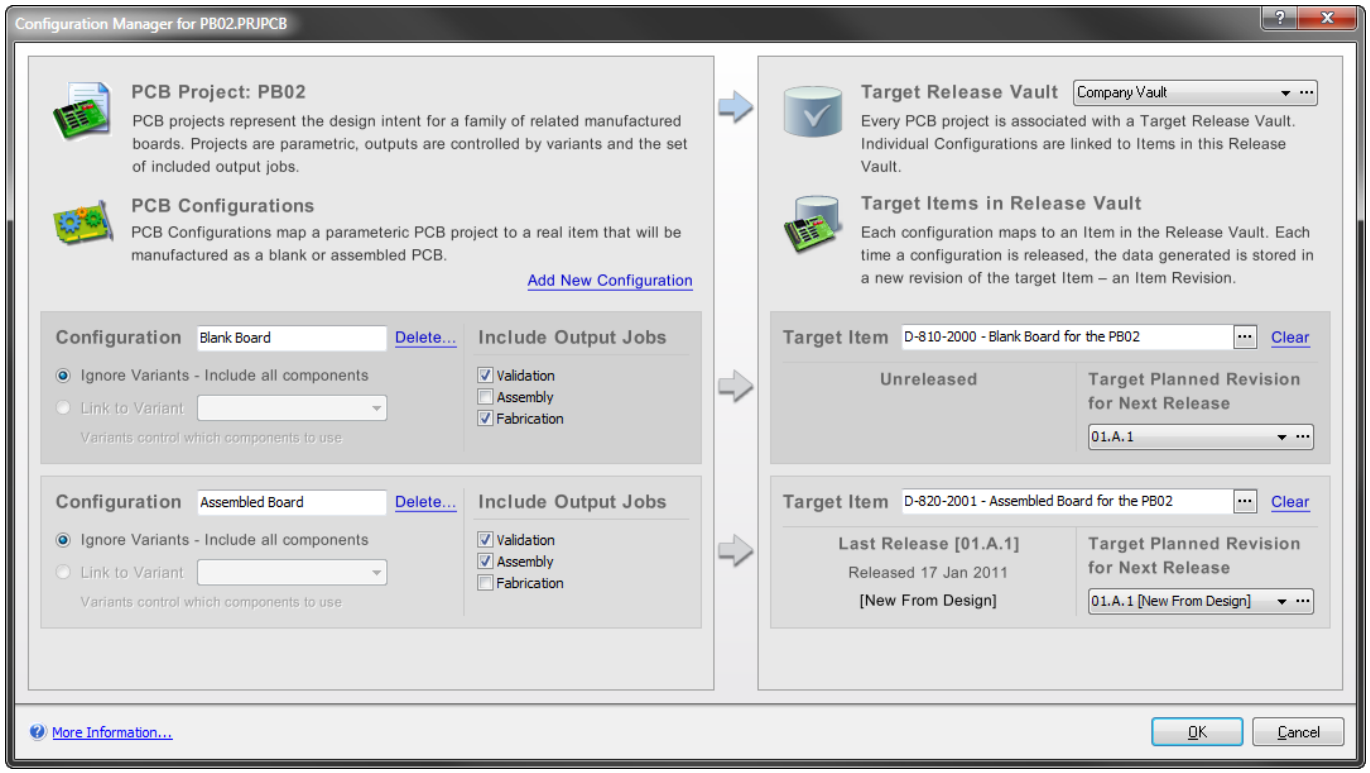


Generated data from a board design are securely stored in the vault within either a Blank Board Item or an Assembled Board Item. This high-integrity data are then used by the supply chain to build the specific item required.

Mapping a Design to a Target Item – Project Configurations

Providing a formal configuration structure, to map from source PCB project in a Design Repository to a specific Item in an Altium Vault, Altium Designer employs the concept of *PCB Project Configurations*. Each configuration of a PCB project on the design side maps to a specific Item on the manufacturing side. As its output is simply a generated set of instructions, or 'blueprint' for how to make the Item, a configuration simply needs to define which variant and associated project Output Job files to be used, and nominate which Item – in the chosen Altium Vault – to target.

All definition and management of configurations for a PCB project is performed from within the *Configuration Manager* dialog (**Project»Configuration Manager**). The *Configuration Manager* dialog is divided into configurations on the left, that map to target Items on the right. Each unique configuration must map to a unique Item in the target vault.



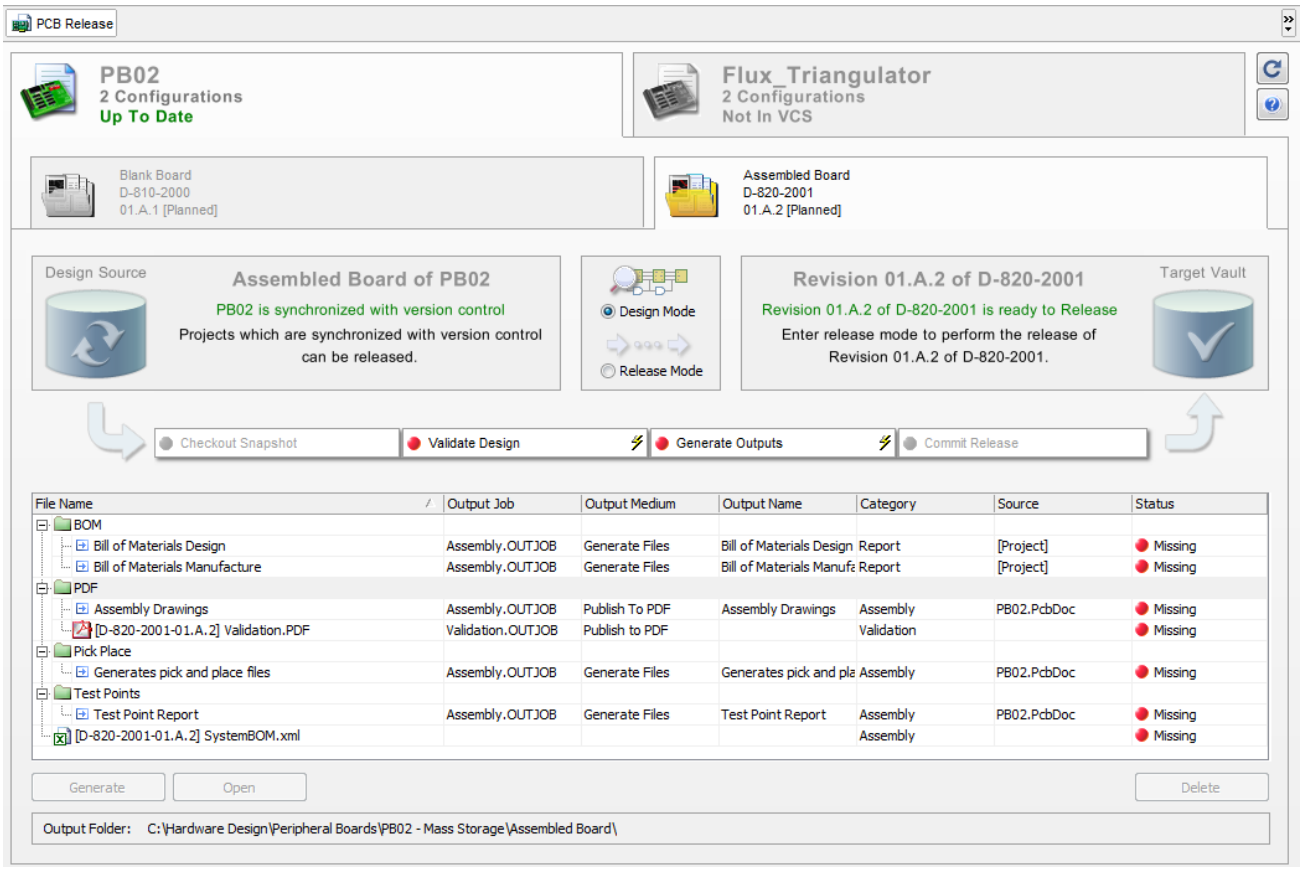
The Configuration Manager streamlines the process of defining configurations for the active PCB project.

The beauty of the *Configuration Manager* lies in its intuitive simplicity – enabling you to define the configurations you require with streamlined efficiency. For each configuration, simply specify a particular variant to be used (if applicable and/or required), assign which Output Job file(s) to be used, and browse the nominated Altium Vault to choose the Item to target (or map) the configuration to. Don't forget to give each configuration a meaningful name, so that you can easily distinguish the purpose of each!

The Board Design Release Process

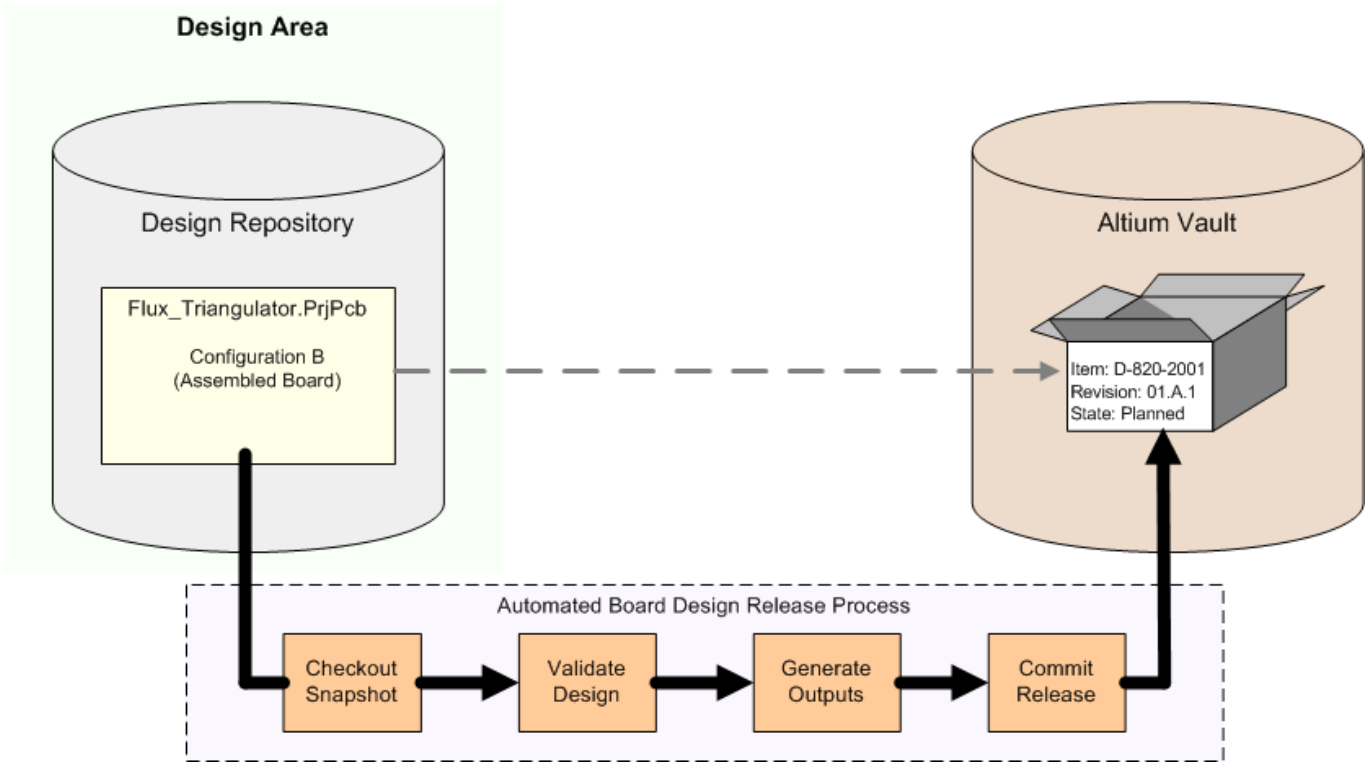
The **PCB Release** view (**View»PCB Release View**) is the graphical interface to Altium Designer's *PCB Process Manager* – used to perform the board design release process – and presents a high-level 'Dashboard' that operates in two modes:

- *Design Mode* – for controlling and managing the Board-Level design process. Here you can run validations and generate outputs as needed, and in any order, 'testing the waters' as it were to ensure all is as it should be prior to initiating the actual release of the intended configuration.
- *Release Mode* – for initiating releases of PCB Project configurations. This is where the high-integrity release process is performed, taking the specified configuration of the design project and processing it to obtain the data required to physically produce the revision of the Item referenced by it.



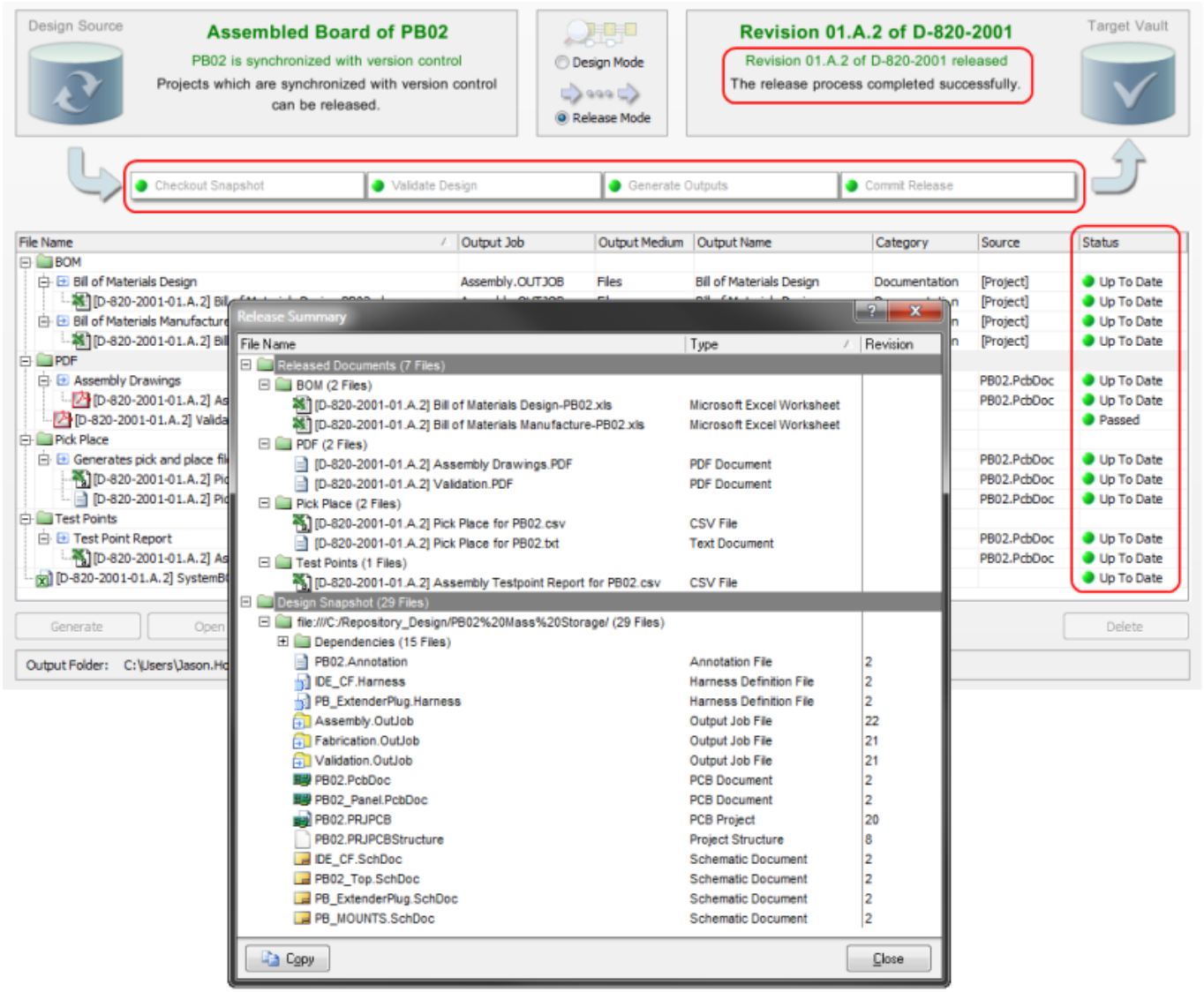
An example of the PCB Release view.

With the **PCB Release** view in `Release Mode`, initiating the release is an automated affair, kicked off by pressing a single button. The process consists of several stages run in sequence – a process flow if you like. The following image illustrates the stages involved in the overall flow, which are discussed in more detail thereafter.



The various stages that constitute the automated board design release process.

- **Checkout Snapshot** – a snapshot of the data, including dependencies, is checked out from the Design Repository.
- **Validate Design** – all defined validation output generators, defined in an Output Job file assigned to the configuration being released, are run. This includes running any of: Differences Report (using the comparator to determine if the source and PCB design documents are correctly in-sync); Electrical Rules Check (checking the electrical/drafting validity of the captured source design); Design Rules Check (checking the validity of the PCB document in relation to specified board-level design constraints); Footprint Comparison Report (comparing footprints on the board against their source library to ensure they are up-to-date, and matched).
- **Generate Outputs** – all other defined outputs in the assigned Output Job file(s) are run. These are the outputs that drive the release of the target Item, the instructions from which the physical Item will be produced to exist as a tangible product which can be bought and sold.
- **Commit Release** – pushing the generated outputs and validated design document snapshot into the defined new revision (planned revision) of the target Item, stored in the nominated vault.

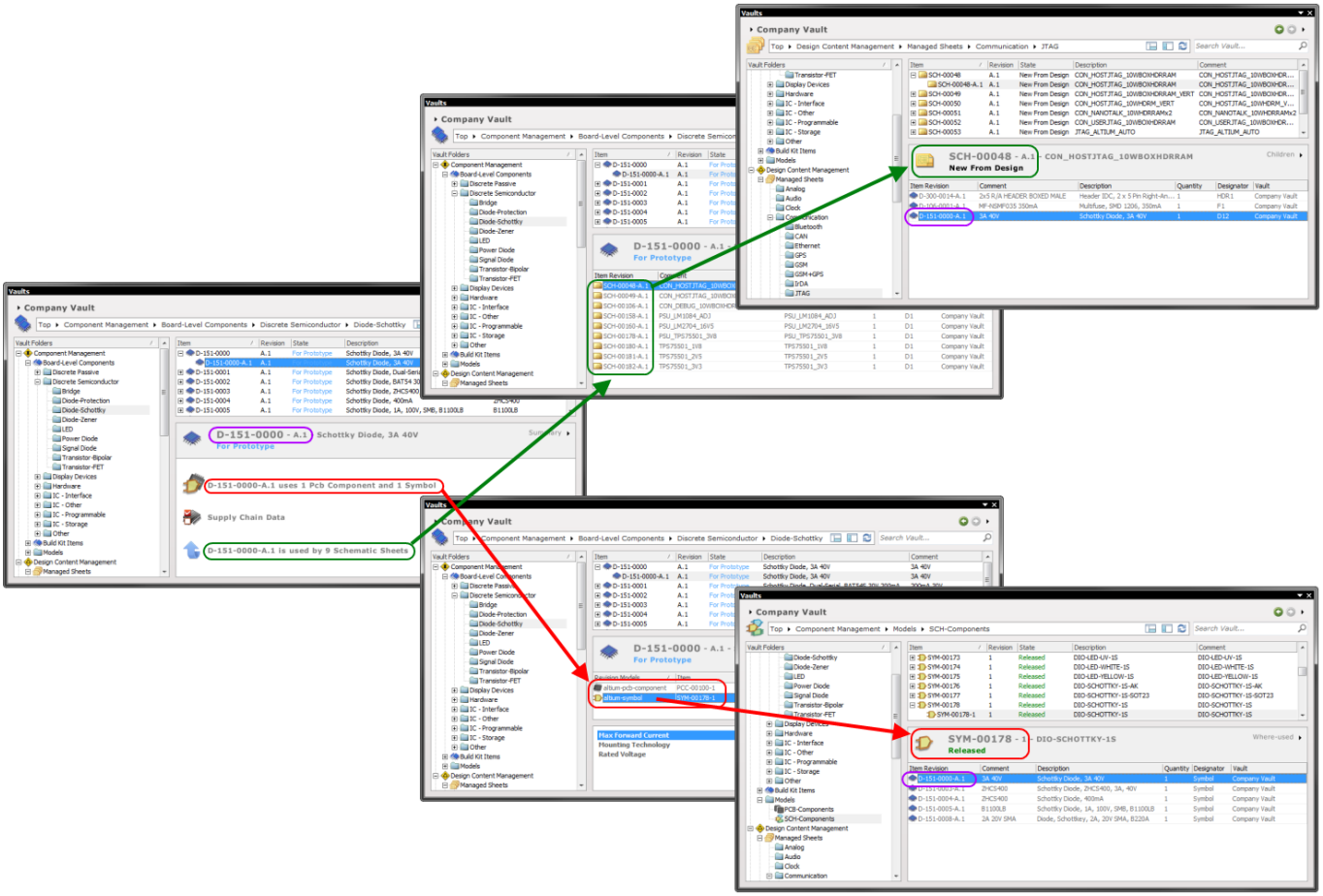


Successful release of a configuration of a design project! The release data (snapshot + outputs) is committed to the target vault, stored in the revision of the targeted Item. The released documents and design snapshot are summarized in the Release Summary dialog.

All stages in the process flow must run successfully, otherwise the release will fail and no data will be committed to the new Item Revision in the vault. As the process is fully automated, the risk of errors associated with a manual release process are no longer a consideration. Full validation, full checking. The data set for use by the supply chain is exactly what it needs to be, to produce the product exactly as you designed it.

Where-Used Capabilities

The highly relational structure of the 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 board designs it has been used in, which managed schematic sheets, and so on. Conversely, and from a parent Item's perspective, you are able to quickly browse its children.



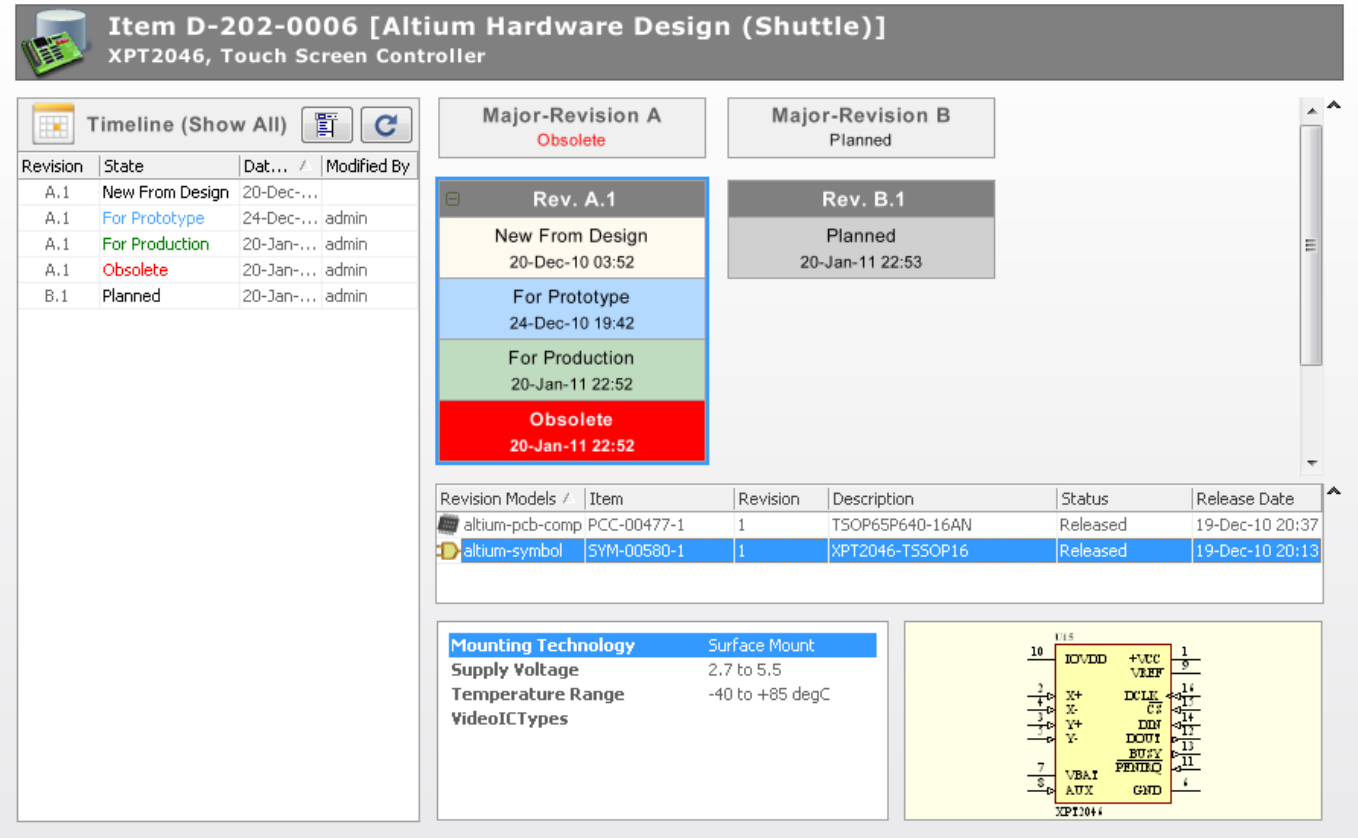
Where-used functionality enables you to quickly browse parent-child relationships in the vault.

The ability to traverse the content of a vault in this way can offer great benefit, not least of a time-saving nature. For example, 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 assured confidence.

Managing Item Revision Lifecycles

For a detailed view of the Revision and Lifecycle history of an Item, right click on the Item in the **Vault Explorer** panel and select **Full Item Details** from the menu. The **Item** view will open, as shown below.



The Item view gives a detailed history of the release timeline of Revisions and lifecycle state changes.

The detail that is presented in the **Item** view will depend on the type of Item being examined. For example, for an assembled board Item the view presents a list of released documents, the snapshot of the design project used to generate the release and the generated Bill of Materials (BOM) that will be used to physically manufacture that board.

The **Item** view displays a column for each major Revision. Each column then displays the Lifecycle state changes for that Revision. Right-click on a Lifecycle state cell to:

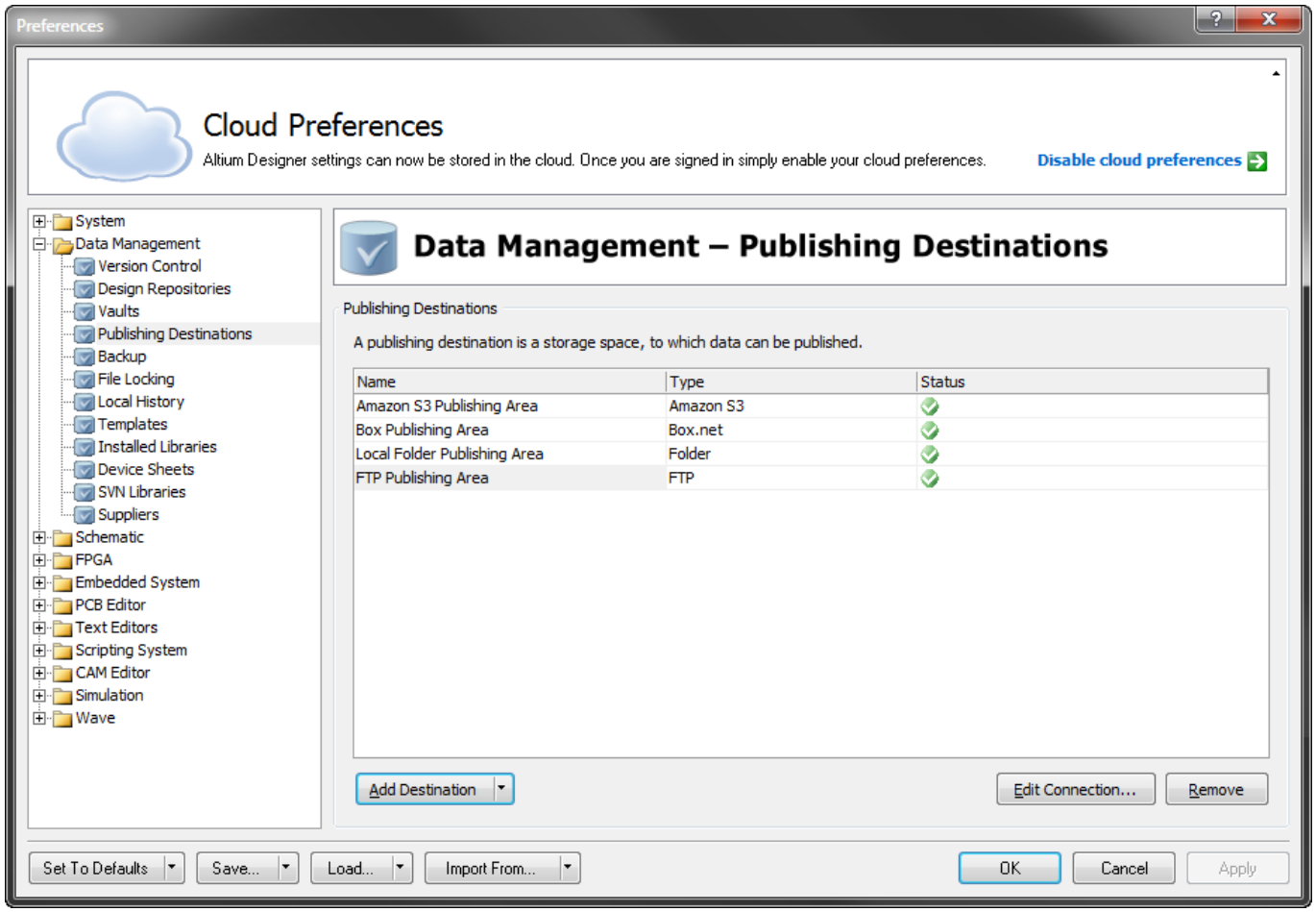
- Establish a new Planned Revision of the Item, in accordance with the Revision Naming Scheme selected for that Item.
- Manage the Lifecycle state for a particular revision of the Item, in accordance with the Lifecycle definition selected for that Item.
- View Revision properties.
- Retrieve documents associated with each revision of the Item.
- Publish release data directly to a nominated publishing destination (for Blank Board and Assembled Board Items only).

An Altium Vault allows you to implement security to ensure only those people allowed to change the revision to a certain state, for example from *Prototype* to *Production*, have access to do so.

Publishing Released Board Design Data

For released data generated from a PCB design project, an Altium Vault supports the ability to publish those released documents – generated output from Output Job files assigned to the released project configuration – for any Item Revision, to a storage space, such as Amazon S3, FTP, Box.net, or a simple folder location on a shared network. In terms of distribution and collaboration, this provides an unparalleled advantage in a world where the collective members of the overall 'product team' – the design team, the manufacturing team and all others involved in the process of getting a product from thought to reality – are often dispersed around the globe.

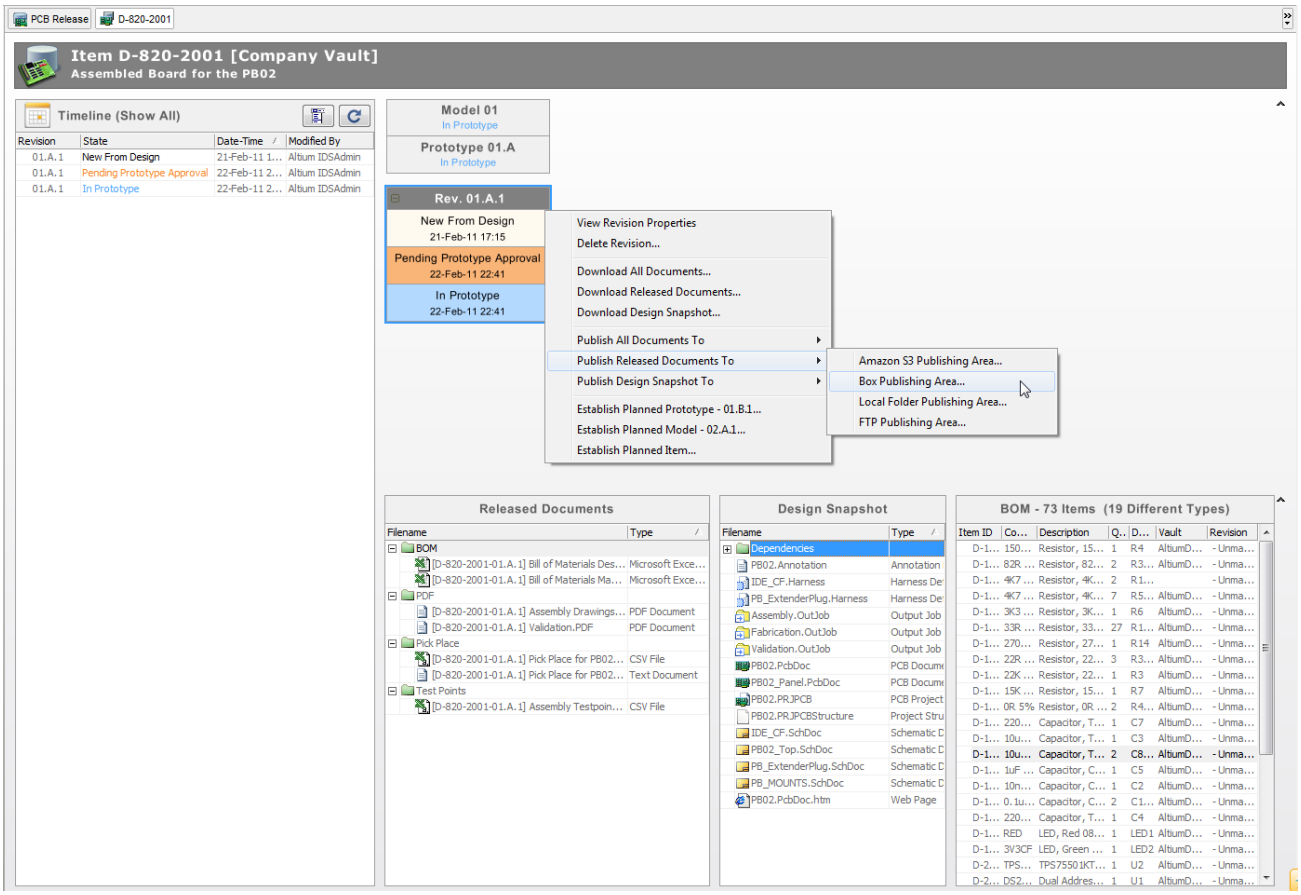
Publishing is simply a matter of defining a *Publishing Destination* and then uploading the released data for the required Item Revision to that destination. Publishing Destinations are defined on the **Data Management – Publishing Destinations** page of the *Preferences* dialog (**DXP»Preferences**).



Publishing Destinations are specified as part of the system preferences.

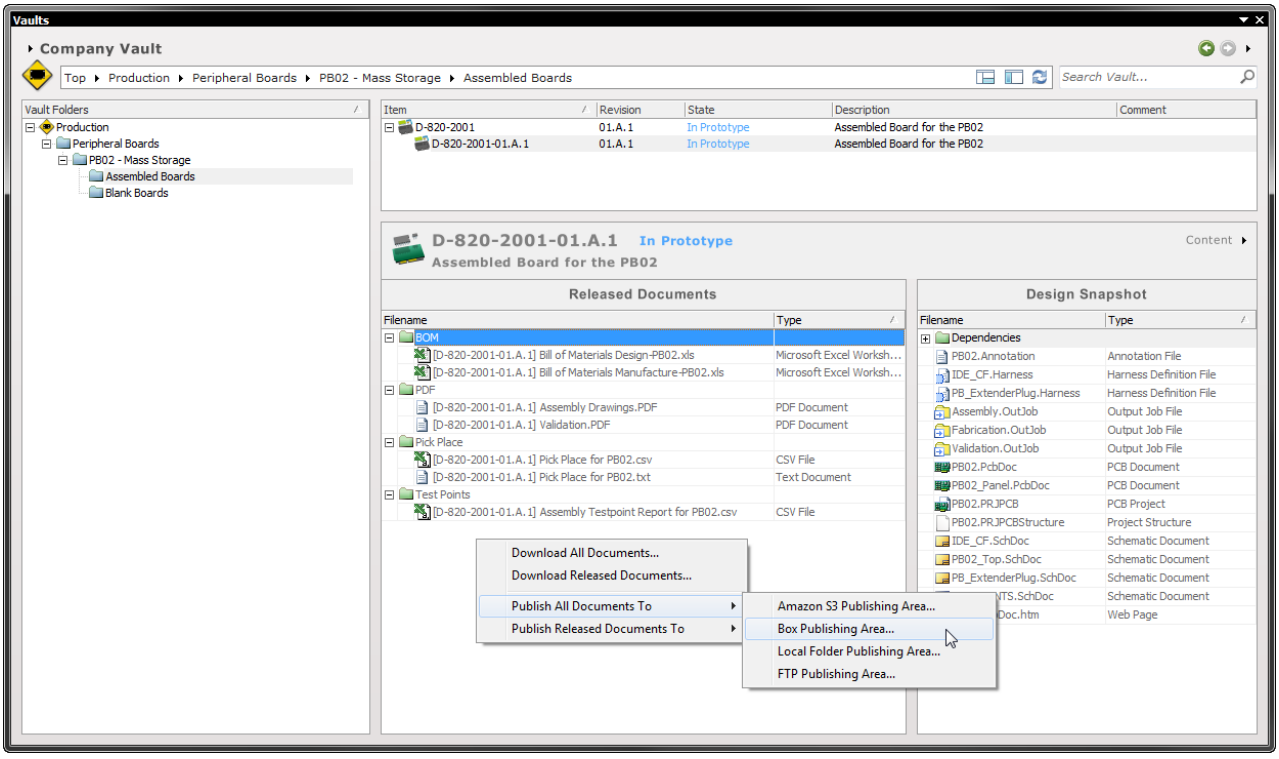
Publication itself is performed from the detailed view for the Item – the **Item** view – or from the **Vault Explorer** panel.

From the **Item** view you can publish released documents – the generated validation reports and outputs from Output Job files assigned to the project configuration that was released – and/or the source design snapshot. Simply select the Item Revision whose documents you wish to publish, then right-click and choose which documents to publish – All, Released Documents or Design Snapshot – and then choose which Publishing Destination to publish to. The sub-menu lists all available Publishing Destinations, by name, as defined on the **Data Management – Publishing Destinations** page of the *Preferences* dialog.



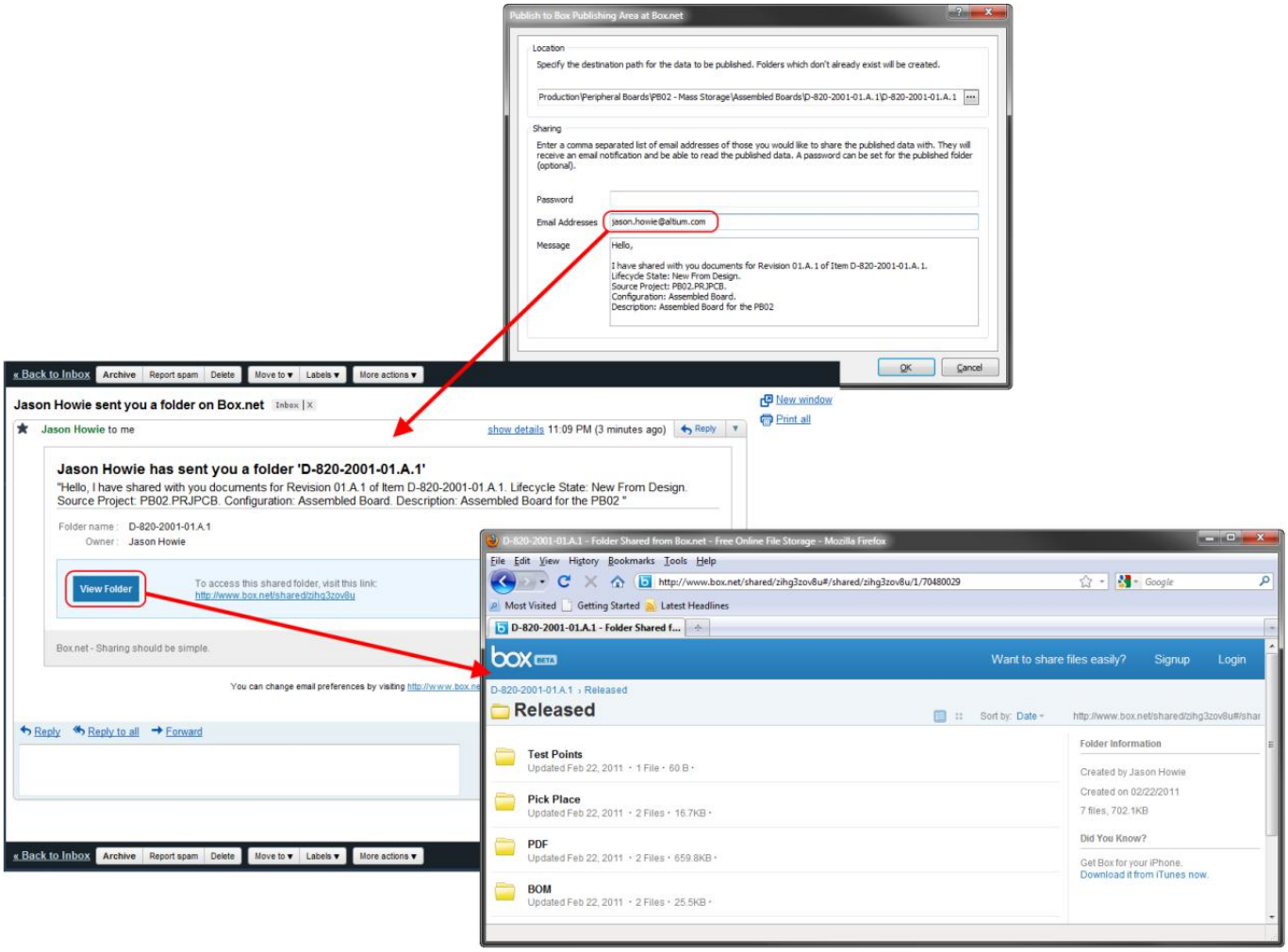
Access publishing-related commands for a particular revision of an Item from within the detailed Item view.

From the **Vault Explorer** panel, simply select the specific revision of the Blank Board or Assembled Board Item you wish to publish documents for, and switch to the **Content** view. Publishing commands are available from the right-click menus for the **Released Documents** and **Design Snapshot** panes respectively.



Access publishing-related commands for a particular revision of an Item from within the Vault Explorer panel.

Use the subsequent *Publish to* dialog to define the required destination sub-folder in which to store the data. With a Box.net Publishing Destination, you can also determine – through email – who to share the published data with.



Example data published and available in a cloud-hosted Box.net account.