

Summary

This document provides a conceptual walk-through of Altium's solution to board design release management – a solution that ensures the integrity of the design data being released.

Innovation. Creativity. Flexibility. These are just three words synonymous with the 'holy grail' of an engineer's preferred design environment. An environment in which the designer has the freedom to implement new ideas and to breathe life into old processes. In short, the environment in which to nurture that next cutting-edge electronic product.

Stifling. Restricting. Limiting. Three more terms that could well be used to describe a company's existing data and release management system. The need to ensure high-integrity data often walks hand-in-hand with layers of bureaucratic 'red tape'. How many times has a quick change to enhance a design become stalled in a plethora of required signatures and approvals? Ultimately, the designer is confined to design according to formalized processes, locking down design changes to ensure minimal impact to the integrity of the design data.

Ironically, these rigid and restrictive manual processes are, at the end of the day, still error prone. What is needed is a release management process that helps you solve design management problems *without* tying your hands behind your back.

Considering the electronics development process, Altium Designer has been harnessed to effectively and efficiently tackle the problem of providing the high data integrity, while ensuring that the demand for product enhancement and innovation can continue to be satisfied with complete design freedom. As part of its wider Design Data Management System, Altium implements a model that allows a formal definition of the links between the design world and the supply chain that is ultimately responsible for building the actual products. A model that maps the design data to specific production *Items* (blank and assembled boards) that the supply chain is actually going to build.

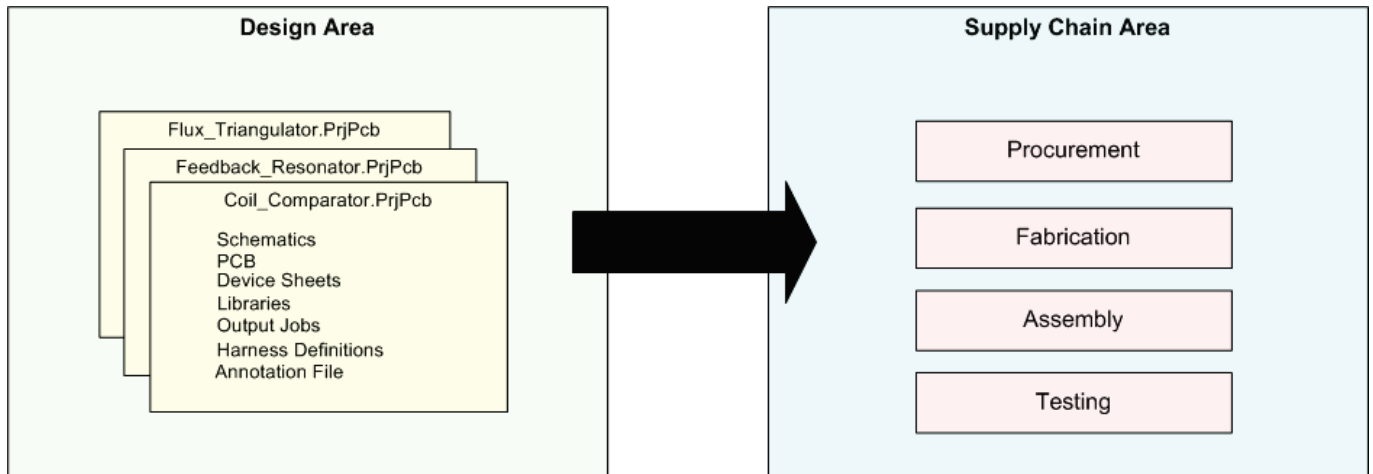
With this model in place, and with a range of supporting features and technologies, you are able to pass data from the design domain to the production domain in a pain-free, streamlined, and automated fashion – generating data output of the highest integrity, literally at the touch of a button!

The Product Development Landscape

Before taking a look at Altium's solution to board design release management, it is a good idea to take a step back and consider the people that are involved in the process of turning an idea into a fully-fledged product for sale in real-world markets – the 'players' in the game as it were.

On the one side, we have the people that take an idea and capture it as a fresh design. These are the engineers and designers who revel in their technical mastery to overcome design challenges, to create and innovate with design flair and panache. These people collectively form the *Design Team*, with their work environment affectionately tagged the *Design Area*.

On the other side, we have all the teams that are collectively responsible for turning a design into a physical product. This includes, fabrication, procurement, assembly, testing, and so on. They are responsible for realizing the fruits of the design team's labor, to build that tangible product exactly as the design team envisioned. The area that these teams work in can often go by different names, including the *Production Area*, or the *Manufacturing Area*. However, the umbrella term *Supply Chain Area* better reflects the collection of all teams and sub-teams in this area.



The task of getting an innovative new idea from design to production involves teams divided into two distinct areas - the Design Area and the Supply Chain Area.

Managing the Design Data

The design team works, often in a highly-collaborative nature, with the raw source design documents. The supply chain runs with sets of data produced, or generated, from a design. But just how are the data – both source and generated – handled? How do the design team keep track of their changes during the design phase, and where does the supply chain go to get the generated data they require to ultimately manufacture? The following sections take a closer look at where these data sets are stored.

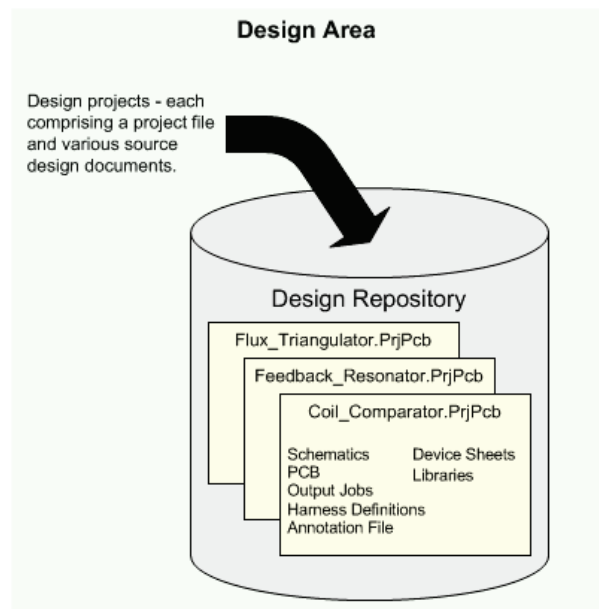
Design Repository

The design team keep track of changes to their source design projects using a *Design Repository*. This is a centralized repository in which all source design projects are stored. Owned by the design team, the Design Repository contains a high-resolution view of the history of the design process and is the primary collaboration tool used by the design team.

Design projects in Altium Designer represent the design team's intent for how the system should be built and how it should work. Each design project includes a project file and a variety of source design documents that together define the object being designed. Furthermore, a design project is parametric in nature, whereby a single project can potentially define many different Items that get manufactured in the real world.

The Design Repository becomes the central repository from which multiple team members can check data in and out, all the while maintaining a complete revision history of all changes made to the design. A design is therefore stored as a series of versions of its constituent project and source documents, building a progressive picture of the intent of a designer over time. By using a version-controlled Design Repository, you have the integral assurance that no revision of a design is ever lost, allowing for safe collaboration on the same design between members of a team that can be geographically diverse in their locations. The very nature of the version control system provides an audit trail for the design. Full accountability arises through transparency of who changed what, in which source document, and when.

Additional features and capabilities within the software, such as PCB Design Collaboration, allows multiple designers to run with the same design, confident in the knowledge that their individual layout changes can be collated and merged into a single PCB design document, and checked safely back into the Design Repository with controlled ease. The repository's history of the



The concept of the Design Repository - a centralized repository for the design team.

design – the various checked-in versions of the project representing incremental changes made over its lifetime – allows you to 'roll back' to any of the previous versions, with total control.

As part of Altium's Design Data Management System, support is provided for creating and connecting to version-controlled Design Repositories that use Subversion or CVS as the version control provider. By connecting to a repository you are, in effect, registering that repository with the system – telling Altium Designer of its existence, as it were. Furthermore, there is no manual specification of paths to 'unofficial' or 'rogue' repositories. Through Altium Designer, you can only interact with those VCS-based Design Repositories you have purposefully connected to the system.

You are in no way forced to use a version-controlled Design Repository to gain access to Altium Designer's high-integrity board design release process. You can still release design projects whose source documents and dependencies are stored in folders on or across local or networked storage locations, outside of version control. However, the full benefits of the Design Data Management System really do come into play through the dedicated integrity inherent to version control. A version control repository provides safe, reliable, and secure storage for all your design documents, giving you peace of mind that all the data checked into your Design Repository will be protected against accidental loss.

Altium Vault

The data that are generated from a source design project are the products of a high-integrity release process, as we shall see later in this document. These data are therefore said to be 'released' from the Design Area. To the supply chain, it is the 'instruction set' by which to fabricate a board or assemble it.

Providing an efficient means by which to handle such data in a highly secure, yet readily-accessible manner, Altium delivers a server-based engineering content management system solution — an *Altium Vault Server*. As 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 it has reached in its 'life' and therefore what it can be safely used for.

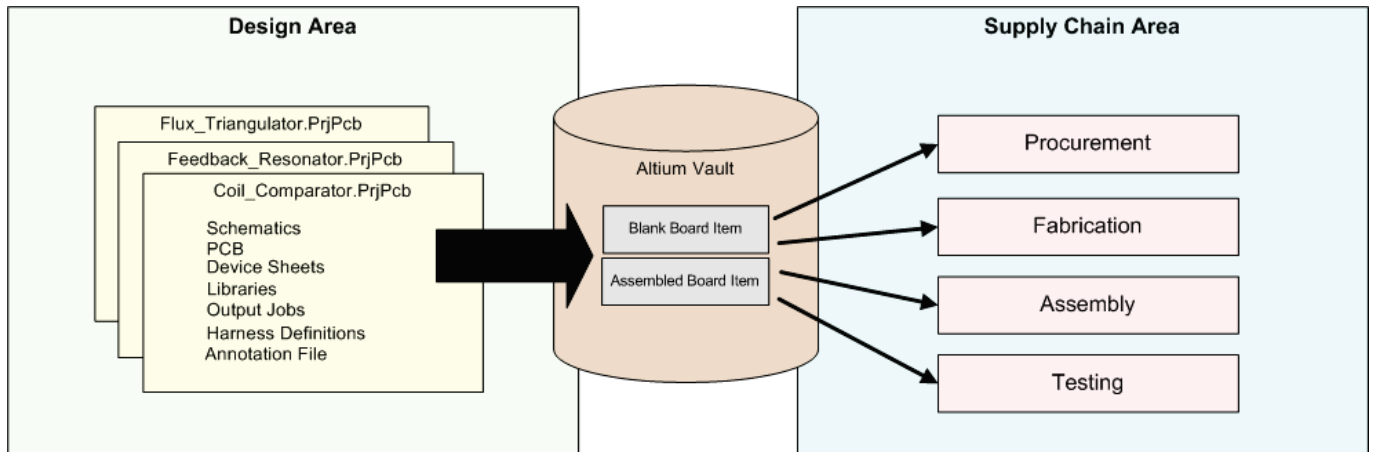
The various data entities (released from the Design Area or otherwise) are represented in an Altium Vault by unique Items. An Item simply represents a specific object, and is uniquely identified through use of an assigned Item ID. An Altium Vault then, is a centralized storage system into which all data, for each Item, are stored. The vault essentially provides the common ground between the Design and Supply Chain areas. Access by both areas to a single repository of data that are stored with secured integrity.

What exactly is represented by, and stored in an Item can vary, depending on what has been mapped to that Item. For the release of a PCB design project, an Item 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).

Altium's vault technology comes in a single flavor, formally known as the *Satellite Vault Server*, or more commonly *Satellite Vault*. With a Satellite Vault the authentication server runs on the AltiumLive site while the data and data server are installed on your LAN/WAN. User authentication for vault access is handled by AltiumLive's Identity Service – as such, login to a Satellite Vault is made using your AltiumLive Credentials.

A *Satellite Vault Server* is freely available for installation and use, provided you have a valid Altium Designer license and current Altium Subscription for that license. Installation is through a separate, downloadable setup executable.

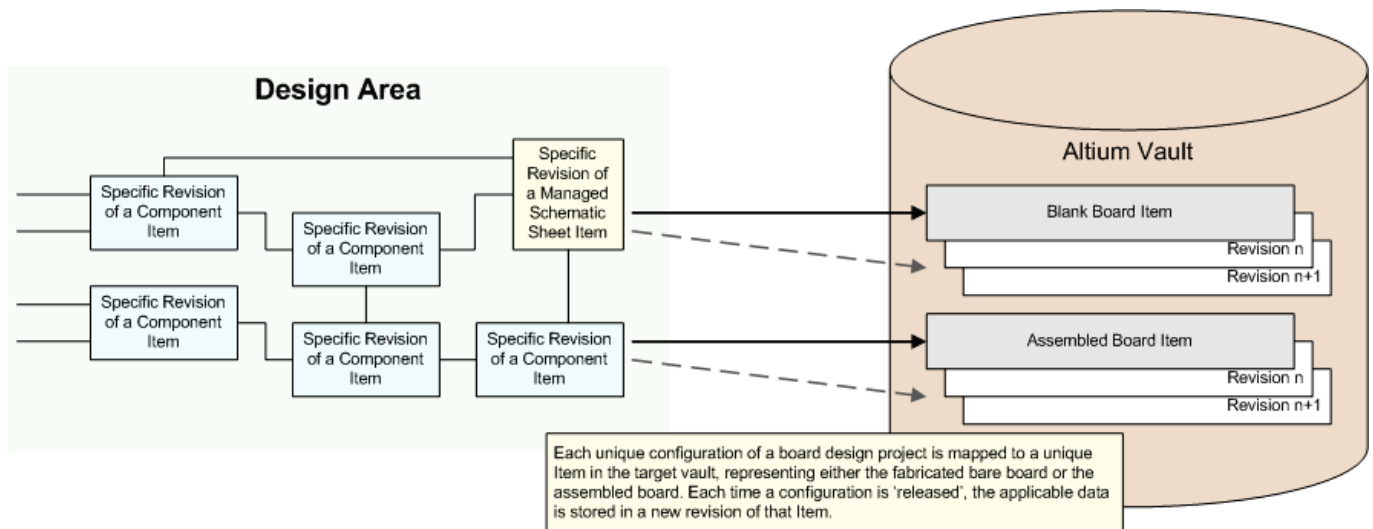
The Vault Server can always be accessed by opening Altium Designer and connecting to it on the same physical computer on which it is installed and running. This can be done by different users. When a different user signs in to Altium Designer on the same computer, and connects to the vault, they become a local administrator of that Vault Server instance. This means that if the previous owner leaves the organization, or there is a disconnection from the Internet (and hence, Altium's ID server), then the vault can still be used and accessed at least at a local level. Therefore data recovery is always possible in these circumstances.



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

In fact, an Altium Vault is used to store more than just the data generated from a board design. It is used to manage other sets of data obtained from the Design Area through the running of specific release processes. This includes the release of component definitions, domain models, schematic sheets of circuitry and design templates. And the vault is not just the destination for released data traffic from the Design Area. It is also used to handle other data, data that could originate from the Supply Chain Area too. Part Choice Lists are a good example of data that are very much vault-based in nature, but are not sourced from the design side, and are therefore not 'released' entities.

In a vault, each Item 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.



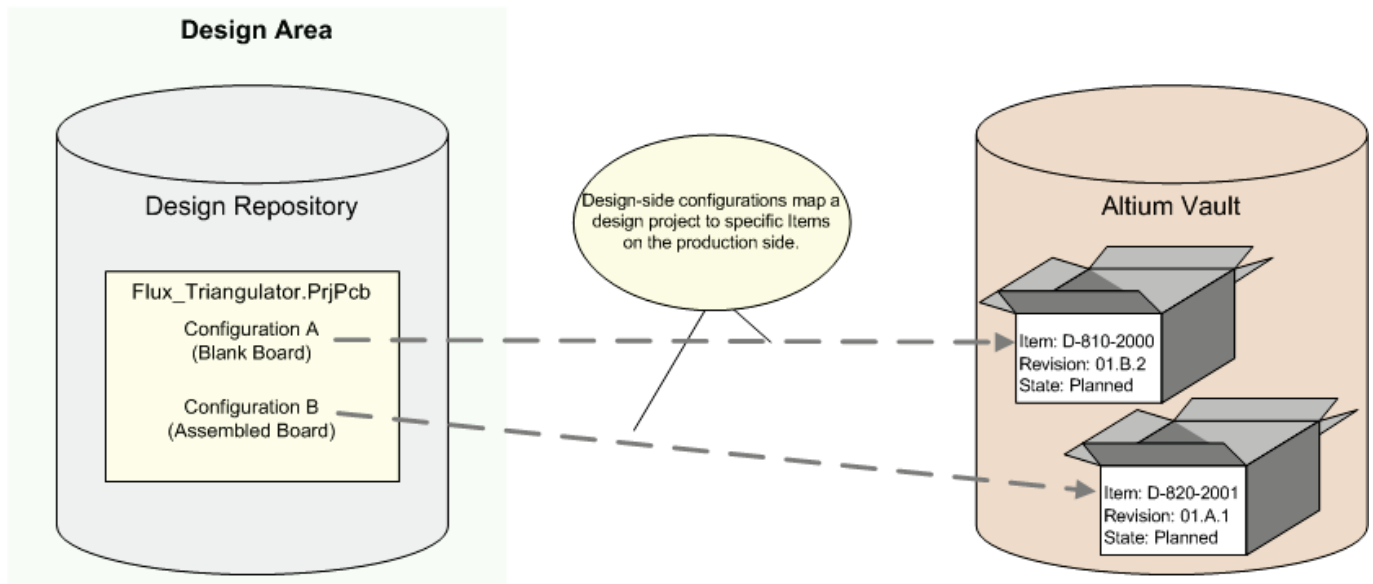
The concept of Item Revisions, illustrated for data that are generated from a source PCB Design project.

Source to Vault – The Vital Link

So then, on the one side we have the design team, doing what they do best – creating innovative designs. On the other, we have the supply chain, taking data to build the end product – the physical embodiment of the captured design. But just how do we specify which Item is to be made from the design data? 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*.

Configurations are part of the actual design project and provide the link from the design world to the manufacturing world. Each configuration represents an Item that we want to build in the real world, defining the data that will be required by a manufacturing organization to actually build that Item. When we release a board design project, we are in fact releasing a configuration of that project. The generated 'release data' are stored in a revision of the target Item specified as part of that configuration.

Each unique configuration of a board design project maps to a unique Item in the vault referenced by that project. Each time we release a configuration, the data generated are stored as (and in) a new revision of the target Item – an Item Revision. Before we can release a configuration therefore, we must ensure that a new Item Revision is created ready to accept the release data, and that this revision is in the *Planned* state.



Configurations provide the vital link between the Design Area and an Altium Vault, each targeting a specific Item in the vault.

Board design projects are parametric in nature. But what does this actually mean, and how does it relate to configurations and the Items manufactured from those configurations? In other words, what makes a design parametric? The answer becomes evident if you think of a board design project as data that can be generated from it, and how those data are 'driven'. Data are generated using one or more *Output Job* files. The output generators in these files can, in turn, be driven through the use of *Variants*. By changing the variant used to drive (possibly different) Output Job files, we can create differing and unique configurations of a design. The variants and Output Job files are then, in essence, the variable 'parameters' of a design – the very entities that make a design parametric.

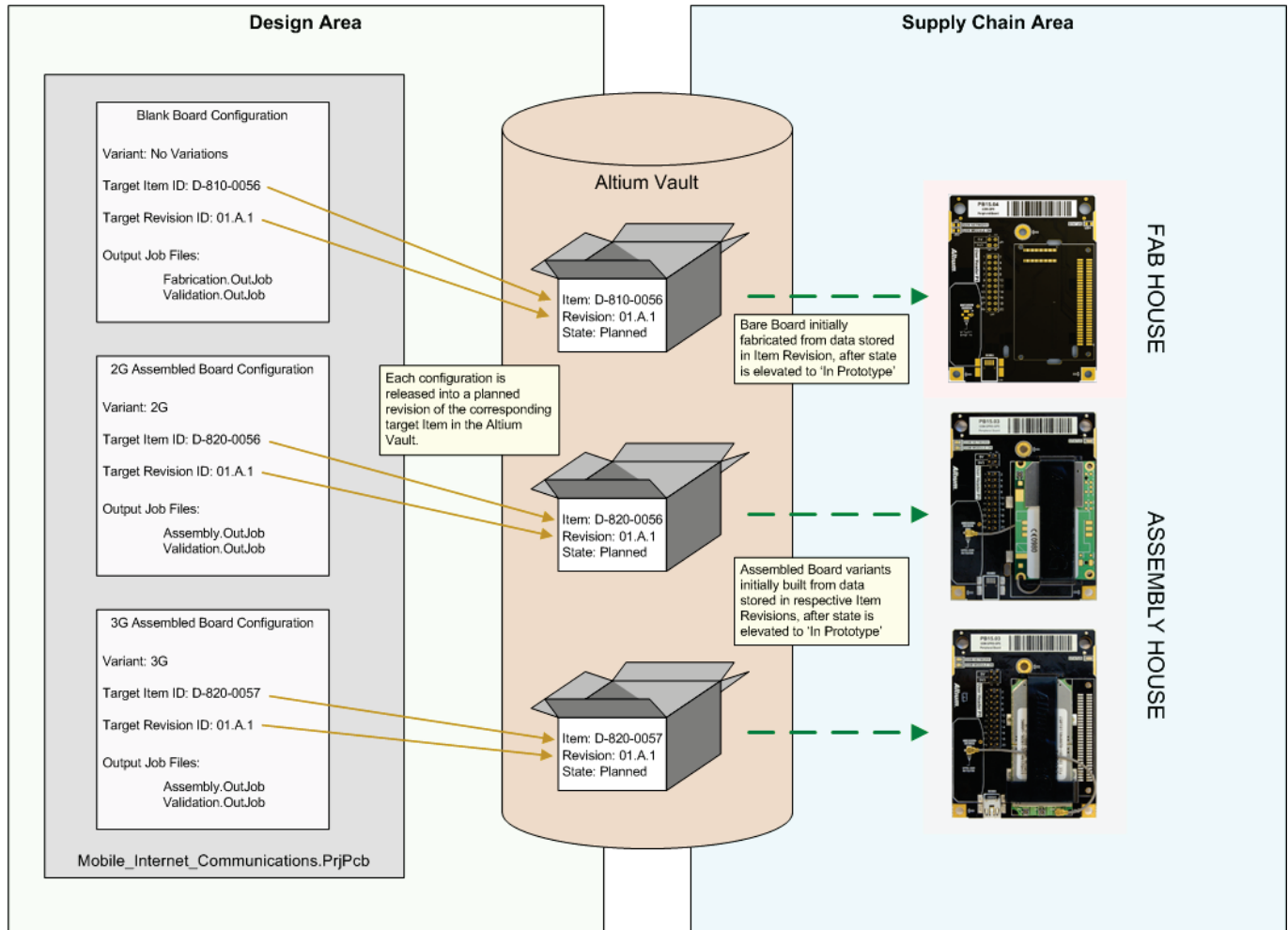
From an assembly perspective, the one board design project could have multiple variants defined that use the same fabricated bare board, but differ in the components used to populate that board, the result being distinctly different products when assembled. For example, consider the design for a communications board that provides mobile Internet functionality. From a 'superset' of components, two assembled variants of that design could be manufactured – one populated with components to offer 2G-based communication, the other to offer 3G. These are variants of the design, specified as such within the PCB project, and defined in terms of which components are fitted, or not. In fact any number of these 'assembled board variants' could be defined. From a fabrication perspective, we can also use variants to control changes in the silk screen for a bare board, giving rise to 'Fabrication variants' (of a sort).

The point to make here is that each variant, be it used to drive creation of a bare board or an assembled board, gives rise to the manufacture of a different, unique Item in its own right – and a unique Item requires that we create a unique configuration of that

PCB design. It is the choice of variant that drives the generation of the outputs defined in underlying Output Job files. So, when we define a configuration, we not only map that configuration to a dedicated Item, but also:

- Assign the Output Job file(s) that will be used to generate the data when releasing that configuration.
- Specify which variant (if required) is to be used when driving those outputs.

Considering the mobile Internet communications design, we would define three configurations – one to target the fabricated bare board Item that is common to both assembled boards, and the other two to target the 2G and 3G assembled board Items. The BOM for each assembled board would include the fabricated board Item.



An example PCB design project using variants to produce two different assembled boards, sharing a single common fabricated bare board. Three unique configurations for the project map to three unique Items in the target vault. Once authority to manufacture a prototype is given, release data from the relevant Item Revision is used to manufacture the real-world board.

The assembly process, which will be different for the 2G and 3G assembled boards, will need to include labeling of the PCB to indicate which revision of which Item was used to drive the assembly process. This can be catered for by inclusion of components in the design project to represent the labels that will be applied to the PCBs at assembly time. There would be two of these, one included in each variant.

The bare board will have its own identification which can be fabricated into the PCB. This will relate to which revision of which Item was used to drive the fabrication process.

A configuration also provides the means to constrain the release process, through the ability to define validation-based output reports as part of an Output Job file assigned to it. These validations – ERC, DRC, checking that source schematic and PCB design documents are synchronized, and comparison of footprints on the board against their source library – ensure the validity of the data that are being sent to the manufacturing team.

The Board Design Release Process

Getting deep and dirty 'under the bonnet' of a design truly is an Engineer's playground. The fun of creating that next intelligent (and lucrative!) electronic product, the mastery of laying out the design on a PCB or testing, in real-time, the operation of an embedded design running in a physical FPGA device – but that's just the half of it. Once a design is considered ready for flight into the wider world, it needs to be released – a process that can often be underestimated and therefore approached with a slight sense of nonchalance.

Without a regimented and fundamentally-sound release process, tracking an ever-maturing product over time can be fraught with any number of pitfalls. Just imagine needing to go back and release a previous revision of your product, only to find that all required source design files were never included, as a snapshot, with that particular release! That's OK, just get the generated output files from the relevant folder(s) for that release – trusty Gerbers and any additional fabrication and assembly information, that's all that's really needed anyway. But imagine those 'trusty' outputs have been overwritten or become corrupt somehow. If only the release process were more robust.

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 or, more specifically, defined configurations of those projects, without the risks associated with manual release procedures. When a particular configuration for a 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 to which the configuration is mapped, and which represents the physical object that will be manufactured by the supply chain.

The overall result is the highest-integrity board design release management possible. Not only is your actual design project tightly monitored, backed-up and managed in a version-controlled Design Repository, but also too, the releases of its configurations in a similar manner within the target Altium Vault – robust, safe, secure.

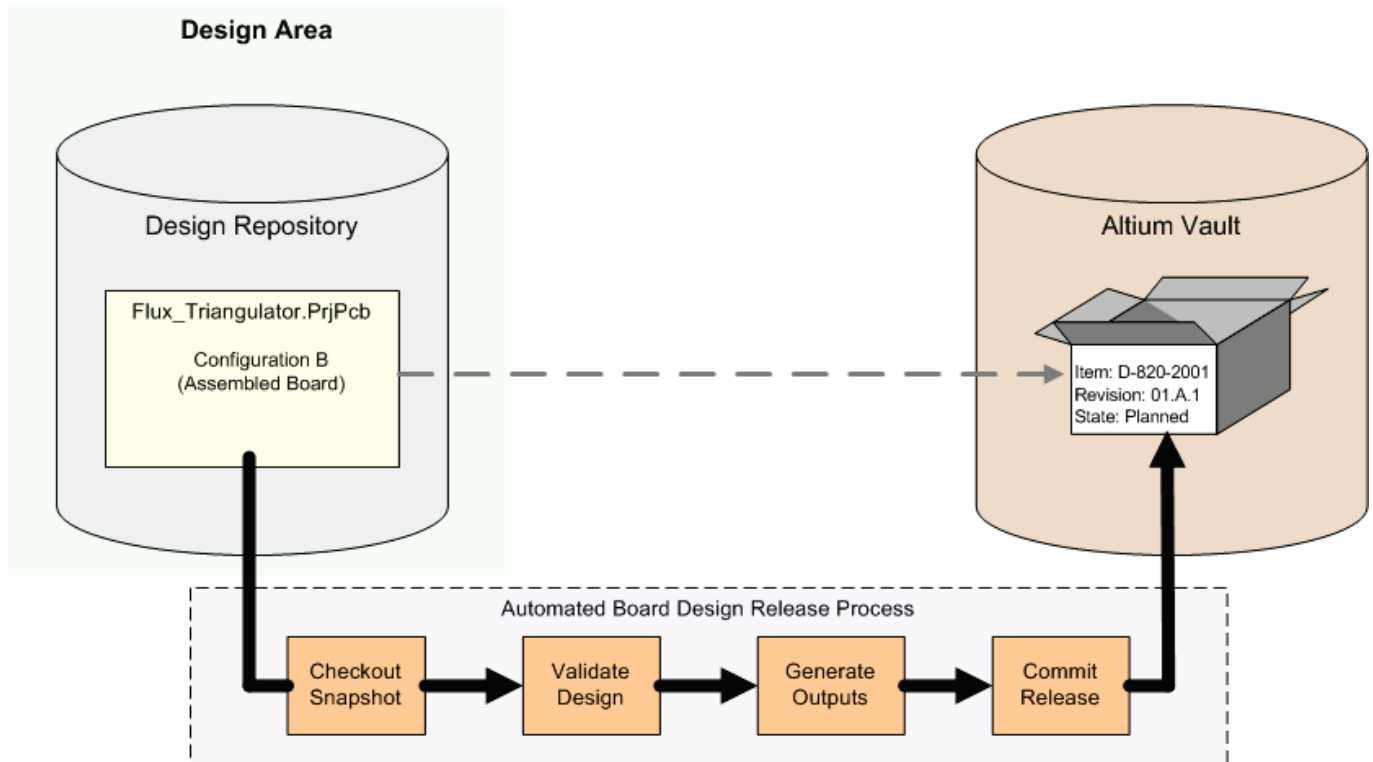
The act of release – referred to as the *Board Design Release Process* – simply generates the data required to build a newly 'planned' revision of a target Item and stores that data in that revision within the target vault with the utmost integrity. The process of releasing a design involves three key steps:

- Creating a new *Planned Revision* for the Item you wish to release in the target Altium Vault. This reserves that revision and essentially locks the Item, so no-one else can release to it.
- Update the corresponding configuration for the project – that maps to that particular Item – to reference this new revision of that Item.
- Initiate the release.

The third step – the release process itself – is performed using Altium Designer's *PCB Process Manager*, the user-interface to which is provided courtesy of a dedicated view within the software – the **PCB Release** view. This view 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 board design 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.

With the 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.

Whether running validations and generating outputs on-the-fly in Design Mode, or as part of a full release in Release Mode, Altium Designer provides powerful output path definition capabilities as part of its output container setup when configuring and assigning Output Job files to your PCB Project Configurations. Intuitive and sleek, you will always have full control over where your outputs get stored.

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.

The whole concept of configuration release management provides the ability to identify the point in time at which any given configuration of the PCB design was released, go back to this point and retrieve, modify and re-release that configuration with the correct dependencies.

For a design project stored in a VCS-based Design Repository, during the release process Altium Designer records the version control repository address and revision of the project and commits this information to the target Altium Vault. This method means that at any point in the future it is possible to retrieve an identical snapshot of all files in the design project from version control.

Accessing the Released Data

You've released a configuration of your design project. The board design release process is complete – all stages have run successfully and release data generated from the design have been committed to the nominated vault, stored within a specified revision of the targeted Item. So what now? How do you 'get at' those generated data, and of equal importance, how do you manage the lifecycle of the revision? The answer is by using a dedicated view of the Item.

This view provides a detailed look at a chosen Item in the vault. Information regarding the life of the Item – in terms of its revisions – is presented in the form of a 'release timeline', as well as graphically. For each Item Revision, detailed release data are presented – the data generated from the released configuration of the board design and stored in that revision (generated outputs and validation reports, design source snapshot, System BOM). The view provides all the controls you need to interact with the Item and its revisions – including the ability to download generated documents and to manage the lifecycle of each revision.

The screenshot displays the Altium Vault interface for Item D-820-2001 [Company Vault]. The interface is divided into several sections:

- Timeline (Show All):** A table showing the revision history.

Revision	State	Date-Time	Modified By
01.A.1	New From Design	02-Mar-11 1...	Altium IDSAdmin
01.A.1	Pending Prototype Approval	02-Mar-11 1...	Altium IDSAdmin
01.A.1	In Prototype	02-Mar-11 1...	Altium IDSAdmin
- Model 01:** A section showing the current model and its revisions.

Model	State
Model 01	In Prototype
Prototype 01.A	In Prototype
- Rev. 01.A.1:** A section showing the details of the current revision.

Revision	State	Date-Time
New From Design	02-Mar-11 16:25	
Pending Prototype Approval	02-Mar-11 16:25	
In Prototype	02-Mar-11 16:25	
- Released Documents:** A table listing the documents released for this revision.

Filename	Type
[D-820-2001-01.A.1] Bill of Materials Design...	Microsoft ...
[D-820-2001-01.A.1] Bill of Materials Manu...	Microsoft ...
[D-820-2001-01.A.1] Assembly Drawings...	PDF Docu...
[D-820-2001-01.A.1] Validation.PDF	PDF Docu...
[D-820-2001-01.A.1] Pick Place for PB02.csv	CSV File
[D-820-2001-01.A.1] Pick Place for PB02.txt	Text Docu...
[D-820-2001-01.A.1] Assembly Testpoint ...	CSV File
- Design Snapshot:** A table showing the design snapshot for this revision.

Filename	Type
Dependencies	Annotation File
PB02.Annotation	Annotation File
IDF_CF.Harness	Harness Definition File
PB_ExtenderPlug.Harness	Harness Definition File
Assembly.OutJob	Output Job File
Fabrication.OutJob	Output Job File
Validation.OutJob	Output Job File
PB02.PcbDoc	PCB Document
PB02.Panel.PcbDoc	PCB Document
PB02.PRJPCB	PCB Project
PB02.PRJPCBStructure	Project Structure
IDF_CF.SchDoc	Schematic Document
PB02_Top.SchDoc	Schematic Document
PB_ExtenderPlug.SchDoc	Schematic Document
PB_MOUNTS.SchDoc	Schematic Document
PB02.PcbDoc.htm	Web Page
- BOM - 73 Items (18 Different Types):** A table showing the Bill of Materials for this revision.

Item ID	Comm...	Descr...	Q...	D...	Vault	Rev...
D-100-0005	150R ...	Resist...	1	R4	Altium...	- U...
D-100-0076	82R 1%	Resist...	2	R...	Altium...	- U...
D-100-0088	4K7 1%	Resist...	2	R...	Altium...	- U...
D-100-0088	4K7 1%	Resist...	7	R...	Altium...	- U...
D-100-0098	3K3 1%	Resist...	1	R6	Altium...	- U...
D-100-0102	33R 1%	Resist...	27	R...	Altium...	- U...
D-100-0111	270R ...	Resist...	1	R14	Altium...	- U...
D-100-0113	22R 1%	Resist...	3	R...	Altium...	- U...
D-100-0114	22K 1%	Resist...	1	R3	Altium...	- U...
D-100-0125	15K 1%	Resist...	1	R7	Altium...	- U...
D-100-0137	0R 5%	Resist...	2	R...	Altium...	- U...
D-101-0003	220uF...	Capac...	1	C7	Altium...	- U...
D-101-0007	10uF ...	Capac...	1	C3	Altium...	- U...
D-101-0009	10uF ...	Capac...	2	C...	Altium...	- U...
D-101-0027	1uF 25V	Capac...	1	C5	Altium...	- U...
D-101-0077	10nF ...	Capac...	1	C2	Altium...	- U...
D-101-0080	0.1uF ...	Capac...	2	C...	Altium...	- U...
D-101-0088	220uF...	Capac...	1	C4	Altium...	- U...
D-153-0003	RED LED, ...	LED	1	LED1	Altium...	- U...
D-153-0004	3V3CF LED, ...	LED	1	LED2	Altium...	- U...

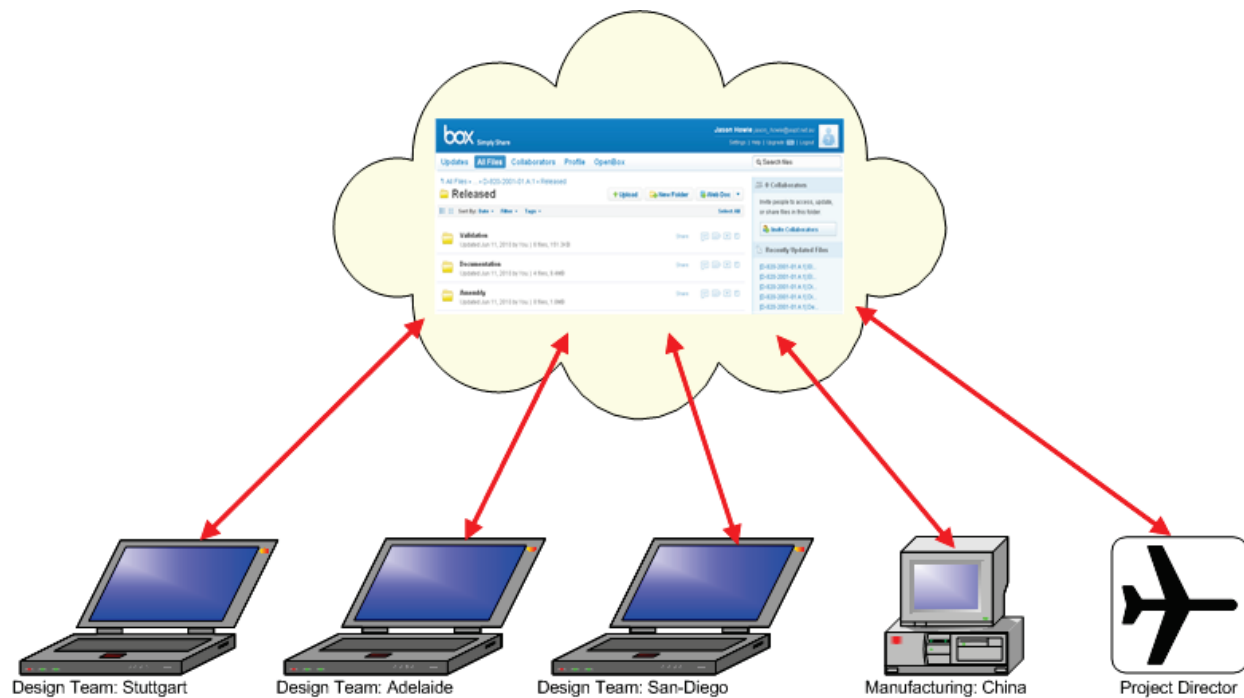
Any Item in the vault can be examined in closer depth using its associated (and dedicated) view. Here, you can inspect the Item and its revisions in fine detail, including management of revision lifecycle through purposefully-made state changes.

Publication of Released Data

For released data generated from a board 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 and Box.net. 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. As part of the system you nominate, by specifying email addresses, whom to share access to the data with. From the manufacturing plant in China, to the design teams in Stuttgart, Adelaide and San Diego, and to the Project

Director in-flight somewhere across the Pacific, everyone that *needs* to know about the new release will receive mail in their inbox with a link to the published folder – shared (and controlled) access to view, discuss and utilize the data with which to build the Item.



Publish your released data to a storage space and share information between team members across the globe.

One Piece of a Larger Puzzle

The high-integrity release of board designs to manufacture is itself part of a greater system still. Along with the next generation of component management and managed design reuse of schematic sheets, it makes up the set of mutually-beneficial, yet independently-functional sub-systems that together form Altium's wider Design Data Management System.

Use the board design release process on its own, or complement it by using these other sub-systems. The choice is yours. And by using all three sub-systems together, you can maximize the management and integrity of your design data. From the humble domain models to the parent vault-based components that reference them, to sheets of reusable schematic circuitry that is captured using those components, and ultimately to the board designs that use a combination of the two, all these data elements are stored, revision-controlled and lifecycle managed side-by-side in the same Altium Vault.

And even after a project configuration is successfully released, the vault provides the smarts to prevent lifecycle state elevation - for example to "Prototype" or "Production" - unless the child Items 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.

Ultimately, this parent-child dependent lifecycle relationship means that you can be assured that when you send the generated board data to the supply chain, they will be using data that is guaranteed in its integrity, right down to its constituent, company-approved components.