
Anaplan
Way

TRAINING MATERIAL DISCLAIMER

Anaplan training materials are intended to assist with the understanding of the Anaplan model building capabilities and are a tool in providing current and accurate information about the subject matter covered. Training materials are distributed and presented with the understanding that Anaplan does not render any accounting, legal or other professional advice. These materials are to be used for training purposes only. Some of the procedures and other practices as described may be different than actual work environments.

Material in the training materials may include technical inaccuracies or typographical errors. Changes may be periodically incorporated into the training material. Anaplan may make improvements and/or changes in the products, services and/or work aids described in these materials at any time without notice.

Each person engaging in Anaplan training activities bears full responsibility for his or her own training. Neither Anaplan nor its staff warrants or makes any representations regarding the use or the results of the use of the materials or information in the training materials. These training materials are provided 'AS IS'.

Table of Contents

1. Introduction

Anaplan is a unique software platform and requires a unique implementation approach. The Anaplan Way is a methodology that helps to ensure complete transparency during every phase of an implementation. Successful implementations require transparency amongst the customer, Anaplan, and engaged partners. This approach ensures strong deployment and adoption of the Anaplan platform.

The Anaplan Way is designed to be flexible and dynamic, allowing for the twists and turns that a project may take. We systematically and methodically document changes; we build rapid re-planning into Agile sprints embedded in the Anaplan Way.

Audience

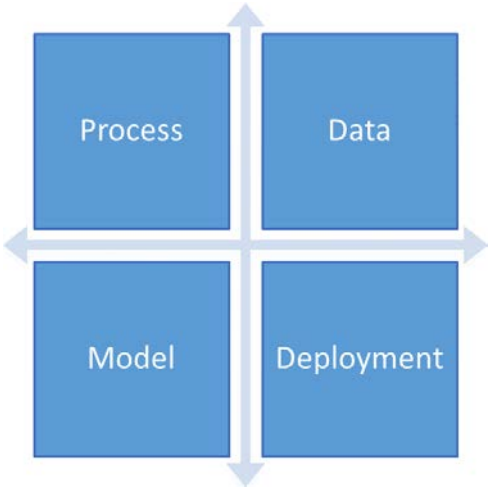
Who should read The Anaplan Way?

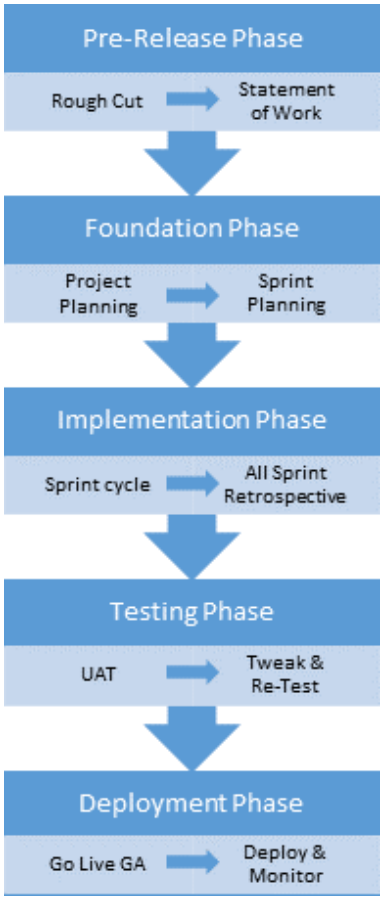
Anaplan customers, project teams, employees, and partners will find The Anaplan Way useful. The Anaplan Way ensures your team is leveraging best practices in planning, design, implementation, testing, deployment, and maintenance of the Anaplan platform.

What is Included in The Anaplan Way?

The Anaplan Way is an end-to-end document that covers all major phases and considerations during an Anaplan implementation. Use The Anaplan Way for the first release and any subsequent releases. While not exhaustive, The Anaplan Way is comprehensive.

Anaplan's platform has been successful because both the platform and the Anaplan Way flex and adjust to changes that often occur during a project. The Anaplan Way continues to drive successful releases because it is grounded in four implementation cornerstones (described in detail in chapter two). All four cornerstones should be considered during each project phase.





Each Anaplan Way phase is explained in its own chapter. The first phase, Pre-Release, includes:

- Preparing a rough estimate – called a Rough Cut
- Running a scoping workshop to collect necessary SOW data
- Creating a detailed estimate used to create the Statement of Work (SOW) that is signed by the customer

You'll learn about the types of information needed and the steps involved in putting together this important document in chapter three.

The Foundation phase (chapter four) includes:

- Setting up Launchpad, our initial customer training
- Conducting a kick-off meeting
- Writing user stories

Once you have defined user stories, you'll design the model and have the design reviewed. You'll also work with the customer to determine the user stories' priority and plan which stories will be built in each sprint. You'll learn how to manage the buckets. Using the managing the buckets process, you'll work to ensure that sprints are evenly allocated, even when new user stories are added. Missteps here will cause problems in subsequent phases.

The next phase is the Implementation phase. Chapter five includes the steps involved in building the model using an Agile Scrum methodology. In this chapter, you'll learn:

- How to conduct daily Scrum meetings
- Handling feedback from a Sprint Review and Retrospective
- What to track in the Agile Implementation – The Anaplan Way App

When the sprints are complete, the Testing phase begins, including:

- Testing usability
- Managing bugs
- Prioritizing change requests and enhancements

During the Testing phase, you'll work with your customer to determine what absolutely must be done in order to have the model generally available. Chapter six includes the steps involved in running UAT successfully.

Chapter seven continues with the Deployment phase. Remember that deployment is one of the cornerstones and needs to be considered from the start of the project. Activities in this phase include:

- Implementing a deployment plan
- Communicating about Anaplan platform and process rollout
- Confirming all of the documentation is in place to ensure a smooth handoff

Anaplan promotes customer self-sufficiency; our Center of Excellence model is included in chapter eight.

Anaplan security is covered in chapter nine.

Get Started with Key Ideas

Before getting started, carefully consider the following ideas. All Anaplan users need to understand these key concepts:

- Anaplan's pros and cons and reminders of some do's and do not's
- An Anaplan implementation is not about the destination, but rather the journey
- An Anaplan implementation is about fixing a broken process.

Anaplan Pros and Cons

Pros	Cons
You can build almost anything in Anaplan!	You can build almost anything in Anaplan!

It's true: Anaplan is the most flexible modeling and planning platform in the world. You can build world-class models--and really poor models. In order to launch successful, sustainable models, follow The Anaplan Way: design and build models using only tried and true techniques and best practices.

Anaplan Dos and Do Nots

Do	Do Not
Talk to other successful customers and business partners with experience in your use case & get their take on how to tackle your business problem.	Be tempted to replicate a process currently done in Excel into Anaplan. It almost never works, and what's the point anyway?
Take a look at the current processes and see if it needs updating.	Try to rebuild a broken process in the middle of trying to build an Anaplan solution. You will fail.
Keep it simple.	Boil the ocean. Do not try to jam everything into the first release. Remember, you will quickly iterate on each release.

It's a Journey, Not a Destination

Anaplan is a very different kind of platform. Anaplan was designed from the ground up to be imagined, created, and modified by customers – citizen developers. Customization is simple, and in many cases, Anaplan models need not be taken off line to make changes. You can try new ideas, incorporate feedback, make improvements, or implement new methods in hours, minutes, or less.

Business goals and strategies adjust as markets shift – more frequently than ever before in a modern business landscape. Anaplan models can change and shift with the way organizations do business.

A traditional waterfall methodology promotes a practice of gathering myriad requirements in order to force them into the first release, because in many instances, organizations will not see another release for a long time (sometimes years!). Waterfall methodologies cause much worry over what might have been forgotten. In traditional implementations, teams get unfocused, bored, and lack visibility into the outcomes of the project. New priorities come into play, putting projects on the back-burner; sometimes even leaving people wondering why the company embarked on the journey in the first place.

Businesses and people think in cycles. Miss a cycle and they are onto the next thing.

The Anaplan Way takes advantage of the natural cycles that exist in business. Anaplan helps teams think in short, focused releases, with each release building on the last. Generally, a release is ready for general availability in about two to three months.

Anaplan can change and shift with businesses because it helps customers:

- Keep their first release focused and short.
- Identify what they want to achieve and when they want to achieve it.
- Align their organization around four cornerstones and their Anaplan goal or manifesto. (More on that later!)
- Stay aligned with a less is more, keep it simple mentality.
- Follow a crawl, walk, run philosophy.
- Achieve and promote a quick win, focused on immediate pain versus a long process, focused on many pains.
- Demonstrate the value of an Anaplan investment.

By using the Anaplan Way, we help our customers stay focused on the quick win.

It's Not Just About Anaplan

When customers embark on an Anaplan journey, it's likely they are going to change an existing process that is missing functionality or broken altogether. In most cases, this requires modifying one or more processes that are upstream and/or downstream of the changing process.

The Anaplan platform will operate as a new component in the business process landscape. The customer must be clear on all processes that impact what is going into and out of Anaplan – as well as a clear understanding of the people, time, and systems really involved. Consider the process in-depth, as well as all processes that provide input to or rely on outputs from the model. Can all upstream or downstream processes support Anaplan?

Let's look at an example of a Territory & Quota Planning model. This use case calculates the quota to be allocated to a field sales representative to sell goods and services on behalf of the company.

The basic inputs to the process are:

- Strategic targets (planned and historical data)
- Sales organization list with a hierarchical structure
- Products list with a hierarchical structure

The Anaplan platform brings the targets together. Based on historical sales data, this can be easily mapped to sales staff and sales territories to calculate quota over time. Further, these quotas can also be easily allocated across the products in a multi-dimensional platform.

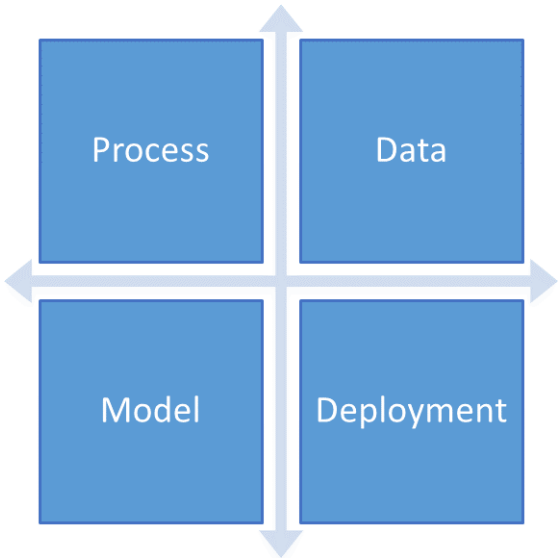
To calculate these pieces correctly, the Anaplan platform needs clean data and metadata. Often customers perform processes like this in Excel. These processes were not designed to take a central feed from sources that hold historical data and/or metadata. This becomes a new requirement ---one that exists *outside* of the core Anaplan model use case. The process of cleaning data and metadata is critical to project success and needs to be aligned with the project.

Also remember that while you are an expert in how Anaplan works, you are not expected to be a process expert. When major changes to a process are needed, it is best to recommend a process expert who can work with the customer to streamline the process.

2. The Four Cornerstones

When you embark on the Anaplan journey, you will need to focus on more than the Anaplan model build. The Anaplan platform cannot be successfully deployed in an unstructured manner.

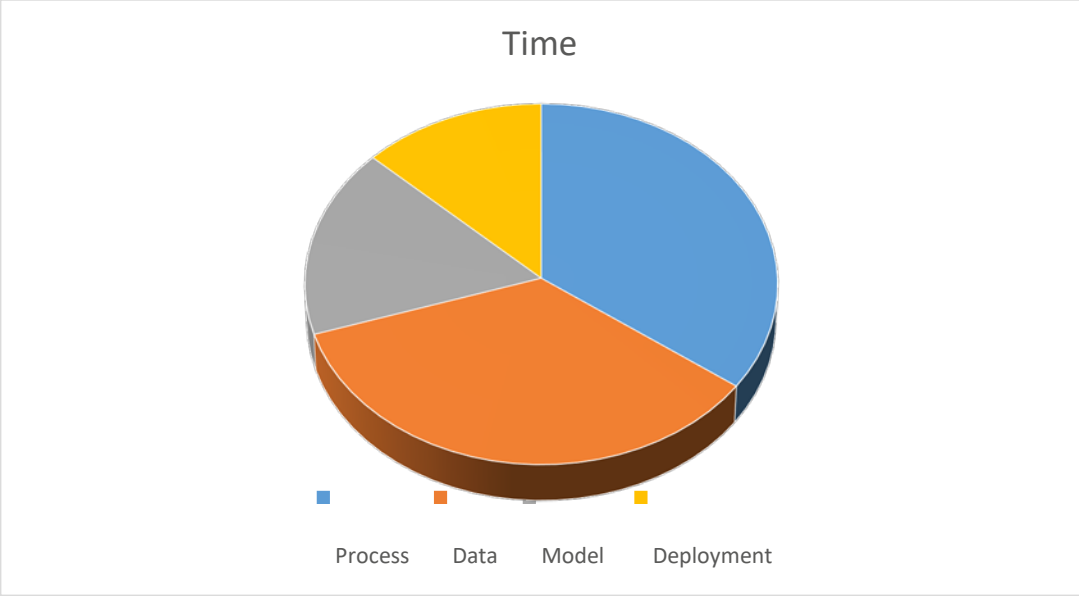
Like other mission-critical applications and processes, an Anaplan implementation and deployment needs to be planned carefully to be executed well. The keys are in the pre-planning, (requirements, team building, etc.), team collaboration and the fast, iterative execution using Agile implementation methodology. Once these components are in place, it is important to teach the customer about the fundamentals that lead to a successful implementation. We call these fundamentals the cornerstones of The Anaplan Way. Why? Cornerstones provide the foundation for the corners of a building.



The Anaplan Way cornerstones provide the foundation for an implementation of the Anaplan platform. These cornerstones must be planned, executed, and tracked during each phase. They are:

- **Process**: The wider business process that he Anaplan model supports.
- **Data**: All the components needed for the model: master, meta, and transactional data.
- **Model**: The design, build, and test of the Anaplan model.
- **Deployment**: A plan to ensure the Anaplan model and business process are adopted in the organization.

Anaplan is specifically designed so that you can build models quickly. It is not the model build that takes the most time, but rather the surrounding processes that support the build that require the most time. This graphic shows the approximate percentage of time you should spend on each cornerstone.



Cornerstone	Time of the Release
Process	35%
Data	35%
Model	17%
Deployment	13%

Process and data cornerstones present the biggest challenges, and require the most project time. Keep the focus on process and data simple to start and build upon the first release in subsequent releases.

Process

Process is a very important Anaplan Way cornerstone. If the customer does not have the desired processes agreed on and documented, it significantly impacts the success of the project.

Establish the process flowing through the model before the project work begins. This vital step is not something that can be done during the project. That is like building the plane while flying it.

Without clear understanding and alignment of the end-to-end business process being addressed, it will be very difficult to collect, document, and prioritize user stories. You may also risk misunderstanding associated upstream and downstream process dependencies.

Customers almost always use Anaplan to fix a broken process or system – sometimes both. Therefore, the actual process must be either fixed or optimized before the Anaplan implementation.

Anaplan's Role in Process Change

Anaplan's role is not that of a business process expert; our expertise is in the product, not in process re-engineering. Anaplan can offer guidance and provide industry best practices; customers must own their processes. Quite often, customers require specialist support when it comes to process. This can involve third party consulting organizations who specialize in planning and analytics process change.

Establishing Process in Scope

During an initial scoping session, the team workshops processes and identifies the areas Anaplan will address. This honest, transparent discussion helps to identify processes that are broken or need to be changed.

During the workshop, discuss Anaplan's approach and how it quickly addresses process changes in small release cycles. Anaplan's approach allows the project to focus on one small piece of the problem at a time. The customer must think in the one small piece at a time mindset in order to successfully move forward.

All parties should come to the workshop with a basic understanding of how the customer intends to use Anaplan. Typically, this first, quick win demonstrates significant value, a glaring pain point, or utmost strategic importance. The workshop begins from this shared knowledge and moves through these steps:

1. Begin by white boarding the entire process work-flow Anaplan will address, start to finish.

2. Next, identify which area(s) of the process work-flow Anaplan will handle. Many times the customer wants to do it all in Anaplan. That's fantastic! Before making that ambition a part of a plan, remember: Anaplan implementations fare best when they focus on one or two areas at a time. Plan other process areas for future releases. Keep it simple: achieve success in a small, specific area, and build on it.

3. Once focus areas are defined, discuss other (non-Anaplan) areas and gain agreement on how they will be handled:

    a. In other software
    b. In other systems
    c. Completed manually.

    Clearly define process touchpoints and integration points. Identify and document the processes scoped for future Anaplan releases, building an Anaplan roadmap beyond the first release.

4. Once the Anaplan pieces are agreed upon, discuss and align on:

    A. Shared dimensions
    B. Data inputs and outputs flow charts, including data sources and destinations
    C. Calculations needed
    D. Expected end user experience (what people need to see)

Who should be invited?

- Business process owners who understand the end-to-end process
- End users who are responsible for their individual in-scope process components
- Data specialists who understand the data inputs, calculations, and data outputs
- Core project team

This workshop will likely take a full day. Larger projects or complex processes can take much longer to work out so make sure to account for the time when setting customer expectations.

The result of this workshop session is documentation outlining the customer ecosystem and Anaplan's role in it. It will take you a day or two to put this together. The next step is to share the documentation with the team. Be sure to follow up with the team to confirm your understanding.

Data and Data Integration

Data integration, when handled well, helps to ensure a smooth implementation. But if the customer fails to plan properly or doesn't provide resources to work on clean up, data can stop a project in its tracks.

Make no assumptions about data. Data typically needs a significant time investment on an initial Anaplan implementation. Optimize your data time investment and minimize risk with good planning, dedicated data resources, and careful foresight.

As with any change, data changes often expose underlying assumptions or gaps in a customer's data. Project teams must think critically about and candidly discuss the dataset that moves in and out of Anaplan. Involve data specialists as well as end users early on in data discussions. Help stakeholders understand that end users will see incomplete or inaccurate data as a software failure, not as a symptom of poor data quality.

Data readiness best practices include:

- Start the data discussion with key customer stakeholders before the project begins.
- Clearly identify data sources and data components (lists, properties, hierarchies, subsets, and transactional data) in the statement of work.
- Assign your customer homework: have the customer's team ensure production-quality data is ready and available at the start of the project. Gain early insight into data completeness, quality, and integrity. Build contingencies and identify critical risks and dependencies during the planning phase.

Think of yourself as a 'data detective' who needs to check your facts with multiple witnesses. Align all customer stakeholders so they agree on the source, quality, and availability of data. Since functions tend to exist in silos in a business, especially in larger organizations, it is not uncommon for an IT team to be unaware of data quality issues. Usually, a business analyst realizes the need for manual massaging and manipulation of data to make it useable. Validating assumptions across IT and business functions will help build a complete, accurate picture of the data.

Moving data in and out

As you prepare for moving data in and out of Anaplan, understand the primary data input and output methods:

Manual upload/download

In the Anaplan platform, simply pick a text or comma-separated-value (csv) file from your desktop (or other location) and import it into the Anaplan model. There are multiple export options to make it easy for other systems to receive Anaplan data.

Semi-Automated Using Anaplan Connect

Anaplan offers a way to automate the import and export of data using a proprietary tool called Anaplan Connect. Anaplan Connect is a small-footprint model that can control the importing and exporting of data from the Anaplan platform.

Fully Automated Using Anaplan Connect & API

Anaplan offers a way to automate the entire process of importing and exporting data using a cloud API which can also extend to other cloud APIs compliant with Java, ODBC, or JDBC.

ETL Vendor Integration

Anaplan offers several integration points with ETL vendors such as;

- Anaplan Connector for SnapLogic
- Anaplan Connector for Dell Boomi
- Anaplan Connector for MuleSoft

Third-party Data Integration

Anaplan also connects with third-party services, either directly or through an Integration Platform as a Service (iPaas):

- Anaplan Connector for Boomi
- Anaplan Connector for MuleSoft
- Anaplan SnapLogic Integration

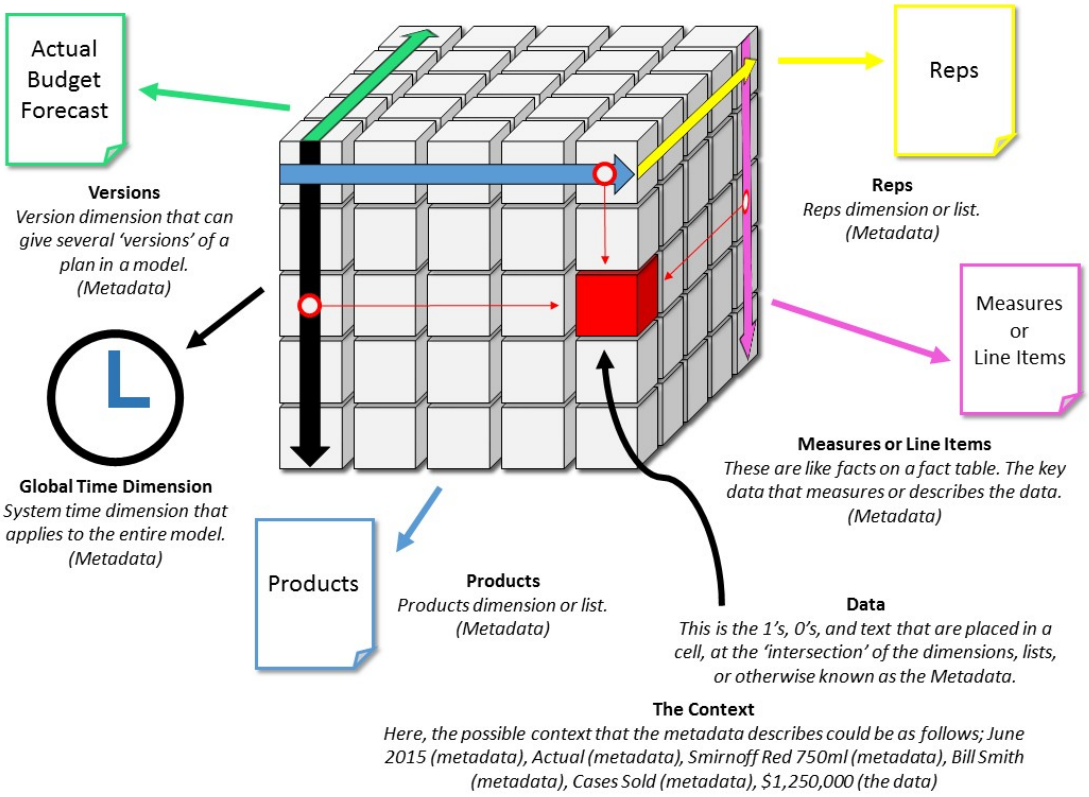
You can find more information on these options on Anaplan.community.com

A Brief Overview of the Data Needed

Consider how Anaplan defines data:

- Data is simply numbers and text; the content of a cell in Excel. For example, a cell might contain the number $1,250,000.

- Metadata describes the data in context. If you see the figure $1,250,000 in a cell, it is just that – a figure in a cell. On its own, it lacks meaning and context. You need to know what that number represents for it to be useful. For example, it may be that $1,250,000 represents the value of cases sold of ACME Red Vodka 750ml, in the month of June of the current year, in London. The description and context (measure of product sold, time, location) for the dollar amount is the metadata. Without the right metadata to provide context, data is meaningless.



Defining Quality Data

How does data go from simple to complex? How does data become hours upon hours of work? Anaplan has identified two key challenges:

1. As organizations grow, so do their systems - mainly in silos: one for managing the customer relationship (CRM), one for managing the workforce (HRIS), one for accounting (ERP), one for projects, and so on.

2. Over time, these systems contain different data -- but usually, they contain the same metadata, which becomes disconnected. Customers are a great example of how shared metadata disconnects across company systems. A customer called "Acme" in one system is called "Acme Co." in another system, and "Acme Company" in yet another. Data for Acme exists in the ERP system; opportunity data exists for Acme in the CRM system, and marketing data for Acme exists in the marketing system. Systematically it's difficult to match up systems, as the link between the data has been disconnected.

Anaplan's platform can help solve this problem. Anaplan models bring these items together to give a complete view of the business so the company can analyze and plan ahead.

When embarking on an Anaplan implementation, forewarned is forearmed. Decide where you are right from the start and take necessary steps to fix issues. Data challenges may not be resolved overnight, but the right level of transparency allows the customer to put the right plan and the right people in place to make data preparation a priority.

Data Integration: Consider These Factors

There are many ways to get data in and out of Anaplan; some more involved than others. In our experience, starting simple is always the best approach, unless the customer has a large, dedicated data team. In addition, as previously mentioned, if the data needs cleaning in some way, recommend to the customer that it is best to start the data effort right from the beginning, or even before the project gets started. Suggest that the customer bring in expert help if that makes sense. Many organizations rely on external expertise to supplement the core team and many also bring in special data cleansing tools that help reconcile differences between systems. Here are some guidelines to keep in mind as you evaluate the steps you will take to manage and prepare the data for the integration process:

- Assign overall accountability of the data work stream to a member of the customer team; ensure the customer holds the data work stream owner accountable throughout the implementation.
- Start small and focus on data quality.
- Automate later. It is best to begin with manual uploads. The fact that data and metadata loads are not automated will not stop the project. Poor data can.
- Add data tasks as user stories in the Agile Implementation – The Anaplan Way app.
- Pay as much attention to data as you do to building the model.
- When you uncover data issues, document them clearly and socialize them broadly. It is important that the entire project team, including executive sponsor, understands the issues with upstream data. It is much better for people to have a clear understanding of the precise and specific data issues, than to say "the data is terrible, so the model won't work".

Model Building

This cornerstone involves the basic building of an Anaplan model. This cornerstone also includes considering:

- Customer training on model building
- Model building knowledge and experience
- Model design

Customer Training on Model Building

Customer model building knowledge is critical to the implementation process. Customers should complete Launchpad training before requirements gathering begins. Model building training:

- Provides a baseline: Everyone on the team should understand Anaplan's capabilities and how the platform can be used to solve business problems.
- Drives efficient requirements gathering: Understanding the art-of-the-possible (what can be achieved through model building) helps ground users in Anaplan's functionality, which in turn helps them understand how business requirements might best be realized in a model.
- Provides context that leads to increased stakeholder involvement: When stakeholders understand Anaplan's capabilities and functionality, it is easier for them to make educated decisions about issues that arise, instead of avoiding or delegating decision-making.

Anaplan recommends all project teams have two or more trained model builders on staff in order to assist with the model build, help create knowledge transfer to end users, and effectively maintain Anaplan models one they are generally available.

Model Building Knowledge and Experience

Anaplan project teams should structure their model building team around Anaplan certified resources (including certified partners), and leverage the entire team's expertise, especially in initial releases. Customers must ensure their model building resources have adequate knowledge and experience, and are authorized to build on the Anaplan platform.

Model Design

Successful model builds rely on strong model design. *Measure twice, cut once* is something you've probably heard before. In this instance, it means the model has been designed as an integral part of the business process where it resides instead of as a stand-alone solution.

Deployment

When you think of deployment for a typical project, it involves a plan to roll the application out to the end users and will usually include a training approach for getting the end users trained. Deployment is a cornerstone because it is so much more than this! Deployment must be considered right from the start, by designing an elegant solution for the end user. If you fail to capture the appreciation of the end user, you will lose their buy-in and ultimately lose adoption of Anaplan and the new process, effectively negating any ROI for the customer.

Involve end users in the Anaplan implementation project at the onset -- at the latest, during the second sprint. Customers should identify champions who can help promote the Anaplan model when it becomes generally available.

Many times, end users may be used in early 'sneak-peeks" at the application and the User Acceptance Testing (UAT). Getting early buy-in to the design is always a good idea for a successful deployment. In addition, work with your customer to ensure the most influential people in the user population are involved in the process early on. Encourage the customer to let these influential people own some of the decisions and participate in the design. This is a great way to get early buy-in and these people will then act as advocates to the wider user population once the application is in UAT and during the go live phase.

Deployment, and the run up to deployment, is effectively a sales and PR job. Keep in mind that your customer is selling the application to their end users. Remember that it is almost impossible to over-communicate that a process change and new tool are being implemented.

Work with your customer to involve business subject matter experts (SME's), who are well respected by the business in the early stages, preferably in the requirements phase. Then, around the middle of Sprint two, once you have the model at about 90% built (model only, no dashboards), start to get them involved in evaluating the solution.

What this means is three-fold:

- You get their early buy-in involvement in the project
- You start to force joint ownership of the solution
- You have time to steer the ship if things need to be adjusted

While this buy-in and evaluation of the solution is happening and tweaks are being made, you need to prepare for how you will introduce the business to the new solution and how people will be trained.

The next section is customer-focused. Work with the customer to make recommendations regarding how to effectively sell the Anaplan solution to the business. Remember that an effective deployment often leads to expands within an organization. A deployment that does not get off the ground leaves the customer wondering why they invested in Anaplan.

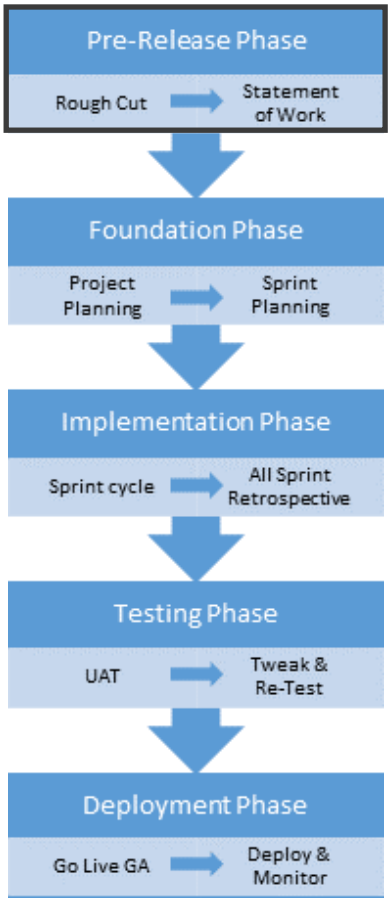
Positioning Anaplan to the Business

- Provide an early introductory on-demand video for end users to view that answers these questions:
    - What and who is Anaplan?
    - What process are we changing?
    - Why are we changing?
    - How does it work (demo)?
    - What is the timeline for completion?
- Provide a software simulation for users to see what it will be like to perform the process.
- Create a web site where anyone can browse topics and documents.
- Conduct regular webinars or lunch-and-learns for users to attend and ask questions; openly discuss the impact of the new process on the business and their jobs.
- Conduct on-site evaluations and feedback sessions in which users see a presentation from the project team, share comments, and generate ideas.
- Put yourself in the place of an end user and answer:
    - Why are you changing the way I am doing this process?
    - How will it change the way I do things in the future?
    - What was wrong with the old way?
    - How will this make me do my job better or make me more efficient?
    - When does the project start? How will I be involved?
    - What if I get stuck or do not know what to do?

Create a Change Communication Plan

- During project planning, create a plan to introduce the business to the Anaplan model and process. Adjust this plan throughout the release phases.
    - Create a training plan for end users. Include why the change is important to the end user, what the change includes, how their work may change due to the change, and where they can go for more information or assistance.
    - Use documentation to create job aids on process flows or changes.
    - Evaluate how many end users need to be trained; how many resources are needed to reach them all effectively.
    - Identify two or three key success measures: how will you know end users are effectively adopting the model and process?
- Deployment is change management – a critical promotional role. Anaplan teams must sell the greater business on the Anaplan model in order for the business to adopt both the Anaplan model and resulting process changes. Plan layers of communication as part of the deployment plan.
- Over-communicate that a process change and new tool are being implemented.
- Find ways to effectively position the Anaplan model to the business. An effective deployment often leads to additional Anaplan implementations.

3. Pre-Release Phase



The Pre-Release phase of an Anaplan implementation begins when a customer asks for an estimate. During the pre-release phase, all a customer requires is rough estimate. The customer completes a form and may participate in a short scoping session and Anaplan uses that information to prepare a Rough Cut estimate.

When the customer decides to move forward, Anaplan holds a full scoping meeting and ask a lot of questions. Anaplan uses scoping information to prepare a Statement of Work (SOW). Once the SOW is signed, the project is added to the implementation project schedule.

The Rough Cut and SOW set the tone for the Anaplan-customer relationship. In this chapter, learn how to create both Rough Cut and SOW documents.

Tasks:	Deliverables:	Tools:
Rough cut	Estimate for customer	Rough Cut Estimation model
Scoping Meeting	SOW, including: ▪ High level model design ▪ To-be business process overview	Scoping Deck Project Scoping and Estimation Model

Rough Cut

When to Use a Rough Cut

The Rough Cut process provides a customer with an initial estimate of Anaplan cost and timeline before signing a Statement of Work. The estimate provides a price range for the implementation of the Anaplan platform.

Details	Rough Cut
What is it	Rough estimate referencing similar projects and project teams
When and where	Session between 1 hour and 1 day Onsite preferred, remote ok
Who attends	Anaplan: AE, Pre-sales, BP/CSD Prospect: project sponsor, IT, SME/Process Owner
Deliverables	Rough Estimate presentation from template with a proposed estimated range
Tools	Customer Success Pitch Deck Scoping Questionnaire Benchmarking Tool
Process	Based on a short session with prospective project team
Turn-around time	Two business days

Preparing a Rough Cut

Prepare a rough cut estimate using the official Anaplan Rough Cut template:

- First, gather information.
    - What is the presenting business problem?
    - What should the project accomplish?
- Next, determine if Anaplan will recommend a single phase or a multi-phased implementation.
    - What is the typical project length for this type of implementation and project team?
    - Based on comparable data, decide on recommending a single or multi-phase approach.
    - Create an estimated project timeline based on the approach.
- Determine the project team's complexity and effort drivers. Drivers range from low to high. Enter the drivers the Anaplan Rough Cut model to generate a cost range:
    - User base
    - Resources
    - Data reconciliation
    - Metadata and model
    - Reporting
    - Data integration
    - Training and change
- Customize the Anaplan Services Proposal presentation with this information.

Scoping the Project

Defining the scope of the implementation project is part art, part science.

Consider the state of a customer's business when they decide to embark on the Anaplan journey. The customer has decided to change, or at least improve, a key process. The customer has also decided to change the application that enables the process.

A single change effort brings a single set of unknowns. Simultaneous changes to both process and platform increase the set of unknowns by an order of magnitude.

The Statement of Work (SOW) is a letter of intent. At the point in time it is written, the SOW defines:

- Project-specific activities
- Project deliverables
- Project timeline
- Business requirements
- Pricing

Think of the Statement of Work as a starting point where customer expectations are established. You will manage those expectations in alignment with the scope defined in the SOW.

A Statement of Work requires significant, detailed scope information. To create a SOW, gather scoping information by meeting with a customer, asking questions and using tools to help estimate:

- The size of the model (memory requirement or size in memory)
- Scope of work (business requirements)
- Level of effort to design and build the model

Model Size

Model size determines important factors such as:

- The model topology (whether it is a single or distributed model)
- The technical infrastructure
- And in some cases, the cost of the model subscription

In a multi-dimensional environment such as Anaplan, each cell represents a piece of data, whether or not it contains data. Empty cells and sparsity are handled well in the Anaplan platform; however, count each cell when sizing. Each cell is approximately 10 bytes of data. From this calculation, determine the MB and GB model size.

Use the Agile Implementation – The Anaplan Way App to determine size; but first, understand how size is determined. Understanding size helps you realize if something is amiss when working in a model.

In the following illustration, you'll see how model sizing calculators work. Anaplan has calculators for each type of use case, as dimension intersections play a large role in determining size. Common use cases include:

- Territory and Quota Planning
- Sales Planning
- Financial Planning
- Workforce Planning

Our example looks at three modules from a Territory and Quota Planning model. We calculate size by looking at the major dimensions and then multiplying to get to a cell count.

Module 1: Sales Forecast

Dimensions	Line Items
Time Versions: actual, budget & forecast Products: 2000 items in the list Sales Organization: 2500 people in the list	Units Price Total

SALES FORECAST													
	J	F	M	A	M	J	J	A	S	O	N	D	FY
Units	3	12	78	43	97	34	89	56	98	29	55	23	617
Price	$23.99	$23.99	$23.99	$23.99	$23.99	$23.99	$23.99	$23.99	$23.99	$23.99	$23.99	$23.99	
Total	71.97	287.88	1871.22	1031.57	2327.03	815.66	2135.11	1343.44	2351.02	695.71	1319.45	551.77	

The number of cells for each row is 13: one for each month and a column for the fiscal year.

Module 2: Expense Forecast

Dimensions	Line Items
Time Versions: actual, budget & forecast Sales Organization: 2500 people in the list	Airfare Meals Entertainment Total

EXPENSE PLANNING													
	J	F	M	A	M	J	J	A	S	O	N	D	FY
Air	$2,879	$1,278	$3,258	$4,300	$1,230	$2,100	$0	$2,340	$0	$5,400	$1,780	$2,200	26,765
Meals	$200	$400	$350	$300	$450	$300	$0	$560	$0	$670	$250	$350	3,830
Ent.	$300	$300	$300	$300	$300	$300	$300	$300	$300	$300	$300	$300	3,600
Total	3,379	511,200	1,140,300	1,290,000	553,500	630,000	0	1,310,400	0	3,618,000	445,000	770,000	1,359,779

Module 3: Income Statement

Dimensions	Line Items
Time Versions: actual, budget & forecast	Sales Expenses Total

INCOME STATEMENT													
	J	F	M	A	M	J	J	A	S	O	N	D	FY
Bookings	$72	$288	$1,871	$1,032	$2,327	$816	$2,135	$1,343	$2,351	$696	$1,319	$552	14,802
Expenses	$3,379	$511,200	$1,140,300	$1,290,000	$553,500	$630,000	$0	$1,310,400	$0	$3,618,000	$445,000	$770,000	10,271,779
Total	3,451	147,225,600	2,133,501,300	1,331,280,000	1,287,994,500	514,080,000	0	1,759,867,200	0	2,518,128,000	586,955,000	425,040,000	2,032,072,051

To calculate the size of the model, multiply the numbers across and add up the column.

Module	Time	Product	Sales Org	Line Items	Versions	Cells	Bytes	MB
Sales Forecast	13	2000	2500	3	3	585,000,000	5,850,000,000	5,578
Expense Planning	13	0	2500	4	3	390,000	3,900,000	37
Income Statement	13	0	0	3	3	117	1,170	0.0011
							5,853,901,170	5,615

User Base & Concurrency Estimates

In addition to calculating the size of the model, it is necessary to be aware of the user base. Who will be accessing the model and how often? Having an accurate estimate of how many users will be using Anaplan at the same time is critical to building the model to meet performance expectations. The table below is an example of calculating concurrency based on three business units within three main regions of an organization for a model.

Business Unit	AMS	APJ	EMEA	TOTAL
1	.10 x 2,470 = 247	.10 x 2,070 = 207	.10 x 3,700 = 370	824
2	.10 x 810 = 81	.10 x 890 = 89	.10 X 700 = 70	240
3	.10 x 350 = 35	.10 x 320 = 32	.10 x 500 = 50	117
TOTAL	363	328	490	1181

This graph represents 10% of total user population using Anaplan concurrently. This is a higher than average number of concurrent users. A distributed model may be required.

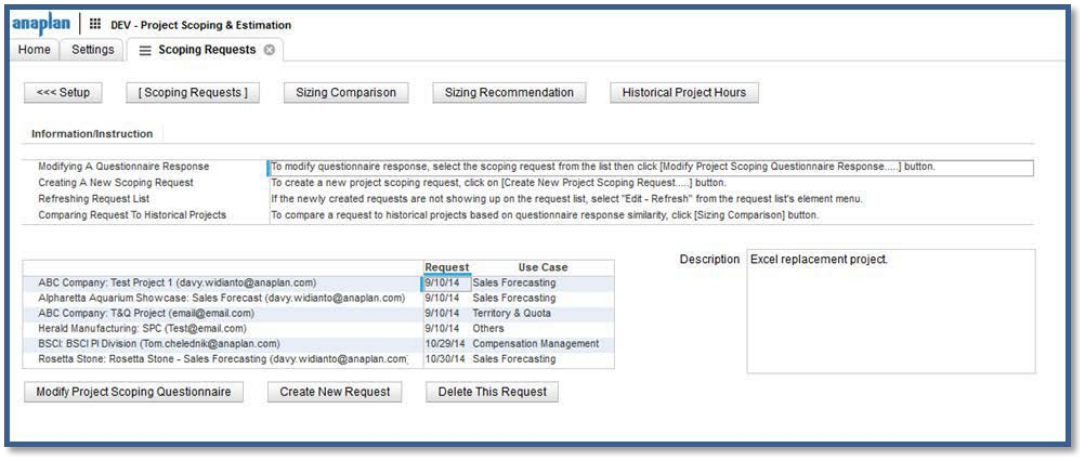
Scope of Work

Scoping focuses on obtaining the right business requirements so the rest of the implementation is successful.

After estimating model size and potential user concurrency, gather the business requirements to fully understand the scope of work. Plan two to three days to a maximum of one week to gather a complete set of requirements. Once you have requirements defined, estimate the effort needed to get the model built in the time required.

Project Scoping and Estimation Model

Anaplan has completed many successful implementations over the years. The Project Scoping and Estimation model includes historic project data and logic that helps facilitate the scoping process and produces accurate scope estimations for new projects.



Start on the Scoping Request dashboard, which creates a scoping request comparing the new project with the scope of a past project. Start by selecting a use case, then answer a series of predefined questions. Answers are scored and weighted. Use the results to determine how similar the new project is to past projects.

Project Pricing

The SOW covers the estimated project charge. The estimated project charge outlines services for which the customer will be invoiced:

- Expected number of hours required of the Anaplan team to complete the project
- Number of Anaplan (and possibly partner) resources assigned to the project
- Rate for each assigned role
- Training costs
- Reasonable out of pocket travel and related expenses

The official fee is specified and all payment terms are governed by the signed SOW.

Project Timing

The Rough Cut or SOW includes a project timeline. Here is an example:

Project Milestone	Timeframe
Project Start	Jan 07
Requirements Phase	Jan 07 to Feb 01
Sprint 1	Feb 04 to Feb 22
Sprint 2	Feb 25 to Mar 15
Sprint 3	Mar 18 to Mar 29
Sprint 4	Apr 08 to Apr 26
UAT Phase	April 29 to May 17
Tweak Phase	May 20 to Jun 07
Production/Go Live	June 07

The Statement of Work

The Statement of Work (SOW) is a statement of intent. It is not an exhaustive document that outlines every single task, story, and item that will go into a customer's solution. Neither the customer nor Anaplan can possibly anticipate the final solution. Help customers understand the SOW as a statement of intent – which is likely different from past experience. Anaplan does not delineate every detail in the Pre-Release phase because seldom does end product match scope exactly. Use the Managing the Buckets whiteboard to educate and gain buy-in.

Project Scope

The Project Scope section includes several paragraphs describing:

- Current state overview
- Project objectives
- A high level drawing of the customer ecosystem, with the Anaplan solution identified
- Know unknowns

The Known Unknowns

No matter how project requirements are defined or how a project compares to others, every project is going to have its share of surprises. Document both knowns and unknowns:

- Resources for the project
- Data and meta data
- Process surrounding the model

Acknowledge those things that you just don't know, that may have a material effect on the release.

What is an example of a known unknown? Perhaps you don't know what shape the metadata is in. That potentially could be a problem, but at this point you don't know. Because you don't know, it is difficult to determine just how much time and resources clean-up of the metadata might take. The clean-up is part of the project, but you can't say exactly how it will affect the project. Define this as a known unknown that will have a material effect on the project.

How do we manage known unknowns?

- Everyone should acknowledge them.
- Everyone needs to think about a contingency, so that when you understand what the known unknown involves, you've budgeted for it. This helps the customer handle bumps in the road.
- Prepare a list of known unknowns and document them in the SOW. Ensure the whole team understands and agrees with the list.
- In some situations, a project cannot begin with the known unknowns. For example, a customer may not know what SMEs they can provide. These problems indicate unknowns that must be known before the project starts.

Project Timeline

Add the timeline from a completed Rough Cut (if you have one) to the Project Timeline section of the SOW. In the absence of a Rough Cut, create a timeline that includes:

- Project kickoff
- Planning
- Model build and execution
- Testing
- Parallel support and go live

Project Deliverables

This section of the SOW includes a list of deliverables, by phase, and who (Anaplan or client) is responsible for each deliverable. For example, deliverables in the Foundation Phase include:

- ETL Requirements – Client responsibility
- User stories and Acceptance Criteria –Client responsibility
- Model Design schema – Anaplan responsibility

The deliverables should not be edited. If there is a deliverable that isn't applicable for the project, include NA after the item. Discuss the list with the customer to help set expectations.

Project Team

Complete the roster with the names and roles of the Anaplan/Partner resources as well as the customer resources and their roles. If a rough cut estimate was completed, include the resource page from the deck in this section.

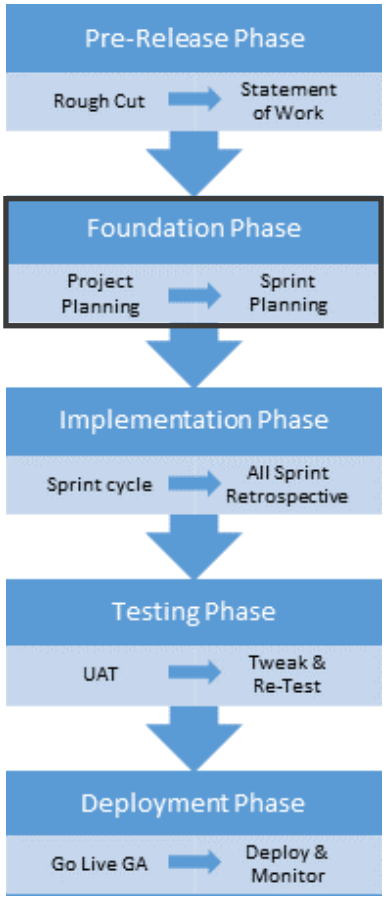
Other Project Considerations

Enter all scoping information into the Customer Success SOW tool. The tool calculates an estimate of hours required to successfully implement the platform. The tool also determines a final cost estimate. This section includes a screen shot of the Price Table in the Customer Success SOW tool and the total cost estimate, which is the dollar amount of the number of hours required times the rate for each different role, plus travel and expenses.

The SOW also includes these standard sections:

- Assumptions (Cost, Billing Schedule, and Payment Terms)
- Change Management Process

4. Foundation Phase



Complete two major areas of planning during this phase:

- Project Planning
- Sprint Planning

Project planning includes all the tasks which get the project underway, for example; training and holding the kick-off meeting.

Sprint planning includes requirements gathering and scheduling project work into sprints.

Tasks:	Deliverables:	Tools:
Sales to delivery hand-off meeting	N/A	Coming soon!
Create Scrum team	Scrum team identified and training scheduled	N/A
Launchpad	▪ Trained model builders	N/A
Kick-off	▪ Kick-off meeting ▪ Model Manifesto ▪ Verify training	Coming soon!
Verify high level schema of customer ecosystem	N/A	SOW
Initial Requirements Workshop	▪ Workshop document	N/A
User Stories: ▪ Write user stories ▪ Size user stories ▪ Sign-off on user stories	▪ User stories completed ▪ User stories estimated	Agile Implementation –The Anaplan Way app
Project Planning	▪ Plan for how to handle the cornerstones	Agile Implementation –The Anaplan Way app
Sprint Planning	▪ User stories allocated to sprints	Agile Implementation –The Anaplan Way app
Process definition	▪ Wireframes or prototypes of dashboards	Balsamiq or other tool
Model design	▪ Model Design schema ▪ Dashboard / end user experience design / prototype	Lucid charts or other tool
Data Integration	▪ Data Flow diagram	Lucid charts or other tool
Change management	▪ Change Management Plan – work with the customer to develop	

Set Expectations

Set mutually agreed upon expectations for everyone involved in the project during the Foundation phase. Document feedback and state ambiguous areas in the SOW. Clarify outstanding issues, project risks, or contingencies in the SOW prior to sign off.

With a strong SOW in place, aligned expectations begin with understanding what the customer has been promised. When Anaplan makes a commitment to a customer, we have an opportunity to meet customer expectations. As the Business Partner responsible for the implementation of Anaplan, understand the power of commitments to a customer. Clarify Anaplan's commitments; be careful not to promise something Anaplan cannot deliver.

The next sections include expectations for each phase going forward.

Foundation Phase

Measure twice and cut once: memorize this phrase as you set a project's foundation. Project planning and sprint planning take place in the Foundation phase. A strong foundation contributes to overall release success. Help the customer understand that the information provided during this phase must be as accurate as possible to avoid issues later. For example, if the customer commits to having two model builders that can work 20 hours per week on the project and they only work ten hours, the project timeline will slip.

Don't forget the cornerstones when building the project plan in the Agile Implementation – The Anaplan Way app. Build activities around model, data, process, and deployment to address each cornerstone and ensure that you meet customer expectations.

Use Planning Poker as your sprint planning exercise. Customers may or may not have experienced this type of planning. Manage expectations by explaining the process in advance and highlighting the benefits:

- Build consensus on time it will take to build a user story
- Build consensus on the priority of the user story
- Determine the necessary resources to build those user stories with highest priority

Manage expectations by agreeing with the customer on how to manage the buckets. Managing the Buckets provides a process for ensuring that when new user stories are added during sprints, other user stories are shifted to different sprints or removed to keep sprints balanced. See *Managing the Buckets* later in this chapter.

Implementation Phase

This phase includes building the model in sprint cycles. During the first sprint, customers can feel a bit nervous, especially if the customer has not used an Agile methodology. Reassure the customer that nervousness is normal and to trust the system. Agile differs from most other approaches because at the end of sprint one, there will be some pieces in the model that show the structure and how things will work. It will not be perfect, but the result can be imagined. The first sprint gets end users involved to achieve buy-in needed for a successful release.

At the end of sprint two, the customer will imagine the possibilities with Anaplan, and may want to include additional requirements. Hold firm. You cannot allow the customer to add requirements willy-nilly without running the risk of over-promising and under-delivering.

During sprints three and four, data and data integration become the focus. You have completed pre-work and discussed the importance of the data earlier in the project, and the customer begins to see how important the cornerstones are.

Testing

User testing is usually a time when bugs and enhancements come to light. The customer may call things defects that are actually enhancements. Carefully consider these requests; capture the enhancement ideas for another release. Create a triage committee to sort through UAT feedback. Be careful about over-promising. Do not add to the model during UAT; instead, ensure the model is robust and stable.

The model must meet the customer's usability expectations. Usability includes performance: wait times must be kept to the minimum described that you and the customer have agreed to. (See the *Performance Discussion and Agreement* section in Chapter 5 Implementation Phase.) If human testing shows that a certain task or item is misunderstood, work with users to adapt the dashboard so it works smoothly for them. The adoption of Anaplan is at stake: happy users make a satisfied project team.

Deployment

The deployment phase includes end user training, communication planning, and change management. Since deployment is one of the cornerstones, address deployment plans well in advance.

Consider one final expectation. As your customer deploys the Anaplan model, they will collect user feedback for use in the next release. Sometimes customers may have experienced a challenging rollout and may sense risk in gathering feedback: What if it comes back negative? Remind the customer of the process: from the beginning, you have not set out to create something perfect. Continue to improve upon what has been built; the best way to improve is to collect user feedback.

Project Planning

The project planning activities included here ensure the project starts in the right direction. Remember the saying: measure twice, cut once. During project planning, measure twice. Project planning includes these activities:

- Sales to delivery handoff meeting
- Customer training
- Facilitating the Project Kick-off Meeting
- Working with the customer to create a Manifesto or goal statement
- Working with the customer to select the Scrum team
- Working with the Scrum team to establish team ground rules
- Planning for a Center of Excellence or Competency Center

Sales to Delivery Handoff Meeting

The Sales to Delivery Handoff meeting occurs once the customer has signed a contract. In this meeting, the sales team provides the customer success team with details regarding the customer's requirements. The sales team provides sample data files or demo builds created during the sales cycle.

Week Zero/Launch Pad/Training

First focus on customer training. Getting contributors ready for Anaplan helps create a collaborative and successful project. As part of the deployment cornerstone, ensure that at least two people at the customer achieve Official Anaplanner status.

Why does training come first?

Customers must understand key Anaplan platform concepts before working on requirements gathering. Customers who complete the training understand how to define and use dimensions, lists, modules and dashboards.

Project Kick-Off Meeting

Introduce the customer to The Anaplan Way methodology and review project objectives during project kick off. The executive sponsor typically presents an overview of the company, and the business objectives the project addresses. In addition to the executive sponsor, include an influential IT leader in this meeting.

Objectives of this meeting include:

- Review the executive sponsor's overview of organization and business objectives
- Agree on project schedule and responsibilities – focused on planning
- Introduce team members
- Discuss and align on customer resources needed
- Review the timeline that was included in the SOW
- Check in to ensure training has been completed
- Revisit data preparation activities
- Orient the customer to The Anaplan Way Methodology
- Introduce user stories and sprint planning

After this meeting, work to keep the momentum going. Consider regularly scheduled steering committee meetings to keep executives involved and informed. Executive involvement elicits buy-in and raises project visibility.

Think about how you involve IT. Can they be used as the first line of user support? If so, IT will need to learn the Anaplan platform and dedicate time and resources to support it. In some cases, IT would be the best group to support a data hub, and may be eager to move other lines of business to Anaplan. IT could become the driving force for expands, as using Anaplan could significantly reduce the number of software applications supported.

The Manifesto

Manifestos state what the customer intends to build.

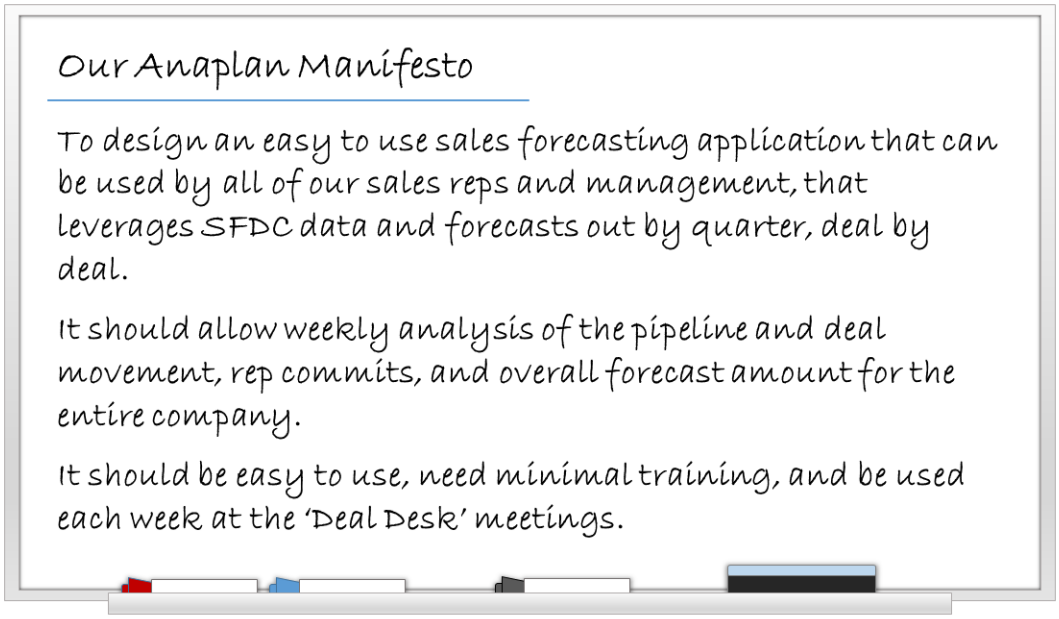
Creating a Manifesto:

- Gather key stakeholders in a room
- Key stakeholders craft a paragraph or two that describes exactly what the team will build.
- When the project is at the go live date, you should be able to read your manifesto and say yes, the model we built aligns completely with the manifesto.

The manifesto should be clear, concise, and focused on the overall project goal. Creating a manifesto early in the project ensures that you have agreement and buy in among stakeholders.

The manifesto acts not only as a checkpoint at the end, but also as document referred to regularly during the release. Use the manifesto as a checkpoint to ensure the project is going in the right direction. Return to the manifesto and determine if current work achieves the manifesto. If not, determine what needs to be done to get the project back on track.

This image represents a simple sales forecasting manifesto.



Tips for Writing a Manifesto

Most customers like the idea of a manifesto, but may object to writing one. Help customers understand that a manifesto acts as a tool which ensures the project remains on track.

- To get people thinking about the manifesto, ask: What are we trying to accomplish? and/or How do we achieve success for this project?
- Ask the customer how he or she will know the project has been a success. Write down his or her thoughts and then turn it over to for polishing.
- Refer back to notes taken during the sales cycle and shared during the sales to delivery hand-off. Often the customer has shared his/her vision with the sales team; this can be a good start to a manifesto.

Project Resourcing

While thinking about who is going to help with the model build and determining which resources will be assigned to the project, keep in mind the customer also must consider resource needs. The customer's resource needs will typically extend beyond the life of the project and it is important that the customer has thought about and made decisions around the following questions:

- Are the people on the team expected to be on the project full-time or just part-time?
- Select the correct key role of project sponsor.
  - The project sponsor understands the overall process and the system landscape, knows the different types of users, and understands their roles in the process.
  - The project sponsor has a vision for what needs to be built and will support the business in defining the requirements (user stories).
  - The project sponsor is ultimately responsible for the prioritization of the user stories and the successful delivery of the model.
  - This role requires an average of two to three hours per day for the duration of the project.
  - Who will take on this role and will this person be able to dedicate this amount of time?
- Determine the hours needed each week to:
  - Get the project up, running, and ready to go live
  - Manage the platform post go live
  - Extend the platform to connected up-stream and down-stream processes
  - Run through another life cycle (processes may require design tweaks and changes as the business changes)
- Determine required effort on the four cornerstones:
  - Process
  - Data
  - Model
  - Deployment
- Build time on the Anaplan model is usually small compared with data and metadata work, process changes, and the deployment.
  - Should I be looking to involve a partner to supplement my efforts, especially around process change and deployment?
- Are the skill sets of those involved on the project the right skill sets?
- Are the resources actually available for the proposed time line?
- What is the customer's resource commitment? Are they able to provide dedicated resources to the project to assist in the build? Note: Care should be taken not to put too much dependence on customer resources as this often changes.
- Have all the resources involved bought in to the process and the decision to choose Anaplan?

Selection of Scrum Team/Customer Resources

While the customer is ultimately responsible for selecting the resources to assign to the project, be involved in that process. These roles will not only have project responsibilities, but may smooth or hinder the organization's Anaplan deployment.

Ideally, people selected to fill these roles should have most or all of these characteristics:

- Clear vision. This person should have a clear vision of what project success means for the organization, and the ability to clearly communicate that vision. This needs to be consistent, as someone who changes his or her mind frequently or seems to be all over the place can scare people off and make them unsure.

- Patient, yet persistent. Every project can be frustrating and seems as though no progress is being made. Someone who understands that good things take time and the importance of following a process will probably not share their frustrations or "the sky is falling'-type concerns with co-workers. They may feel that way, but understand that sharing their frustration will not help and may hinder the project's progress.

- Asks tough questions. The person who asks tough questions has some skin in the game – he or she has bought in to the solution and asks good questions to ensure that the solution is really the best for the organization.

- Knowledgeable. This person understands the business and the process and is sought out by coworkers who need help. He or she is able to provide an explanation at the correct level of detail.

- Strong relationships. This person is approachable and reliable. People in the organization trust this person.

- Positive, but realistic perspective. This person is good at seeing the big picture and what the new process can mean to the organization, but is not simply a yes-person. He or she can spot issues or problems and makes suggestions for how to get around them. A person with a negative perspective tends to focus on the problems and doesn't offer any ideas for how to correct them.

This chart shows the Scrum team roles, approximate number of hours per week the project work requires from the role, and the role's responsibilities. Note in the chart that the roles identified in the Other Team Member section usually are not putting hours in for the entire project. Instead, their hours are for certain phases: for example, the Data SME is involved during requirements gathering (Foundation phase) and UAT (Testing phase).

Role	Hours Per Week	Responsibility
Project Sponsor	10	With primary responsibility for maximizing return on investment (ROI), the customer-provided project sponsor focuses on leadership needs and is an expert on current processes and upcoming changes in any process. This person is engaged in all phases of requirements and sprint reviews and these key areas: • Guiding the vision and re-prioritizing long-term expectations such as release date, content, and backlog. • Final arbiter of user stories (requirements); accepts or rejects each sprint. • Considers stakeholder interests at all times and may contribute as a Scrum team member when practical.
Scrum Master	15	Like the project sponsor, the Scrum master will come from the customer side and have a leadership role. Unlike that role, this role does not have management authority over the team. This person runs Agile, leads daily meetings, tracks user stories, and updates the backlog. Primary responsibilities are: • Facilitating the Scrum process and resolving any impediments along the way. • Making the Scrum process effective by creating an environment conducive to team self-organization. • Capturing empirical data to adjust forecasts and shielding the team from any external interference and distractions.
Model Builders	30 +	Once the model design has been reviewed, model builders execute on the model architecture, model management and maintenance. They can be an Anaplan employee, partner, or a customer.
Other Team Members		
Data Subject Matter Expert (SME)	25	Owns data sources and arranges for edits and data scrubbing. Involved in requirements, testing, and validation phases.
UAT & Training	25	Develops engagement plan, writes UAT test scripts, develops communications, and trains users. Involved in UAT and deployment.

Take a realistic view when planning resources and sprints. Many times, people must perform their regular job responsibilities; the team must be realistic as to how many hours per day can be dedicated to Scrum activities. When the number of hours is not accurate, sprint planning will not be accurate.

Establishing Scrum Team Ground Rules

You've probably worked with many teams. Some of them have quickly gelled and worked together efficiently without a lot of drama, while others have struggled. Establish a set of ground rules to help teams make better decisions, improve working relationships, and increase team satisfaction.

Devote some time as a team to establishing ground rules. The team creates, documents, and agrees to the rules. The rules should include what is important to each team member in terms of acceptable behaviors and norms:

- Pick a common medium for collaboration (Slack, IM, etc.)
- Set a daily time place for the Scrum meeting
- Require concise and to the point communication
- When critical issues come up, discuss them in person, rather than using email.
- Never use email when angry
- Don't CC the world when emailing
- Feedback should focus on behavior
- Be respectful
- See things from both sides
- Be calm and professional at all times
- Keep commitments
- Be transparent – share all relevant information, including your thoughts, feelings, intentions

Some teams may only need a few ground rules; others may need more. All team members participate in the creation of the ground rules and agree to follow them. Once the team establishes ground rules, ask:

- Does everyone agree to follow the rules? (Important to ensure commitment to these rules)
- How will you point out violations? (This is a function of the team.)
- Where will the ground rules "live" so they can be easily accessed?
- Has everyone provided input?
- How will new members be informed of the ground rules?
- How often will we review / change these ground rules?

One question asked at every Sprint Retrospective meeting: How well is our team working? If you've established ground rules, you can ask how the ground rules are working and if there have been violations that weren't addressed. Remind the team that they are responsible for not only following the ground rules, but also pointing out when a team member is not. With ground rules in play, you won't have to act as a police officer making sure that the rules are enforced.

Center of Excellence Considerations

Anaplan has shifted strategy around the establishment of a Center of Excellence (CoE) from being only needed for large organizations to being considered for every project. Not every customer will recognize the need for a CoE at the beginning of the project. The project may be well into the sprint cycles before the customer realizes they need a CoE. Remember the customer relies on Anaplan expertise; lay the groundwork for the CoE up front. Complete the first three steps during the Foundation phase:

1. Include a data hub as part of the model design. This will make it much easier to add another model for a different use case.

2. Document the business processes. When a new use case is being considered, you'll be able to determine how it fits into the overall business process. You won't have to ask the customer to repeat the exercise of explaining the processes.

3. Document the model logic and business rules:

    a. Technical ecosystem topology
    b. Model blueprint
    c. Module blueprints
    d. Model flow (model map)

For more information, see *Chapter 9, Center of Excellence*.

Sprint Planning

During sprint planning, lay the foundation for managing the project using Agile Scrum methodology. Gather requirements, complete the model design, and use planning poker to craft the sprint schedule. Explain to the customer the proven way to keep sprint planning on track: managing the buckets.

Requirements Gathering – Writing User Stories

Writing user stories ranks at the top of critical project tasks. When the customer writes user stories, business subject matter experts (SME) sit down with the Scrum team to lay out user requirements – what the user community needs from the Anaplan platform so they can follow the business process. For example, a customer may be using the Anaplan platform to improve their sales forecasting process. The team needs to discuss all upstream and downstream processes within the sales forecasting process that Anaplan will impact.

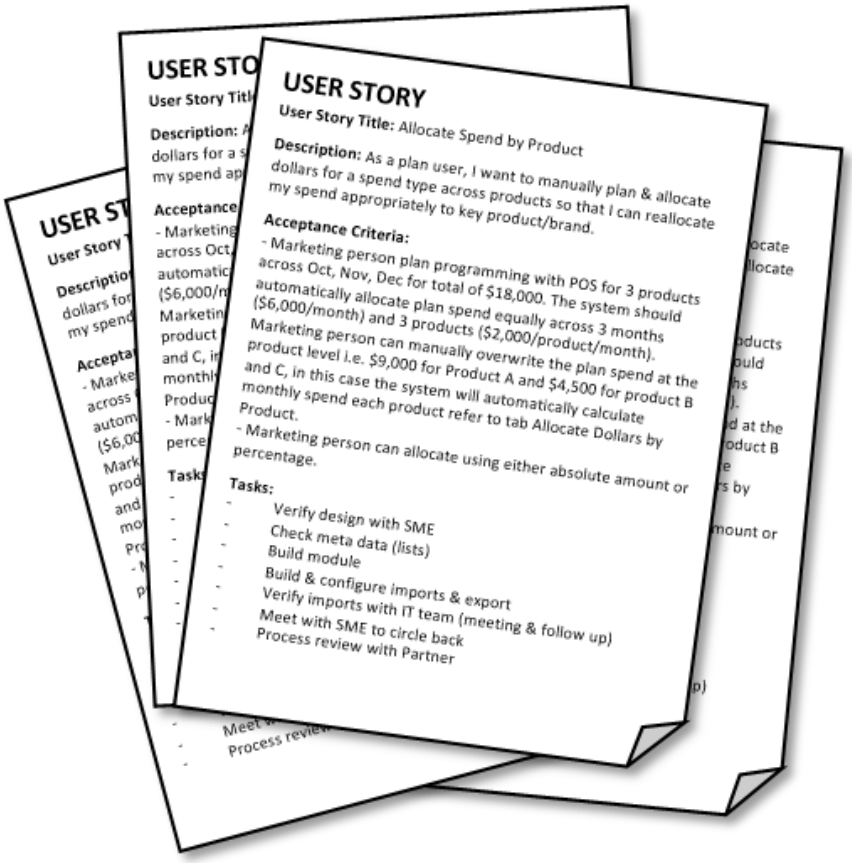
The customer documents requirements by producing user stories. A user story includes:

- Title
- Description
- Acceptance criteria
- Tasks

The description should be as comprehensive as is possible. Model builders take user stories and build the model, set up processes and data feeds. If possible, include an Excel file to be used as a visual guide or template for what is required.

When writing user stories, it is easier to put the work in early and create a good user story. You'll find it more challenging to revisit a user story later in the project timeline. Write comprehensive user stories, however if the story starts getting long break it into separate pieces. In the next step, you'll estimate how long it will take to build out that user story. If the user story isn't well written, it will throw the estimating process off track.

User stories are usually written by a business SME in conjunction with someone from IT. Anaplan Business Partners and the project sponsor review the user story to ensure quality, accuracy, and business need.



Notice the User Story Task list in the graphic above. The task includes things that need to get done in addition to the actual model build. All tasks take time and require resources that need to be estimated along with the actual model build.

Typically, you can expect between 75 and 125 user stories in a project. Generating more than this range may indicate a need for a phased implementation.

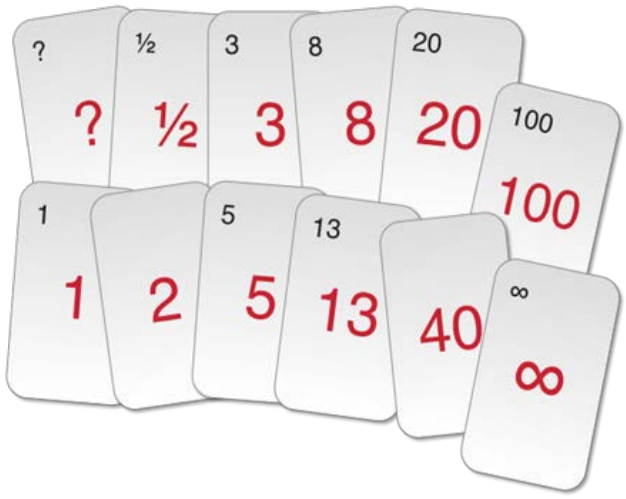
Determining the Level of Effort for User Stories

Sprint Estimation Tool (aka Planning Poker)

How do you plan how long each user story takes to build? Use the Sprint Planning Tool (aka Planning Poker). While it may seem a bit silly to play a card game to determine level of effort, using this tool results in:

- A stronger project team. This collaborative task serves to enhance team dynamics and define roles within the team.
- Diverse perspectives. Two heads are better than one; more are better.
- Consistency. Planning poker provides a unit of measure that you can use to easily recalculate how long all of the user stories will take if your estimate is off.

Cards are an important part of this tool. Members of the team make estimates by playing numbered cards face-down on the table, instead of speaking them aloud. The cards are revealed and the estimates are then discussed. By hiding the numbers and sharing them all at once, the team avoids bias – where the first number spoken aloud sets a precedent.



What do you need?

- Each person needs a set of poker planning cards. Alternate: have participants download a mobile app. They will need a standard set of Agile Scrum cards – search in the app store for Agile Scrum Planning.
- A moderator to facilitate the voting and decision making process and keep time.
- Seating around a table, so everyone can see the cards.

Before the planning poker session

Number and organize the user stories. If possible, group those that are similar together – then during the session, you can confirm that the stories would get the same number. You'll also want to determine the baseline story.

Determine a baseline

The first step is to share the baseline – or in other words, the number of points for a user story of medium complexity and medium build time. In the example below, the baseline is 8 points. The baseline should be between five and 13 story points. This table can be found on the Sprint Planning tab of the Agile Implementation – The Anaplan Way app.

User Story Estimates		Build Complexity		
		Low	Medium	High
Time Intensive	Low	1/2	1	3
	Medium	2	8	13
	High	5	20	40

It is important to understand how this baseline works. The base line IS NOT a straight estimate of time, it is an estimate of complexity and effort that equates to time. The difference is subtle but important.

In the example above a story with medium complexity and medium time equals eight points, while one with medium complexity but low time is two points, and medium/high is 20 points. If the group decides each point is worth two hours of development time then the medium/medium story should take 16 hours to develop, the medium/low should take four hours, and medium/high equals 40 hours.

This comes into play if adjustments are needed. If, for example after the first sprint, the group discovers their estimate was wrong it can be corrected quickly. Say the stories that were Med/Med only took eight hours (half the time estimated). With this system, you can just go through and refigure the math (done automatically in the Anaplan Way app). Now the story ranked two points would change from four hours to two and the 40 hour story is reduced to 20.

This allows you to quickly recalculate not only the story build times but also sprints. In this example, we can now put more stories into each sprint, decrease the time estimate for the release date or add more stories to this release from the master bucket all because the stories are not taking as long as the original estimate.

Determine meeting time

Next, determine how many minutes to spend estimating each user story during the meeting. Take the number of minutes the team will be meeting and divide by the number of stories to determine how much time you have for each story. For example, the team will be meeting for the rest of the day, so you have about 6½ hours of working time (taking breaks into account) and you have 75 user stories. That's 390 minutes / 75 = ~ 5 minutes per story.

Voting process

	Step	Time
1.	The user story owner gets up and reads the story to the group in such a way as all can understand it and can vote on how long they think the user story will take to create, end-to-end. The user story owner can answer questions or provide clarity.	For our example of 5 minutes per story, each story should take no more than 2 minutes. Five minutes – (1 minute for questions + 2 minutes for voting) = 2 minutes.
2.	The voting members think about the story and have one minute to ask clarifying questions.	One minute
3.	Everyone has 30 seconds to pick the card that represents the number of points for the user story.	Steps 3 – 6 take up to two minutes.
4.	When the 30 seconds is up, the moderator calls for the vote and everyone shows his or her card.	
5.	If there is consensus, the points are recorded and the team moves on to the next user story (step1). If there was not consensus, continue with step 6.	
6.	If there was not consensus, but there are consecutive cards, the higher card is chosen for the user story points. If there was not consensus and the cards were not consecutive, then the moderator calls for one minute of discussion and a re-vote. If there is still not consensus, the moderator adds the user story to the parking lot. The user stories in the parking lot will need to be re-visited at a later time.	

At the end of voting, you'll have story points for each user story. In the Agile Implementation -The Anaplan Way app, assign a multiplier for the project. Begin with two or three as the multiplier: see what works best by comparing the number of hours for the first sprint using the multiplier and the total number of build hours available for the sprint. Adjust the multiplier as the project progresses and model builders get past the learning curve and up to speed.

If you are able to get through user stories and have some time left in the day, go back to see if you can get through the stories in the parking lot. If you are out of time and parking lot items remain, schedule another meeting to work through those user stories.

Model Design

Time spent at the front end of the model design process is time saved during the model-building process. Great model design helps the customer avoid a trip back to the drawing board after you've realized you completely overlooked an important piece of structure or processes. In model design, begin by:

- Understanding the business requirements for the model (in the SOW)
- Identifying who will use the model (defined in the SOW)
- Documenting how users will interact with the model (defined by user stories)
- Establishing how data flows through the model (defined in Rough Cut or SOW)

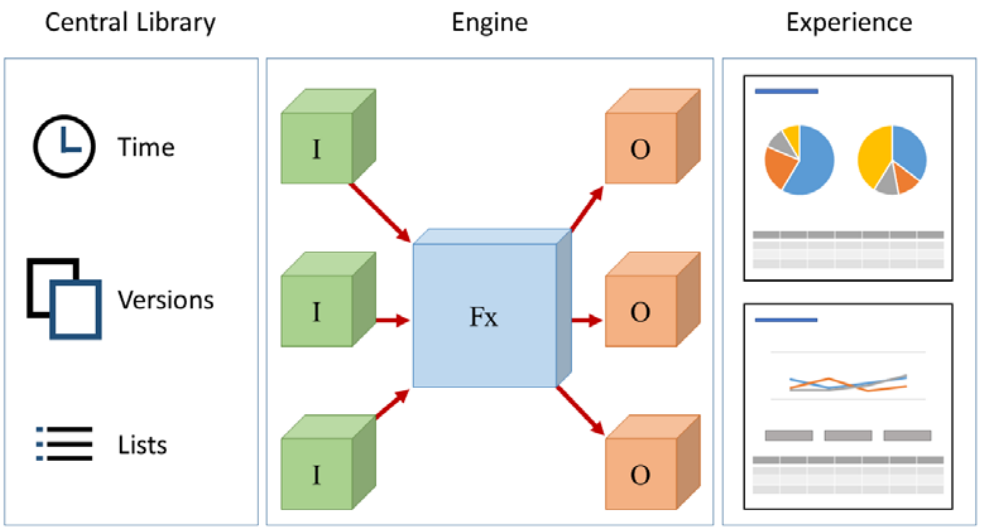
These considerations must be clearly defined to avoid any unwanted surprises after model building begins.

Remember: measure twice, cut once. When completing a model design, double-check information before you get started. Work through your design by white boarding or using flip chart paper to capture the model elements and process flow. Design the model before you begin to build. Anaplan uses a flexible, Agile process for development, so things will change and shift a bit.

Model Design Basics

The architecture within the Anaplan platform that drives model design contains three simple components:

- Shared dimensions and lists (Library)
- Modules (Engine)
- Dashboards (Experience)



Think of the shared dimensions and lists as being a library of the lists and versions needed for the model. It may include organization lists, including hierarchies, product lists, and employee lists. When a change is made to a list, that change is included wherever the list is used in the model.

Three types of modules do the real work:

- Use Input modules to import data. An input module might also be a module where the user will enter data using a dashboard.
- Use an Engine model to perform calculations.
- Use an Output module to collect data (possibly from several driver modules) needed for dashboards.

The user experience (or dashboard) holds the components that a user needs to do their job. They include:

- Charts
- Graphs
- Drop-downs
- Filter or change views
- Data input fields

Modules, parts of modules, and other objects generate dashboard outputs.

Model design can get very complex. There are many moving parts; both project teams and business partners who aren't well versed in model construction can become overwhelmed. Use this white boarding example as a way to help project teams understand Anaplan model design.

Model Design Approaches

You can think about model design two ways:

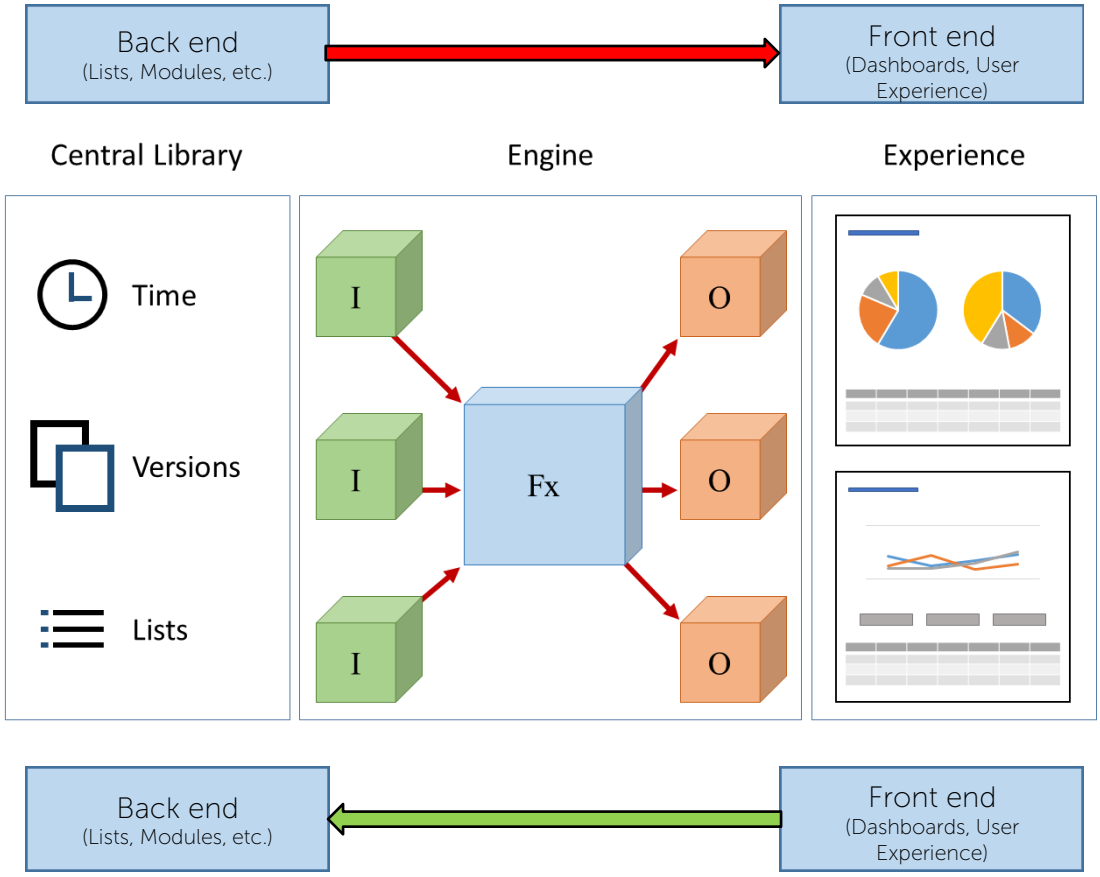
1. Begin with the back end - the lists and modules, and work your way to the front end, or the user interface.

   If you begin with the back end, beginning with lists, adding modules and finally, adding dashboards, you may include data that doesn't need to be included in the model. Your modules may lack clear purpose and your dashboards are an afterthought, rather than what is driving the model design.

   Imagine that you are going to build a piece of furniture, but the only thing you know is that the customer wants to sit on it. At the lumberyard, you gather all types of materials because you aren't sure of exactly what the customer wants. Are you

building a chair or a bench? So you throw materials that you would need to build both of these things in your cart. (Think of this as adding lists to your model.) Now you need to build something with all the parts you bought. (Building modules with the lists.) The customer talks about having a place to rest his arm. So you focus on that and build a beautiful chair with arms. (Now you've added dashboards to your model.) But when you deliver it to the customer, he says, "Nice chair, but what I really wanted was a bench for the garden, with an armrest that I can set a cup of coffee on."

2. Instead, Anaplan recommends beginning with the front end. This approach focuses on user stories and what the customer truly needs in the model.



Learn about the model design process on Community. Search for Model Design Process.

Model Design Review

Once the customer compiles a comprehensive set of user stories, you can take the model design laid out and add the details needed in order to complete the build. Create a model schema. Use a tool of your choice to create the model schema – Anaplan recommends Lucidchart.

Once you have created a model schema, have a trusted colleague review it. Model design check-in ensures project team success; it provides a second set of eyes to ensure the customer and business partner have not missed critical details. A new perspective also enhances your model design skills.

Once you have a model schema in hand, follow these steps for a model design check-in process:

1. Schedule a check-in with your business partner or a model builder with more experience than you. Provide a link to your model design file in the meeting invitation.

2. Prepare for the check-in by completing the Model Design Check-in Checklist document. During the check-in, you are expected to describe the customer perspective, show your model schema, and describe how the model solves the customer problem.

3. During the model design check-in meeting, listen to the changes the model reviewer proposes.

4. Document the meeting by placing a copy of the checklist in the project files.

Setting up Sprints

Plan the number of sprints and user stories that fit into each sprint. The number of sprints varies from company to company, as does the time each sprint lasts. Anaplan implementations have a baseline of four sprints of four weeks each. This does not mean all Anaplan platform projects take this much time. Nor does it mean that large projects will take this little time. Start with four sprints of four weeks each and adjust as you plan.

Decide the number of sprints and sprint durations once you have totaled up the user stories you have and how much time and resources each user story will take. Keep in mind, a sprint should not last longer than four weeks. Sprints that last more than four weeks are no longer a sprint, and the iterative nature of the Agile process could be lost.

Give each user story a priority:

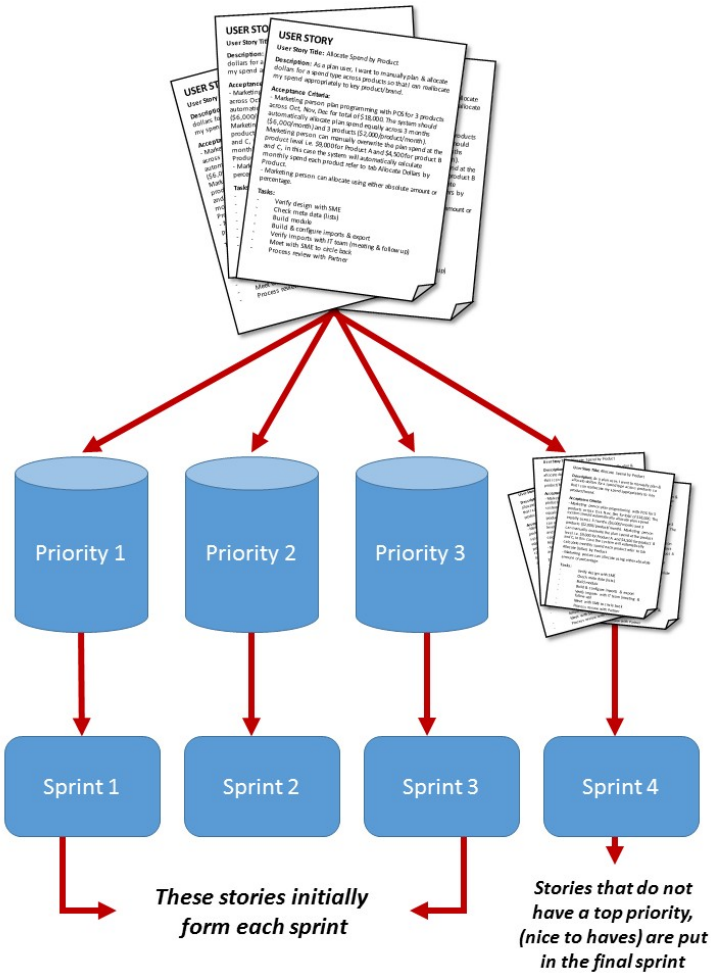
- P1 – Must have (highest priority)
- P2 – Nice to have
- P3 – Should have
- P4 – Won't be able to fit in this release

The project sponsor assigns each story its priority rating. Any user stories that do not fall under these priorities - effectively the P4 user stories - are initially put in the backlog. When you have the raw stories in priority, the time of truth arrives where you see how your stories, resources and time match up, or you may find out they don't match up.

The following table illustrates how to calculate your initial sprint plan. This implementation contains four sprints of two weeks each. Imagine you have one full time Anaplanner working on the project and the customer provides 25% of someone's time each week to the project. Based on a 40-hour work week, you have 100 hours for each sprint to complete the user stories. Next, total up the user story hours and see where you are with each sprint.

	Sprint 1	Sprint 2	Sprint 3	Sprint 4
Weeks	2	2	2	2
Anaplanners	1	1	1	1
Client People	0.25	0.25	0.25	0.25
Total Hrs. Available	40 + 40 + 20 = 100	40 + 40 + 20 = 100	40 + 40 + 20 = 100	40 + 40 + 20 = 100
User Story Hrs.	175	80	127	47
Surplus/Deficit	(75)	20	(27)	53

As shown in the chart, sprints one and three have a deficit. These sprints contain too many stories and not enough resources. However, other sprints contain a surplus. Move user stories around so that you have them more evenly distributed. Remember: only priority one items should be completed in the first two sprints.



Managing the Buckets

As you learn to effectively manage the buckets, you will have an easier time keeping projects on track. Model building accounts for just a fraction of the effort needed to get an Anaplan implementation to go live. Managing the project team, a project team's processes, and change management take the most of an implementation's time.

Use Agile methodology to provide the customer with choices throughout the implementation. Teach the customer the discipline of managing the buckets; instilling this discipline helps prioritize throughout the project and when customers want to make changes after implementation.

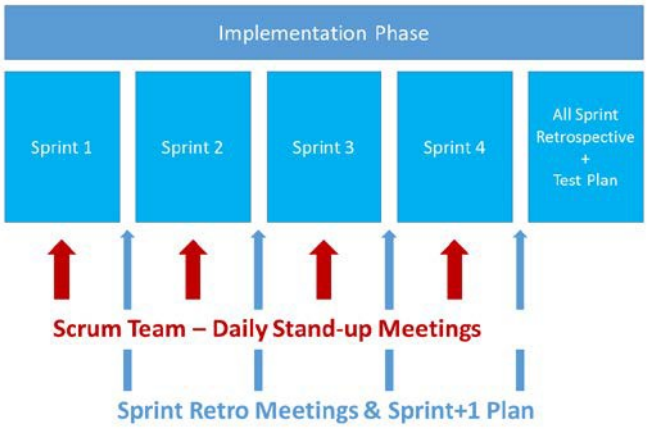
To manage the buckets, start with the business requirements (transferred into user stories). Check user stories for high quality and completeness before putting user stories into buckets. Adding poor user stories to a bucket delays your project and sprint(s).

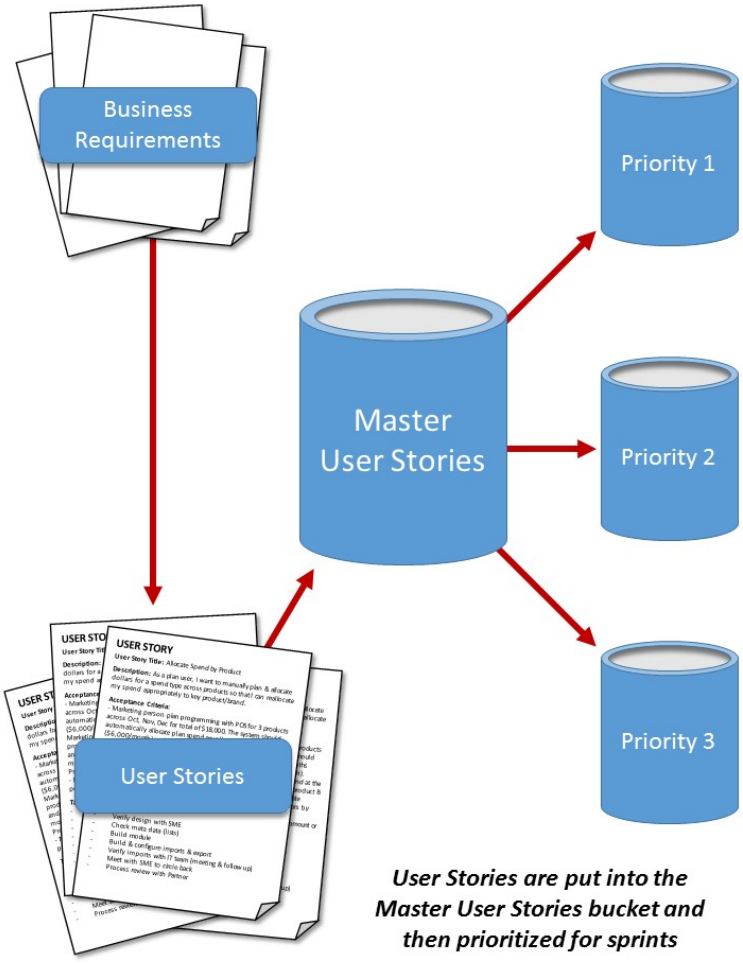
All user stories start in the master bucket. Even mid-sprint, any story that is added, starts in the master bucket. During sprint planning, user stories are prioritized P1, P2 & P3. Size user stories using the Planning Poker method. Place prioritized stories into the corresponding sprint buckets:

- P1 go into sprint one
- Some P2 go into sprint one; most P2 go into sprint two
- Remaining P2 and all P3 stories go into sprint three
- P4 user stories go into in sprint four or are placed in the backlog.

Look at the resources and time you have and balance the sprints (or buckets) so that you can complete the user stories in each sprint. Manage the buckets by considering all of your resources and timeline, and balancing the stories across sprints. Make sure you have the appropriate number of user stories in each bucket for the time and resources available.

When you complete the first sprint, some user stories will be completed and others will not. The stories that are not completed, as well as new stories, are added to the master bucket. Now balancing the remaining buckets becomes more challenging, because your buckets are full. Some user stories will need be shifted to a later sprint (bucket). You may need to decide that some user stories will not be built.

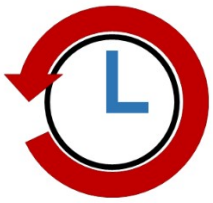




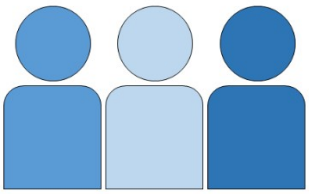
User Stories are put into the Master User Stories bucket and then prioritized for sprints

TIP: You will need to whiteboard Managing the Buckets several times during the implementation to really gain consensus on the balancing process, what goes into each sprint, and ultimately the release.

The Managing the Buckets conversation serves to control scope creep. As long as you ensure that the prioritized user stories are correct and create what everyone agrees is a good minimum viable product (MVP) /Release 1, (not necessarily perfect, but good enough), then you have effectively managed the buckets.



Timeline



Resources

User Stories

Consider the three key levers involved in keeping the buckets balanced:

1. Timeline. Does the timeline get extended to complete additional user stories?

2. Resources. Are there additional resources that can be added to the team to complete additional user stories? If this is the choice, consider ramp time to get new resources up to speed.

3. User stories. Which user stories can be moved to the next release to ensure that this release is completed on time?

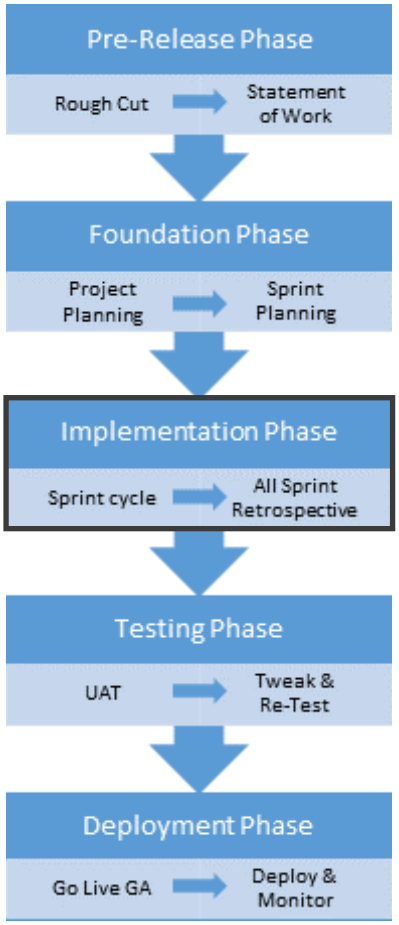
Which lever can be pulled to help keep the balance? Ask this question to your customer and make decisions based on their answer.

Don't Forget Your Cornerstones...

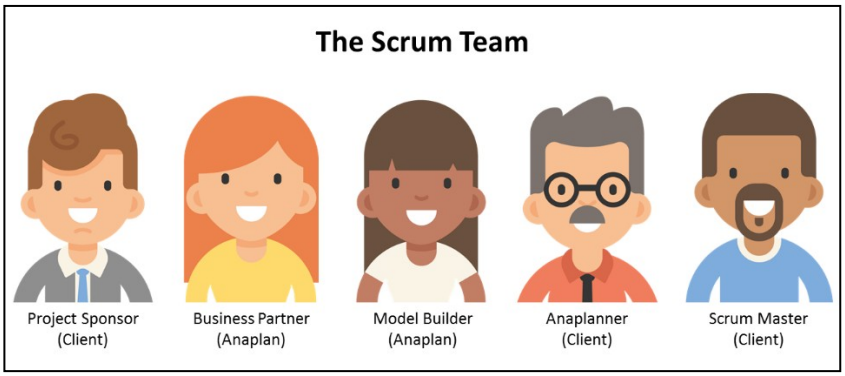
For the Foundation phase, your cornerstones include the items in the chart below. This chart does not include everything; there are probably other tasks your project requires.

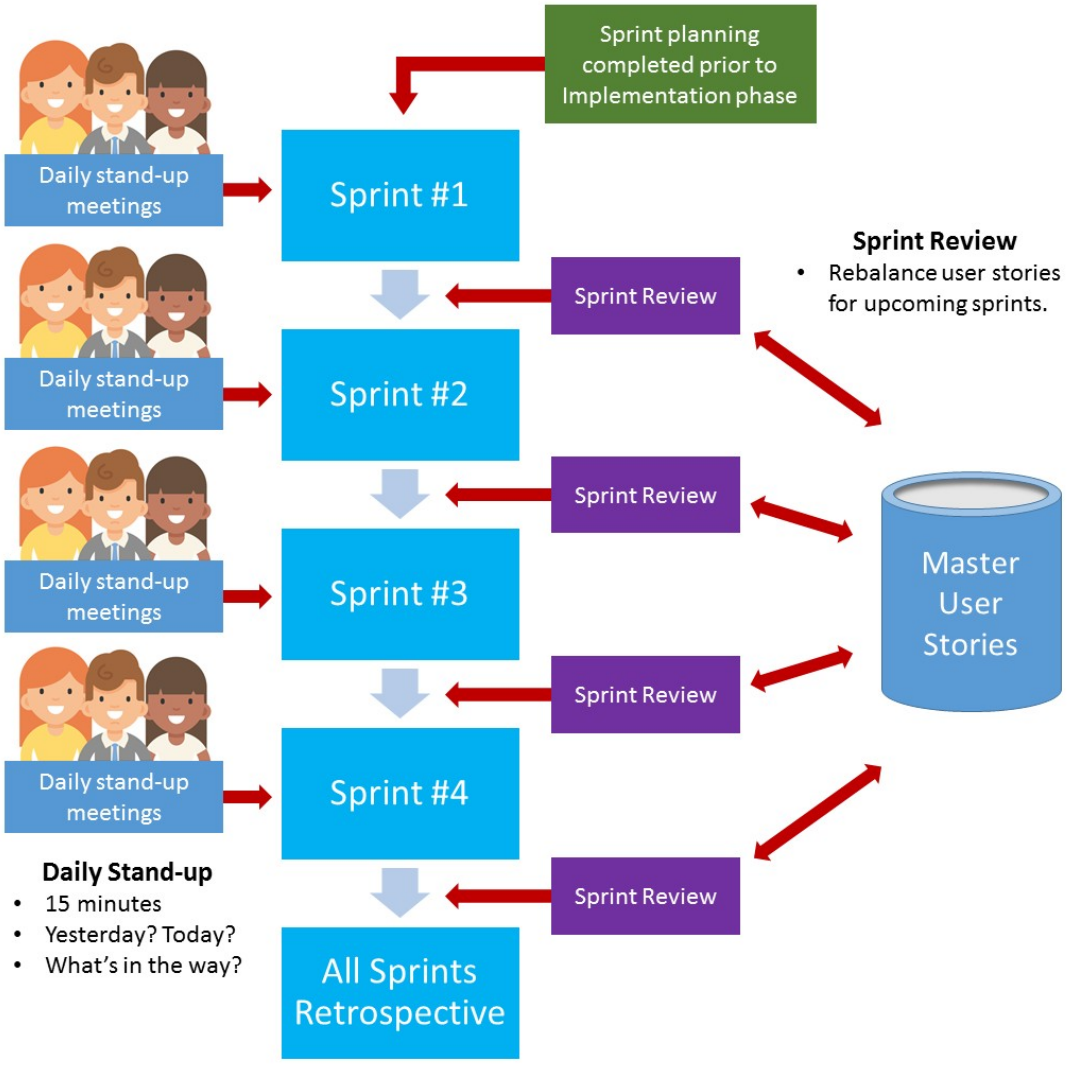
Model	Data	Process	Deployment
Verify design, topology, infrastructure	Analyze data	Verify upstream and downstream processes	Discuss creating the deployment plan with project team
Customer training on model building	Talk to data experts to determine how much cleanup will be needed	Agreement on where Anaplan will be used in the overall process	Discuss the communication plan for the project with the project team
Model design with check-in review	Add data clean-up plan to sprint plan	Define user roles	Select a good Scrum team
Estimate model building effort	Identify automation requirements and ETL tools	Capture existing processes	Agree on concurrency metrics
Write user stories	Identify all source systems		Set up project management tools
Translate process flows into steps and dashboards	Identify lists and hierarchies		Set up risk lob
Identify key reports and key performance indicators (KPIs)			

5. Implementation Phase



When working with the Anaplan platform, you can rapidly build prototypes that typically become the foundation of the production model. Unlike other platforms, very robust prototypes can be built by the model builders in early sprints so that stakeholders can see what the end model may look like. With Anaplan and the Agile methodology, we can quickly see if you are going in the right direction. We can create a prototype, sometimes 90% complete structurally, that can be "touched", discussed, used, tested and critiqued in the first sprint.





Unlike a traditional project management methodology, Agile runs iterative cycles, or sprints. Anaplan bases The Anaplan Way on an Agile methodology called Scrum.

For more information about Agile, take the Agile suite of courses on Community. Search for Agile Training. This chapter highlights Agile concepts that are important to the project, including tracking, daily Scrum meetings, sprint reviews, and the all sprint retrospective. It

also includes a section on rapid forensic analysis which can be used when there are performance problems with the model.

Tasks	Deliverable	Tools:
Model build	Anaplan Model configuration aligning to user stories Anaplan imports configured Anaplan exports configured Anaplan dashboards configured Anaplan User Roles and Security configured (joint ownership with client for all of these deliverables)	N/A
Project tracking	Weekly status reports to project sponsor	Agile Implementation – The Anaplan Way App
Sprint Review and Retrospectives	Sprints that have been adjusted when estimates are off, new user stories have been added	Agile Implementation – The Anaplan Way App Managing the buckets whiteboard
Deployment readiness	In-tool instructions & administration documentation Performance test scripts with SLA Demonstrations to end users on future end-state	Rapid Forensic Analysis Job Aid
Change Management	Updates as needed to change management plan. Work with client on this.	N/A
All sprint retrospective	Plans for improvement for the next release	N/A

The Customer and Agile

To people only familiar with a waterfall methodology, Agile can be confusing. Iterative software development may seem foreign. Emotions will come into play during the various sprints. At the end of the first sprint, the customer may seem skeptical. At the end of the second sprint, the customer will shift to excitement about progress – and may want all sorts of new user stories added. In later sprints, data issues rear their heads and the customer may feel it is too much to fix. Remember: project teams differ, and so will their reactions.

Stay calm and carry on. Manage the buckets and you'll do just fine.

The Importance of Commitment

People are more likely to complete the tasks they've been assigned if they make a verbal commitment to their team. Every Scrum team differs; consider these ideas for how to ensure the team is committed:

- Use the Anaplan tattoos. Ask people to show their commitment by wearing the (temporary) tattoo.
- Use the manifesto. Print the manifesto and have each member of the Scrum team sign it. If you have one place where you typically meet for your daily stand up meeting, consider creating a large version on flip chart paper and posting it in that space.

Tracking the Project

Use the Agile Implementation – The Anaplan Way App (AI-TAW) for project tracking. Download this app from the Anaplan App Hub. Learn the app capabilities in class 342: Agile Implementation – The Anaplan Way App located on Community.

These are the primary dashboards used in the App:

- User Story Details. The User stories dashboard acts as a tracking engine you can use in the app. User stories can be entered and then progress tracked from start to finish by percentage. Assigned users or project managers can update progress. This data impacts the Sprint Review, Project Phase Summary, and the Project Burndown Chart.
- Project Calendar. This dashboard displays tracking information on different timeline items such as the planning phases, sprints, IT environment benchmarks, data integration, and user testing. The dashboard tracks start and end dates, ownership of the tasks, and current status.
- User story. Track of all the user stories in the project here. Additionally, update user story status right in the dashboard.

- Daily Scrum Notes. Keep notes from the daily Scrum meeting on this dashboard. Keep and track outstanding action items on the bottom of the screen. Items can be entered, assigned, and status can be updated all in one spot.

- Sprint Review. This dashboard tracks many different items in a sprint including user story completions, status, and which user stories have not been started. It can be sorted by priority, project phases and people.

- Outstanding Actions. This dashboard calls out items listed in the Daily Scrum Notes dashboard. The dashboard includes a list of action items, status, progress, criticality, and who is responsible.

Daily Scrum Team Meetings

In daily 15-minute Scrum meetings, team members commit to accomplishing their next task. Commitment aligns a self-managing team. Most people are motivated to complete a task when they have made a pledge to the team that they will do so. Problem solving, status updates, and so on do not take place at Scrum meetings.

The meeting consists of each team member answering these three questions:

- What did I do yesterday?
- What will I do today?
- What obstacles are in my way?

That's it. Quick, to-the-point, yet critical.

If there are topics that require additional attention, these topics may be discussed after each team member checks in, or a separate meeting could be set up. Meeting time cuts into model building time.

Sprint Review Meetings and Follow Up

Hold informal sprint review meetings at the end of a sprint. The team demonstrates a working product increment to the project sponsor. External stakeholders, management, and project teams who are interested in seeing progress may also attend this meeting. Gather feedback used to shape the direction of the remaining sprints and the final product. Most users provide better feedback on a working prototype than a user story – seeing how something works gives them all sorts of aha! moments. Agile's iterative process allows us to build a product that couldn't possibly have been specified up front.

Sprint Retrospective Meeting

After the Sprint Review meeting, schedule and run a Sprint Retrospective. This meeting of the Scrum team and project sponsor can be held right after the Sprint Review, after stakeholders, management and project teams have left. First, adjust the sprint backlog is adjusted redistribute the sprints if estimates were off. Second, review how the team process is working.

Sprint Backlog. The project sponsor determines which user stories are considered complete. The Scrum Master returns incomplete user stories to the Product Backlog (Master bucket). Next, the Scrum team, Project Sponsor and stakeholders add any new user stories to the Product Backlog and Manage the Buckets!

Review how close your estimate of time to complete medium user stories was to the actual time they took to build. If the numbers are not close, update the sprint plan.

Note – during the review meeting that follows sprint two, the customer begins to see all of the potential of the Anaplan platform. They want everything and they want it now. Manage their expectations and always, the buckets!

Team process. After reviewing the sprint backlog and adjusting the sprint schedule, focus on the team process. How well is the team working together? In addition to checking in on team ground rules, ask some additional questions to generate discussion:

- What did we do well?
- Did we miss any opportunities to collaborate?
- What are our team's strengths / weaknesses?
- Do we have the right people in the right roles?

These discussions bring areas that need improvement to light. Then, the customer is empowered to make decisions about what to do differently to improve. Are the ground rules in need of an update? Have some behaviors improved? Recognize the good changes and work on eliminating the behaviors that get in the way.

Model Performance Discussion and Agreement

When the model includes most of what is needed to check its performance (after the second sprint), discuss service levels with the customer and agree on the service levels for the completed model:

1. Determine core processes /actions. Examples include:

    a. Adding a promotion to a marketing app
    b. Assigning a sales rep to a territory in a territory and quota app
    c. Assigning a compensation plan to a rep
    d. Changing a leaf location in a major hierarchy
    e. Performing a major allocation

    Expect an average of five to seven core processes in a model.

2. Establish the current performance baseline time in the model for each of these five to seven processes.

3. Discuss with the customer any difference between the current performance and the desired performance. Some processes take time. Work to understand the project team's reasons and requirements while doing all you can to manage expectations.

4. Determine if the performance can be improved to the level the customer desires, using the Rapid Forensic Analysis Job Aid as a guide. You may need to have another discussion with the customer to decide what service levels should be. Document the agreed upon performance levels and obtain a sign-off from the project team. This can be accomplished using email. Keep the agreement with the other project documentation.

Model Forensic Analysis

Usability relies on good performance. No user wants to wait more than a few seconds for data. Anaplan defines usability as both response time of the Anaplan platform and the visual appeal of the model.

You may experience poor model performance long before actual testing begins. Work to resolve performance issues sooner rather than later. The Rapid Forensic Analysis Job Aid provides various model attributes, the analysis tools used to determine that there is a performance issue, questions to ask and remedies to try.

You can access the job aid on Community. It is included in the Implementation Phase Materials in the Anaplan Way Resource Materials section.

All Sprint Retrospective Meeting

Hold the All Sprint Retrospective meeting when all sprints are complete. Include the entire project team. Reflect on the process and adapt the process for future releases. The meeting should not deteriorate into blaming or hostility. Important issues need to be addressed, not avoided.

To avoid those situations, lead the team through a series of steps:

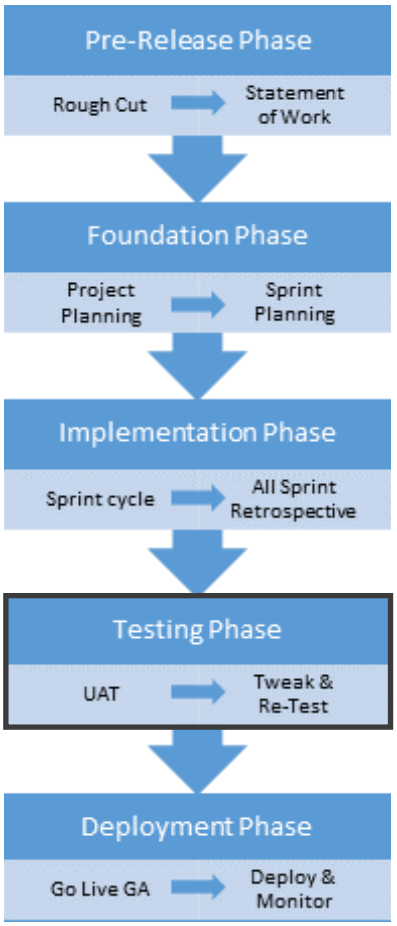
1. Set the stage. Focus on the process, not people. Do not allow blaming.
2. Gather data. Elicit fact-based information on the process. What worked well; what did not?
3. Generate insights. Given any problems uncovered during the data-gathering step, what can we do to solve those problems?
4. Decide what to do. Focus on answering these three questions:
   A. What do we need to start doing?
   B. What do we need to stop doing?
   C. What should we continue doing?
5. Close the retrospective.

Don't Forget Your Cornerstones...

For the Implementation phase, your cornerstones include the items in the chart below. This chart is not intended to be an all-inclusive list; there are probably others that your project requires.

Model	Data	Process	Deployment
Model build	Follow the plan for data clean up	Include business process experts in sprint review meetings	Communicate project updates
Begin writing UAT test scripts to ensure good testing of the model	Determine if additional resources are needed for data clean up	Implement connectors	Include end users in sprint review meetings
	Load data	Refine and validate business processes	Track project
	Implement automation of data imports		Run daily Scrum meetings
			Facilitate sprint retrospective meeting

6. Testing Phase



The Testing phase tests usability (including model performance), and manages bugs and change requests generated from testing. Focus testing efforts on ensuring the model does what the user expects it to do as well as predicting how the model will perform in a robust environment. Considerable tools, time, and resources go into testing. Testing must be scoped into the project during the scoping phase. Generally speaking, there are two types of testing:

- Model functionality and usability testing
- Model performance testing (Automated simulation of user activity at expected concurrency and measurement of response times)

Tasks:	Deliverables:	Tools:
Testing	▪ Test execution (joint ownership with client) ▪ Performance Validation with concurrent user loads ▪ Final performance testing report ▪ Updates to the model, based on UAT feedback	JMeter Splunk L3 Support
Triage and fix defects	▪ Defects and Change Requests tracking	Agile Implementation – The Anaplan Way App
Data Integration	▪ Updates to imports and exports based on UAT feedback	N/A
Process	▪ Updates to dashboards and modules based on UAT feedback	N/A
Deployment	▪ Updates to user access rights based on UAT feedback ▪ Weekly status report	Agile Implementation – The Anaplan Way App

Determining Testing Types

To structure testing that ends with user acceptance, some tests rank higher priority than others. Gather what you need to know about the model's production performance. Consider these three factors:

- Model size
- Model complexity
- Concurrency - the number of users accessing the platform

The table below serves as a guide for determining focus areas that produce the most reliable feedback.

	High	Medium	Low
Size	X	X	
Complexity	X	X	X
Concurrency	X		X
Tests Needed:	Load, Calculation, Concurrency, Usability	Load, Calculation, Usability	Calculation, Concurrency, Usability

Assess the state of a model with a variety of these tests. Prepare as if testing a large, complex, high concurrency model.

Where Testing Can Fail

Testing fails due to poor advanced planning. When starting a project, testing seems a distant milestone; it's easy for a project to avoid factoring it in until much later. However, a good testing process should be planned from the start of the project. This includes:

- Allowing enough time to run the testing and make tweaks
- Identifying individuals for live (UAT) testing
- Identifying the testable criteria upfront (as early as user story writing)
- Identifying potential hurdles to testing (technical limitations, global audience, and localization)

Performance Analysis Test

You can request a performance analysis of your model via Anaplan support. Even though we are talking about this in the Testing phase, you can request this analysis during the Implementation or Testing phases. Earlier is better. Average turnaround time is 7 to 10 days, but you can request it be done quicker – whether it is or not depends on the other projects that also need this analysis completed.

Request this test when you notice that model performance has slowed:

- When the model is slow to open or you have long rollback times
- Specific actions or processes in the model have become slow
- Increased duration of cell inputs
- Slow loading dashboards

The results of the analysis include:

- What is causing the slow performance and recommendations for how to correct it
- Model design issues that result in poor performance and recommendations for how to correct them
- Possible workarounds and best practice advice if suitable

Benefits include:

- A model that performs well with a single user baseline (which allows for better user concurrency testing)
- Information collected during the analysis will be contributed to the community so that all model builders can learn best practices
- Some performance issues may be shared directly with the product design team, leading to improvements and fixes.

UAT Testing

Because humans do unpredictable things, expect a labor-intensive UAT (humanoid) testing process. Humans follow a prepared script and execute steps over a set period of time to test for errors and performance problems. In addition to testing concurrency, capture other critical information from human testers:

- Use of the model with different bandwidths
- Use of different operating systems
- Location differences
- Browser compatibility

This data that can help identify and resolve usability issues.

You will not know if the tester follows the script as you might anticipate. If a tester becomes bored or distracted, steps can be missed and areas of the model with bugs can go undetected.

What Testers Are Testing For

Humanoid testing assesses a set of criteria in hopes that human testers accept model functionality and usability. Key questions used to direct the testers include:

- Is the model fit for its purpose?
  - Model must meet its intended goal and satisfy all stated objectives.
- Is it useable?
  - Model must be intuitive and cannot create confusion or frustration for users.
  - Model must perform as expected for single users and multiple users.
- Does it work as designed?
  - Defined processes must function correctly and work in a sensible way.
- Does the data flow?
  - Data must follow the model's logic to consistently produce the right output.
- Does it calculate correctly?
  - Formulas and functions must be built for accurate calculations throughout the model.
- Will we get end-user acceptance?
  - You will get acceptance if close attention is paid to the top five questions guiding the testing process and the questions are answered affirmatively. When this isn't the case, and you don't get end-user acceptance, the testing feedback must reflect the areas that need to be focused on in the UAT to meet the goal of buy-in from the end-users.

Preparation – Set yourself up for success

Early preparation helps humanoid testing go smoothly. To prepare:

- The customer and business partner should agree which actions are included in the testing script and the steps the testers will follow.
- Simplicity is important; include no more than 10 to 20 steps to complete in a script.
- Make the test scripts comprehensive, but not so exhaustive that the testers become bored with the process and lose interest. Write the scripts from the previous sprint as part of the current sprint process (example, in sprint two write the scripts that apply to the user stories from sprint one).
- Have testable data loaded in advance.
- Determine the role and selective access levels needed for testing and assign appropriate testers to each role.
- Create a presentation used to guide the users on the day of the test.
- Be sure all participants are online - including the testers, the project team, and the Anaplan consultants.
- Provide basic Anaplan end user training prior to testing to reduce the amount of "bugs" reported because testers don't understand how the system works.
- Make sure consultants are looking at Splunk reports and the server log files are evaluated throughout the testing process.

Writing UAT Scripts

Anaplan conducts humanoid UAT because people act differently than computers and are thus unpredictable. Good test scripts should contain:

1. The user story being tested.
2. The success criteria – a broad description of what the test should achieve and how that fits into the grand scheme.
3. Pre-requisites - Steps or procedures that the user must have completed before executing the test (i.e. any standard log in functions or anything they have to do to prepare the test environment).
4. Any known behaviors which may affect the user's ability to complete the script (i.e. any intermittent bugs or undefined behavior).
5. A step by step script, in tabulated form, with instructions on how to execute the test. with the following columns:
    - Step Number
    - Step Description
    - Requirements mapping (if applicable - put the actual Requirement that maps to this step, not all steps will map to a Requirement)
    - Comments - place for UAT tester to mark any pertinent comments (i.e. "I could not find that option/ could not click that button)
    - Pass / Fail - the result that the user got when trying to carry out that line of the script

User Survey

Once you've completed testing, launch a user survey. Base survey questions on conditions related to performance throughout the testing -- internet conductivity, variations of speed, and the performance over the testing period.

Do not send a survey if you already know you have poor performance or the testing results. There's no need to confirm something you already know. Only conduct humanoid testing when you know the results are going to be somewhat acceptable. Major system issues should be eliminated during the automated testing phase.

Triage

During testing, you collect rich information about the model, its performance, and its usability. During triage, determine what to do with the collected information.

Form a triage committee that includes an Anaplan business partner, a customer subject matter expert, and the project sponsor. Make decisions that define the next steps in the UAT. In general, the testing feedback falls into one of 2 categories: it's either considered a bug (defect) or a change request.

Categorize feedback as bugs or change requests; then assess the level of severity. Depending on the severity of the bug or the change request, it will either be included in the current release, or assigned to the Backlog and included in the next release of the model.

Review this guide for assessing how bugs and change requests are handled during the UAT process

Levels of Severity	Bugs	Changes
L1	Must have fixed by next UAT	Show-stopper functionality - must have
L2	Must fix and include in release	Desirable to have
L3	Desirable to have fixed	Likely in future release

If the team identifies a critical bug – it must be fixed and included in the current released – prioritize is as a L1 and factor into the UAT exit criteria. If a change request is identified as a show-stopper – or L1 – it too will be prioritized and follow the path of being included in the current release. Assign other bugs and change requests to include in the current release if possible, or if it's not possible, added to be included in the next release of the model.

Fixing Bugs

Determine time and resources needed to fix bugs so the customer can successfully complete UAT. If you have a number of L1 bugs to fix and the time and resources needed to fix bugs are extensive, all lower level bugs will be assigned to the next release. The UAT exit criteria should be referenced as a guide to follow for fixing bugs.

Adding Change Requests

The Statement of Work (SOW) every customer receives as part of the implementation process contains the requirements for the model and the procedures to follow for incorporating changes in the model. When the testing results include feedback that Anaplan reasonably determines is out of the scope of the SOW, Anaplan notifies the customer with an impact analysis of the request, a quote for the additional work and an action plan for handling the request. All change requests must be mutually approved in writing before the work involved in the scope change will be performed.

As with fixing bugs, prioritize changes requests by level of severity. Any change request considered a "show-stopper" gets top priority; other requests with less severe impact may become part of the next release.

Tweaking and Tuning

With the feedback from the testing prioritized, fine tune or tweak the model. Tuning contains three layers with each layer going a little deeper to validate a model free of defects and optimized for performance.

The three layers include:

1. Model Design
2. Calculation, Formulas, Blueprints
3. Core Code, Model Behavior

Layer 1 – (Model design) involves taking a closer look at some of these model specific details:

- Number of modules
- General dimensionality
- Numbered lists
- Subsets and composite lists
- Sparsity in modules

Layer 2 – (Calculation, Formulas, Blueprints) focuses on these calculation-related issues and items controlled in the Blueprint:

- Use of functions versus long, complex formulas
- Use of Booleans and other Blueprint settings
- Use of Summary Methods

Layer 3 – (Core Code) optimizes the model's code and assesses functioning at the Core level. At this deeper level is where you look at configuration issues and behavior that impacts the model's performance and overall functionality.

UAT Exit Criteria

When it's time to finalize the UAT exit criteria, it's often a decision of the team to set a percentage of the L1 bugs and a percentage of the L1 change requests that must be completed. For example, establishing as exit criteria that 80% of L1 bugs must be fixed and 20% of change requests must be included before the UAT process ends.

The Go/No Go Decision

Place the Go/No Go meeting on the calendar well in advance. This will help mitigate everyone's busy schedules as you get closer to the go live date. It also provides the team with a goal date to drive to completion.

Automated Testing

Work with the Customer Performance Testing team to schedule automated testing of load, performance, and concurrency. Using special testing software with defined instructions that execute processes repeatedly, you can see results that simulate a scenario stretching the model's use beyond its normal activity. Check out the charts in this section to determine if performance testing is needed and if it is, how long it may take.

Load & Performance

For this process, data is loaded into the model to simulate production model volumes. Basic functions are performed such as data input, break-back, (where appropriate), allocations, filtering, pivoting, sorting, list formatted item, and drop down manipulation. Imports and exports can also be included in automated testing. The Customer Performance Testing team can also simulate load on multiple models or multiple workspaces at the same time.

This testing can discover defects or highlight areas of slow performance that would be undetectable without extensive activity; it also determines the model's stability during normal activity. The testing also provides feedback on what level of user experience can be expected at a given level of concurrent activity.

If you determine some or many functions are slow and server memory and CPU are used to the maximum, you may have a case for model distribution. If, however, the model is slow, but user concurrency is minimal, then this could form a case for a single model instance, as the system is merely processing numbers and not being accessed by a user community.

The end user's experience, including performance, must hit the mark when you deploy models (see also section called 'Model Tuning'). In order to optimize performance, system administrators need to take into account the following factors when deploying to determine whether a single instance or distributed instance strategy is best:

- Data volume (memory usage)
- Model complexity (calculation logic and business rules)
- User concurrency

When should I performance test?

The Anaplan performance team cannot test during every project. Follow these guidelines on performance testing:

ID	Type/ Description	Range	Observations	Risk or Issues
01	Model Size	> 8 GB	Models over 8 GB are much more likely to experience moderate to heavy performance issues. Highly variable - dependent on type of use cases (see 04).	Highly variable - dependent on type of use cases (see 04).
02	Number of Cells	> 1 Billion Cells	Models over 1 billion are much more likely to experience moderate to heavy performance issues. Highly variable - dependent on type of use cases (see 04).	Highly variable - dependent on type of use cases (see 04).
03	User Base	> 400 accounts with access	With an assumed 15% user concurrency, a user base of more than 400 is likely to experience 60 or more active concurrent users in a peak. Good indicator of whether models will require performance validation on a mixed read/write model type (see 04).	Good indicator of whether model(s) will require performance validation on a mixed read/write model type (see 04).

D	Type/ Description	Range	Observations	Risk or Issues
04	Type of Model/Scenario	> 60% of concurrent users are actively writing to the model (as opposed to passively viewing the model)	Examples: All sales managers concurrently inputting daily sales in one geographical location in a predictable 1-hour time slot. Users inputting their figures to meet a deadline (accounting/financial period). The type of scenario requiring many cell changes in a peak business hour is a good case for performing load tests. Good indicator of whether models will require performance validation.	Good indicator of whether model(s) will require performance validation.
05	Complexity of Operations - Imports & Exports	≥1 import and/or >2 export operations are frequently used during peak business hours (including any processes)	Each import/export is potentially a very large blocking transaction. More than 1 frequent import(s) and/or more than 2 frequent exports (blocking transactions) in the midst of peak activity is likely to cause noticeable performance degradation for the majority of users. If a process is frequently used where there are more than 3 import actions in sequence - it is very likely for all users to be severely affected. Very high risk of encountering slow response times even for simple actions (such as opening/refreshing a dashboard).	Very high risk of encountering slow response times even for simple actions (such as opening/refreshing a dashboard).
06	Complexity of Operations - Adding Items	> 2 operations involving adding to a list frequently during	Adding items has often been a very slow transaction for the projects we've been engaged on.	After considerations of 01 and 02, there is a high risk of

ID	Type/ Description	Range	Observations	Risk or Issues
		peak business hours (including any processes)	After consideration of 01 and 02, there is a high risk of slow response times.	slow response times.
07	Complexity of Operations - Administrative	Major changes to the structure of the model or changes to user accounts/access, or even model restores	This does not typically happen on any models during peak business activity. These actions usually lead to very large blocking transactions. Very high. Reconsider use-cases/activity flow to one where impact to end-users is minimized.	Very high. Reconsider use-cases/activity flow to one where impact to end-users is minimized.

Testing Requirements

The requirements for automated testing are included here. Note that getting ready for automated testing takes time and should be included in the project schedule.

Model Sanitization

According to Anaplan security policies, all models placed into the testing environment must be sanitized. Sanitization involves the manipulation of data in a model to values that do not identify any company, persons, precise locations, company plans, or sensitive financial data.

Make a copy of the model, sanitize it and provide login access to the Customer Performance Testing Team. If there is insufficient workspace to do a model copy, L3 Support can assist by providing an isolated workspace to carry out the sanitization.

While it is best to sanitize all data, there may be situations where that is not possible due to time and effort constraints. The chart below ranks the priority for sanitization.

Priority	Data	Location	Examples	Sanitization Mandatory?	Responsible Team
1	Company Name(s)	Model Name / Workspace	Anaplan	Yes	Performance Team
2	Other Company Name(s)	General Lists	Accounts, Suppliers, Clients, Distributors	Yes	Business Partner/ Model Builder
3	Financial Data	Data Input Modules	Salaries, Revenue, Expenses, Sales Tax %	Yes	Business Partner/ Model Builder
4	Real Person Name(s)	General Lists	Employees, Partners	Yes	Business Partner/ Model Builder
5	Locations	General Lists	Sales Offices, Retailers	No	Business Partner / Model Builder
6	Products	General Lists	Biscuit Brands, Drink Brands	Yes – Brand specific names No – Generic	Business Partner / Model Builder
7	Services	General Lists	Dental, Advertising, Housing	No	Business Partner / Model Builder

Sanitization Techniques

The Customer Performance Testing team provides additional information and techniques for sanitization on their Confluence page. Sanitization techniques include:

- Modify numbered lists
- Temporary hardcoding of values
- Direct copy and paste
- Use import and export functionality

Test Scripts or Users

Test scripts can also be referred to as discrete users in performance testing. These are step-by-step instructions that can be followed by the simulated user.

The Customer Performance Testing team requires the finer details on test scripts/users, roles and selective access when the model's business processes become clear. A video that demonstrates the user role and its steps would give them the material to start evaluating whether these scripts are fit for performance testing. If a video recording cannot be created, an arranged meeting (and screen share) will be sufficient to talk through the steps. It is important that these details are captured accurately.

The ideal number of scripts is dependent on each unique model, but the team would typically expect to have multiple scripts where each script/user has a specific set of tasks related to their role. Multiple scripts enable greater control over the distribution of the work load, reflecting the load/usage patterns as though real users were using the system. Additionally, the Customer Performance Testing team should know where the user base will be geographically located.

Targets and Customer Requirements

The team needs the customer requirements of model performance:

- 90th or 95th Percentile target response times for each transaction
- Expected load volumes of the model by end users (pacing)
- Expected scenarios by end users
- Concurrency level of the user base (typically 15% to 20%)

These requirements are included in the questionnaire that the team has developed to capture the information they need to perform testing. It is available on the Customer Performance Testing Confluence page.

How long will performance testing take?

There is no simple answer to this question; every project has many variables that impact performance testing duration. The range is from a week and a half to four weeks.

User Concurrency

Model size and concurrency must be managed appropriately for end users to enjoy the best possible experience and get an average less than two seconds in response to most popular request. In many cases a project produces a base model that contains all the dimensionality and calculation logic. The model is subjected to a series of tests that determine end user experience and model performance.

If you have a model that requires an interaction with a large user base, user concurrency tests should be performed. As a general rule, user concurrency comes in at approximately 15% of your total user community. Therefore, if you have a total user base of 1000, around 150 people will be on the live system, performing tasks, at any given time.

In some cases, though, models follow a high concurrency pattern and this needs to be taken into account. For example, a weekly sales forecast may have 1000 users on the system, but very likely, each Sunday, (if forecasts are due Monday), the user concurrency will be quite high, maybe as high as 60%. Account for these factors. Customer processes and experience determine exact concurrency in high traffic models or periods.

Test concurrency in two pieces. First, schedule an automated test to simulate user actions across the system. Second, conduct a human intervention test that requires a group of people actually using the system at the same time to record and react to actual system behavior.

In some cases, automated testing does not reveal idiosyncrasies. Monitor the server while testing to track memory and CPU usage. In either test, tune the model afterward to optimize for all conditions.

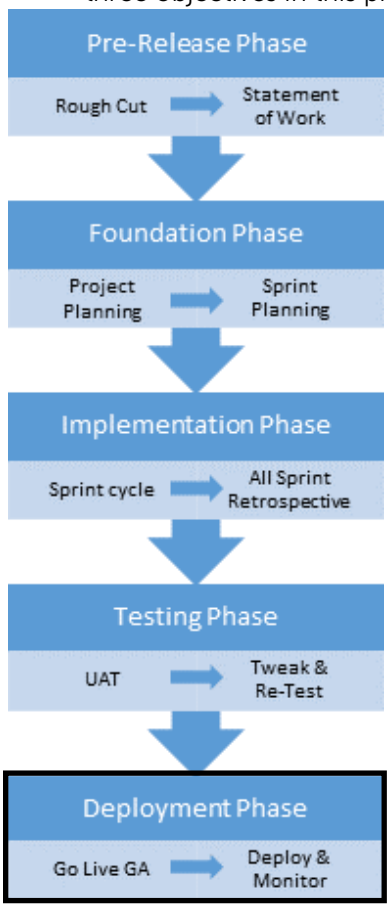
Don't Forget Your Cornerstones...

For the UAT Testing phase, your cornerstones include the items in the chart below. This chart is not intended to be an all-inclusive list; there are probably others that your project requires.

Model	Data	Process	Deployment
Use Rapid Forensic Analysis Job Aid to help identify performance problems	Ensure testing data is loaded	Ensure all types of users are included in UAT testing	Thoughtful selection of users involved in UAT
Request Performance Analysis test		Make sure to assign users to the appropriate role	Communicate UAT results
Analyze model impact based on test results, implement changes to improve performance		Identify tweaks to the business process	Create end user training plan
		Help end users align test scripts with the process by including scenarios	Review deployment timelines based on test results
		Analyze potential process adjustments base on test results	

7. Deployment Phase

You may recall deployment being one of the four cornerstones. You achieve three objectives in this phase:



- Get buy-in from users
- Make the Anaplan process stick in the organization
- Secure return on investment (ROI)

Deployment plans are developed with the customer well in advance of the actual time that the Anaplan platform is ready for general availability.

Tasks:	Deliverables:	Tools:
Communication plan	Work with the project sponsor to execute communication plan	See template included in *Communication* section in this chapter. Communication about the project should begin during the Implementation phase.
Training of users	Work with project sponsor to develop training plan	See tools and examples included in *Training* section in this chapter
Tasks:	**Deliverables:**	**Tools:**
Documentation	Overall model schema; Regional and business unit model schemas; Data and metadata schemas; processes documentation; Model maintenance; Model data flow; Base model blueprints; FAQ's	Coming soon!
Gather user feedback	N/A	Customer Health Model
Monitor Deployment and Performance	N/A	Splunk
Conversation with customer regarding Anaplan roadmap	N/A	See *Plan Ahead and Future Project* section.

Communication

Successful change initiatives start with clear, concise communication. Most organizations believe they provide enough information to employees during times of change; however, employees typically feel that there has not been enough communication. Your project sponsor cannot over communicate!

Work with your project sponsor to create a communication plan. Some organizations may have a team or department that drives all project-related communication. In this case, your customer may only need to provide the content of the messaging. In other organizations, the communication is left to the project team.

You can use this general template for planning communications. Italicized text includes a few examples of how you might use the template.

Communications plan for: *Anaplan Implementation*				
Overall Communication Goal: *Marketing department executives and staff can explain the benefits of replacing the current process of tracking programs with the new Anaplan app.*				
Audience	Communication Objective	Message	Channel	Timing
Marketing Executives	*Inform this group on overall progress and ability to meet deadline of Feb 1.*	*Progress report*	*Monthly Executive Meeting*	*November 15*
Marketing Managers	*Recruit at least two interested managers for the Sprint Review meeting*	*We are excited to be able to offer two chances to see the future of tracking programs at Acme, Inc.!*	*SharePoint article on Marketing Department page*	*November 1*
All marketing staff	*Present key features of the new Anaplan App*	*Joe Smith, (super user of current system) presents some of the key features of the new platform. Includes screen shots, a high level overview of how the process is changing and a side by side comparison of the old process steps and the new process steps.*	*Quarterly Marketing Meeting*	*November 18*

Training

Training facilitates a return on the customer's software investment. If users don't know how to use the Anaplan platform, will they be able to complete the new process? You have already planned for a good user experience by creating easy-to-use dashboards. Keep in mind that users are experiencing a behavior change -- so even the most user friendly model requires some training to help users make the change to Anaplan

Some organizations may have training departments that develop all new software training. If this is the case, plan to include some members of the training staff in sprint reviews and UAT (they generally are excellent testers!).

If the organization doesn't have anyone to assist with training, work with the customer to determine which audiences need training, what they need to know, how training will be delivered (classroom, virtual, etc.) and when the training will be delivered. Keep training simple: develop a job aid, which is a step-by-step guide for how to perform a task. The job aid forms the basis of the training program, with a trainer demonstrating the process. If a live demo is not possible, screen shots can be used. Participants follow along and ask clarifying questions.

Training for Model Builders

All Anaplan users who will be designing, building and maintaining Anaplan models require official Anaplanner accreditation. Only certified model builders who have successfully completed the Anaplan Certification course should be allowed to build on the Anaplan platform. Introduction to Model Building training is a 2-day instructor-led course. The course may also be completed using the On Demand version of the course - available in Anaplan's Learning Center. Introduction to Model Building covers and assesses key elements that will allow users to build on the Anaplan platform.

In addition, Anaplan offers intermediate-level model building courses and advanced topics.

Documentation

Documentation to support training of users and maintenance of the platform will be kept in a collaborative and secured folder (Box.com) and will be updated as needed with full versioning control. Anaplan and/or Partner Project Manager and the respective customer Project Managers take responsibility to create, maintain and distribute appropriate documentation, including, but not limited to:

- Overall model schema
- Regional and business unit model schemas
- Data and metadata schemas and processes documentation
- Model maintenance
- Model data flow
- Base model blueprints
- FAQs

Monitor Performance

After go live, monitor performance to ensure a strong customer experience, expectations and service levels are met, and that the model is being used, promoting adoption. Call quick attention to issues so that you can work with the customer to address them as quickly as possible.

Typically, monitor models with one or more of these characteristics:

- High volume
- High complexity
- High concurrency

Before you share performance information, make sure that you have already established SLAs for model performance. In addition, be selective about who receives information regarding performance, as some statistics will need translation.

Create an Anaplan performance app to monitor performance. Anaplan uses Splunk reports to populate the app with data. Determine the frequency of monitoring, the audience and also create clear translations of the data for the project team.

The app will include Minimum, Maximum, Average and Median values for:

- Model load time
- Model save time
- Toaster time
- Response time by object, measured in milliseconds
    - Dashboards
    - Modules
    - Large calculations
    - List loads
    - Actions
    - Processes
    - User

The Anaplan Way also focuses on maintaining customer's satisfaction with the investment they made in time, people, and cost. Customers should feel comfortable saying they built a solution for their future business success.

No enterprise system runs flawlessly on its own 100 percent of the time. As with any system, Anaplan and the customer must monitor what's happening as more data and calculations are added to models, more users are added to the platform; new and emerging processes cannot be overlooked.

Make monitoring a high priority for Anaplan as a critical process for every project team's success. Put reliable systems in place to track performance and fix defects at the earliest possible stage. These practices help you succeed with your customer and help customers expand their use of Anaplan to drive their business performance.

Gather User Feedback

We want to ensure that our project teams are using their models, they have trained model builders on staff, and they are happy with the way the model is working in their business. Client Directors and Business Partners perform check-ins at the 30, 60 and 90-day marks after go-live. Update the Customer Health model with the information gathered.

System Change Management Process

If the customer does not have a change management process in place for business processes, work with the customer to establish one. This can be as simple as gathering user feedback and tweaking the process accordingly, or it can be complex, including prioritization, scheduling, and delivery of updates to the process. Consider the solution you have delivered as a first step, which requires modifications after delivery. Regular modifications ensure that the model supports changes in the business process. Modifications to the business process can be more quickly added during subsequent releases with Anaplan.

Plan Ahead & Future Projects

After successfully launching the first release, revisit those items identified in the Process Workshop as other parts of the process the customer wants Anaplan to do. Do you remember when you did the white board of the business process and determined what this project would include? Revisit that whiteboard with the customer to determine what the next step is. Determine what the upstream or downstream models might be. What was the project team's first use case? What are the models that send data to the model?

Here is an example to help you think about this:

> You may have started, for example, with a territory and quota model. What's upstream from a territory and quota model? It's usually some strategic targets or it could be some sort of market segmentation data or some sort of share of wallet analysis that you've done. There could be varying model and data sources that could be produced on Anaplan. Today they could be in Excel or they could be in a legacy system. They could be used to actually feed the territory and quota model that you've just produced.

Similarly, there are models that are on the same level as territory and quota. And there are other models that territory and quota could feed. For example, territory and quota could feed an incentive and compensation management system. It could feed a territory allocation-type model. Those models could feed a P&L—cash flow and balance sheet for financial statements. The territory and quota could actually feed a sales forecasting model; this may be itself fed from Salesforce.com but has the strategic targets and then some forecasting within itself. That sales forecasting model then feeds a P&L.

You can see there's sort of a web of models upstream and downstream and models that are actually in line with one another that feed different models but can be used by Anaplan. What's critically important, of course, is that all of these things are fed from a central metadata hub that you can set up with Anaplan so that all of your metadata and all your data can come into that central location and then feed the different use cases, which themselves are connected to one another.

The customer may also look to create their own competency center, or Center of Excellence (CoE). With a CoE, customers build an Anaplan practice and models that will connect across the business. See *Chapter 8, Center of Excellence* for more information.

Don't Forget Your Cornerstones...

For the Deployment phase, your cornerstones include the items in the chart below. This chart is not intended to be an all-inclusive list; there are probably others that your project requires.

Model	Data	Process	Deployment
Set up users in the model	Ensure production data is ready	Business process training and/or documentation is rolled out	End user training
Monitor model performance	Load master data and data from data hub		Roll out model
Model documentation is completed and stored where it is available to the project team			Communication of project success
			Gather user feedback

8. Center of Excellence

Over time, businesses want to be self-sufficient. This means they can build and expand the platform largely on their own or with some help. In order to accomplish this, customers create a Center of Excellence (CoE) or perhaps, more simply, a Competency Center. A CoE can start small – some companies have just one person in their Anaplan Center of Excellence – and grow to the size needed in the company, as the number of use cases increases. Setting up the structure for a CoE early in the first project helps the customer to learn model building and The Anaplan Way (Agile) methodology from Anaplan or partners.

Establishing a CoE provides multiple benefits to the business:

- Maximize their Anaplan investment and adoption through the business
- Consistency in the following:
    - Model design quality
    - Execution of business processes/methodologies
    - User experience, especially when users span multiple models
- Provides a service in the business that has a holistic view of all the business processes and how Anaplan can connect all those processes
- Repeatable, faster "time to market" models for the business
- Central control with local flexibility

Some additional areas for the business to think about include becoming an active member of the Anaplan community and eventually, growing the CoE so that it can provide additional staff with the training needed to build models and use The Anaplan Way (Agile) methodology so that other departments and areas can use Anaplan.

CoE Ownership

In use cases which require a global deployment, (frequent in Sales Performance Management or SPM use cases) Anaplan recommends a centralized team of modelers. Centralization ensures everyone follows best practices, respects naming conventions, avoids duplication, and optimizes integration between models. These types of use cases often have a data hub at the center of the architecture.

Other use cases, for example Finance or Marketing, are owned by the business. In that case, CoE and model builders report in to the business.

In some large deployments involving multiple departments or business areas, the IT department owns the relationship with Anaplan. IT owns the model building activity, the CoE and serves the business. In this configuration, IT should have very open contact between the business and the Anaplan team so that they are aware of business changes and product updates.

For some deployments, IT is involved with data integration and authentication and is not involved in model building or the CoE. In these cases, the business owns the model building activity, and they run their requests for data through IT.

There may be some businesses who have little to no desire to house this type of function internally, instead choosing to form a relationship with Anaplan or an Anaplan partner who will provide a Center of Excellence. While this may meet the needs of the business, care must be taken to ensure consistency as staff rolls on and off, or staffing priorities shift.

Setting up the CoE

Planning for the CoE begins up front. Begin to think in terms of what a CoE means to the business and how to make sure some of the first steps in the project are completed in a way that sets a CoE in place.

Follow these steps in the Foundation phase to establish a CoE:

1. Include a data hub in the model design.
2. Document the business processes.
3. Document the model logic and business rules. Store these in a central location for easy access by all members of the CoE. Include:
    a. Model blueprint
    b. Module blueprints
    c. Model data flow (model map)
    d. Technical ecosystem topology

As you begin building, the customer needs to establish (with your help) the following:

4. Best practices
5. Usability guidelines
6. Governance process
7. Model help desk
8. Training practices
9. Return on Investment (ROI) study methodology

Also needed for a CoE:

10. Determine testing practices
    a. Select a triage team
    b. Create a triage process
    c. Establish requirement and defect definitions
11. Build a user community that will help with adoption and usability. Identify local champions.
12. Maintain or establish executive sponsorship
    a. Create a communication process that provides executives with updates
13. Create a common portal for all things related to Anaplan
14. Create a process for deciding which models to build
15. Establish a model review and upgrade process

While this may seem overwhelming, a successful CoE can be run with a single model builder. Just be sure to set up your first model with an eye toward expanding by using a data hub. Build a portal for shared documents/best practices. Establish a help desk – this is probably you if you are the only model builder! Reach out to the IT department and get their support and determine if it is possible for them to provide the first line of user support. Determine the key processes you need to establish for the CoE and get those in place. Now you have a structure in place that can handle additional model builders. Staff can be added as the number of Anaplan models grows.

9. Security and Privacy

Security is of the utmost importance to Anaplan. All client data supplied to Anaplan is considered confidential data and is handled accordingly. All Anaplan employees are required to sign a security agreement at the commencement of their employment.

Details of an Anaplan employee agreement are as follows:

In the course of normal business activities many Anaplan staff, both employees and contractors, will come into contact with confidential information belonging to project teams, prospective project teams and partners. Examples of this kind of information include:

- Sample data provided in the course of discussing and agreeing on requirements
- Data used when building Proof-of-Concept models
- Data files for importing
- Access to Anaplan models by consultants for assisting with model building
- Access to Anaplan models by development for troubleshooting

This document sets out the policy and procedures regarding privacy and security of such information.

Information Security Policy

All information belonging to project teams, prospects and partners is to be treated as confidential, unless it is known to be available in the public domain (and not as the result of a data breach), or written authorization has been given by the owner of the information to use it otherwise. Information should be shared only on a need-to-know basis.

> Note: The owner applies to the Client business process owner and is someone who is authorized to give permission. The written permission documentation has to be kept in a central location, such as attached to a Zendesk or JIRA ticket.

All Anaplan staff have agreed to non-disclosure agreements in the course of their engagement, and information received from project teams, prospects and partners is governed by those agreements.

Confidential information is not to be passed on to other staff or to third parties without the express written authorization of the data owner.

Storage and Transmission

Box (www.box.com) may be used as required for storing and transferring confidential data such as text files or Excel workbooks. Folder access must be configured so that only the relevant Anaplan staff engaged in the specific use-case have access, along with appropriate individuals employed or engaged by the project team. Data must be deleted once it has served its purpose.

Anaplan models containing customer data should only be kept on the PROD servers. With written authorization from a customer they may be copied temporarily to UAT for testing purposes, but must be deleted once testing is completed. They must never be exported as zip files outside the secure servers. The infrastructure team can assist when needed to copy models between servers without exporting and re-importing.

Confidential documents should not in general be sent by e-mail. If this is unavoidable, the documents must be encrypted and the password sent separately by other means such as text message. Do not send by e-mail to the same recipient.

If you receive a confidential document by e-mail, it should be copied to a secure location and the e-mail deleted. If the sender was an Anaplan staff member, please notify them so the document can be deleted immediately from their 'Sent Items' folder.

Where possible, confidential information should not be stored on personal computers, or on memory sticks, disks or other removable media. If this is unavoidable, the information must be stored on disks encrypted with a strong password.

Under no circumstances should any confidential information be stored in Dropbox. Dropbox files are replicated across many personal computers and this does not provide adequate access controls to ensure the privacy of the data.

Confidential information may not be stored in Google docs. The terms of use of these tools would grant Google access to the data.

Confidential information may not be put into Jira/Confluence, Zendesk or any other issue tracking software. Where confidential information is required (such as data needed to reproduce a bug) it should be stored in a secure location such as Box and a reference to its location included in the ticket.

Access to Anaplan Models

Access for Anaplan staff on Client Anaplan models should only be granted as needed, and should be removed as soon as it is no longer required. Anaplan staff unable to remove their accounts from Client systems must remind the Client to do so

Security Set Up During Implementation

The Anaplan platform includes many features that allow for secure access for users. Users are prevented from accessing each other's data and information. User stories will be constructed in the planning and requirements phase that will detail the security settings, (called Selective Access), that will need to be configured to allow for the appropriate level of security for users.

Follow these guidelines when creating and assigning roles:

- Restrict workspace administrators as much as possible
- Review access to all current models for 'at risk individuals': both workspace administrators and those who have partial model access
- Separate sensitive modules in new models or different spaces as necessary

If integration with Client security protocols, such as SSO, is required, these will need to be highlighted early in the process and an appropriate Client technical resource supplied for the project.