



## VALUE-DRIVEN DELIVERY

## **WHAT IS VALUE-DRIVEN DELIVERY?**

The reason projects are undertaken is to generate business value, be it to produce a benefit or to improve a service. So if value is the reason for doing projects, value-driven delivery must be the focus of the project throughout the planning, execution, and control efforts.

### **RISKS AS ANTI-VALUE**

Negative project risks (threats) are regarded as anti-value, or factors that have the potential to erode, remove, or reduce value if they occur. To maximize value, we must minimize risks, since risks can reduce value. This is why the value-driven delivery domain emphasizes risk-reduction concepts and techniques.

## WHAT IS VALUE-DRIVEN DELIVERY?

### EARLY VALUE DELIVERY

Agile methods promote early value delivery. This means the team aims to deliver the highest value portions of the project as soon as possible.

Stakeholder satisfaction plays a huge role in project success. Engaged, committed sponsors and business representatives who support a project are vital to removing project obstacles and defining success. By delivering high-value elements early, the team demonstrates an understanding of the stakeholders' needs, shows a recognition of the most important aspects of the project, and proves they can deliver. Tangible results raise stakeholders' confidence, build rapport with them, and get them on board early, creating virtuous circles of support.

Value-driven delivery is about making decisions that prioritize the value-adding activities and risk-reducing actions for the project, and then executing based on these priorities.

## ASSESSING VALUE

Business value is usually assessed in financial terms. For business projects value is commonly estimated using methods/tools and techniques such as return on investment (ROI), internal rate of return (IRR), and net present value (NPV).

Using economic models to assess value removes individual bias and emotion from the process of selecting and justifying projects. Instead, with economic models, you can focus on comparing a common variable (financial return) across projects. The results are “in the numbers,” and are (in theory) more objective than other project selection models.

**RETURN ON INVESTMENT (ROI):** The ROI is the percentage profit from the project. For example, if you spend ₦4,000,000 on the project, and the benefit for the first year is ₦5,000,000, then ROI equals  $(\text{₦5,000,000} - \text{₦4,000,000}) / \text{₦4,000,000}$ , which equals 25%.

## ASSESSING VALUE

**NET PRESENT VALUE (NPV):** It takes time to complete a project, and even when a project is completed, its benefits are reaped over a period of time, not immediately. In other words, the project is costing you today but will benefit you tomorrow. So, to make an accurate calculation for the profit, the cost and benefits must be converted to the same point in time.

The NPV of a project is the present value of the future cash inflows (benefits) minus the present value of the current and future cash outflows (cost). For a project to be worthwhile economically, the NPV must be positive. For example, assume you invest ~~N~~3,000,000 today to build a house, which will be completed and sold after three years for ~~N~~5,000,000. Also assume that real estate that is worth ~~N~~4,000,000 today will be worth ~~N~~5,000,000 after three years. So the present value of the cash inflow on your house is ~~N~~4,000,000, and hence the NPV is the present value of the cash inflow minus the present value of the cash outflow, which equals ~~N~~4,000,000 - ~~N~~3,000,000, which equals ~~N~~1,000,000.

## ASSESSING VALUE

**INTERNAL RATE OF RETURN (IRR):** The definition of IRR is the discount rate “at which the project inflows (revenues) and project outflows (costs) are equal.” From an economic perspective, the higher the discount rate, the better the project.

For example, if Project A has an IRR of 25% and Project B has an IRR of 18%, choose Project A to maximize the rate of return. This is because the greater the value for IRR, the more beneficial the project is.

## PLANNING VALUE

The focus on value-driven delivery permeates the agile planning process. We prioritize the work to be done on a project according to business value, putting high-value deliverables at the top of the list, or backlog.

The tools and techniques used in the planning value practice include Chartering, Value stream mapping, Customer-valued prioritization, Relative prioritization/ranking, Product roadmap, and Risk-adjusted backlog.

Prioritization (Level 1) and Agile contracting methods are knowledge and skills applied in this practice.

# PLANNING VALUE

## CHARTERING

The technique of chartering in agile projects has the same general goal as the Develop Project Charter process defined in the PMBOK Guide, but the level of detail and the set of assumptions are different.

Agile charters generally have less detail than non-agile charters, are shorter documents, and focus more on how the project will be run than on exactly what will be built.

Any elements of the agile methodology that may be different from an organization's normal processes, such as how changes are approved and then prioritized into the backlog after approval, should be clearly outlined in the charter.

Chartering in an agile environment should result in a flexible document that allows the team to respond to changing needs and technology and ultimately deliver high-value components that the organization can begin using quickly.



# PLANNING VALUE

## VALUE STREAM MAPPING

Value stream mapping is a lean manufacturing technique that has been adopted by agile methods. This technique illustrates the flow of information (or materials) required to complete a process. It can help determine the elements of waste that could be removed to improve the efficiency of a process. Value mapping usually involves creating visual maps of the process (known as value stream maps), following these steps:

1. Identify the product or service that you are analyzing.
2. Create a value stream map of the current process, identifying steps, queues, delays, and information flows.
3. Review the map to find delays, waste, and constraints.

## PLANNING VALUE

### VALUE STREAM MAPPING CONTD

4. Create a new value stream map of the desired future state of the process, optimized to remove or reduce delays, waste, and constraints.
5. Develop a roadmap for creating the optimized state.
6. Plan to revisit the process in the future to continually tune and optimize it.

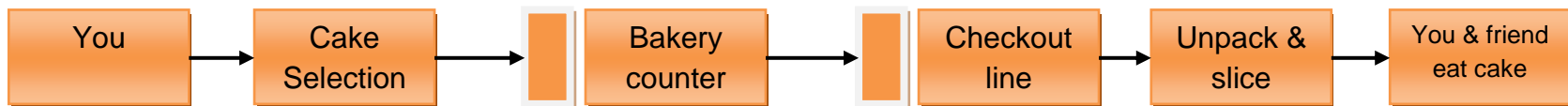


Fig. 1: Value Stream Map – Primary Flow

## PLANNING VALUE

### VALUE STREAM MAPPING CONTD

Figure 1 shows a typical value stream map. It illustrates buying a cake to celebrate with a friend. In this example, there will be waiting delays at the bakery counter and the checkout counter. There will also be motion inefficiencies, with lots of time spent between paying for the cake and being able to eat it, as we travel from the store back home.

To design a more efficient process, we could consider phoning a specialty cake catering service and having the cake express-delivered to our house. In this new, more efficient process, our waiting time to place the order, to pay for the cake and to walk home will be reduced by the fast delivery.

However, process efficiency is not the whole story – if the express delivery of the cake triples the cost, we might think twice about using it. On the other hand, the design does maximize the efficiency of the transaction.

## PLANNING VALUE

### CUSTOMER-VALUED PRIORITIZATION

Customer-valued prioritization is concerned with working on the items that yield the highest value to the customer as soon as possible. This technique aims to engage the customer in the prioritization process, in which the team identifies high-value features and moves them up the backlog of items to work on.

The use of customer-valued prioritization schemes is a common thread through the different agile methods. While the terminology often varies – Scrum, for instance, has a “product backlog”, FDD has a “feature list”, and DSDM has a “prioritized requirements list” – the idea is the same. The project works through a prioritized list of items that have discernible customer value.

## PLANNING VALUE

### CUSTOMER-VALUED PRIORITIZATION CONTD

#### PRIORITIZATION SCHEMES

The team should choose which prioritization scheme to use based on the needs of the project and what works best for the organization. Prioritization schemes include:

**SIMPLE SCHEMES:** One of the simplest schemes is to label items as “Priority 1”, “Priority 2”, “Priority 3”, etc. While this approach is straightforward, it can be problematic in that people have a tendency to designate everything a “Priority 1”. Without a shared, defensible reason for what defines “high” priority, we end up with too many items in this category and a lack of true priority.

## PLANNING VALUE

### CUSTOMER-VALUED PRIORITIZATION CONTD

**MoSCoW PRIORITIZATION SCHEME:** The MoSCoW prioritization scheme, popularized by DSDM, derives its name from the first letters of the following labels:

- “Must have”
- “Should have”
- “Could have”
- “Would like to have, but not this time”

The categories used in MoSCoW are easier to identify and defend than the “Priority 1” or “High Priority” labels of the simpler schemes. “Must-have” requirements or features are those that are fundamental to the system; without them, the system will not work or will have no value. “Should have” features are important – by definition. “Could have” features are useful net additions that add tangible value, and “Would have” requirements are the “nice-to-have” requests that are duly noted – but unlikely to make the cut.

## PLANNING VALUE

### CUSTOMER-VALUED PRIORITIZATION CONTD

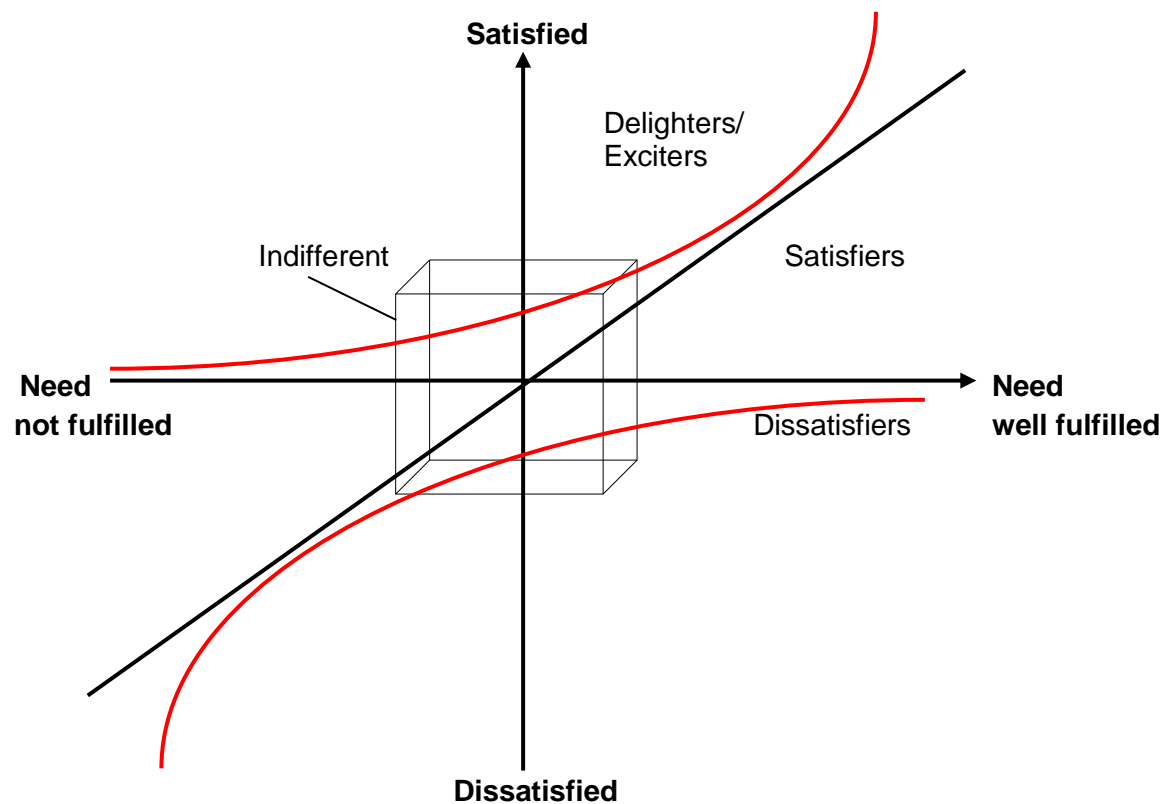
**MONOPOLY MONEY:** In this approach, sponsors are given Monopoly money equal to the amount of the project budget and are asked to distribute it amongst the system features. This approach is useful for identifying the general priority of system components. The Monopoly money technique is most effective when it's limited to prioritizing business features.

**100-POINT METHOD:** In this method, each stakeholder is given 100 points that he or she can use to vote for the most important requirements. The stakeholders can distribute the 100 points in any way: 30 points here, 15 points there, or even all 100 points on a single requirement, if that is the stakeholder's only priority.

**KANO ANALYSIS:** This technique can be used to classify customer preferences into four categories; Delighters/Exciters, Satisfiers, Dissatisfiers, and Indifferent. These categories can help the team better understand how needs relate to customer satisfaction.

## PLANNING VALUE

### CUSTOMER-VALUED PRIORITIZATION CONTD



**Fig. 2: Kano Analysis**



## PLANNING VALUE

### CUSTOMER-VALUED PRIORITIZATION CONTD

From the figure above, each category is described below:

- **Exciters:** These features deliver unexpected, innovative, or new high-value benefits to the customer. Exciters yield high levels of customer support.
- **Satisfiers:** These features bring value to the customer.
- **Dissatisfiers:** These are things that will cause a user to dislike the product if they are not there, but will not necessarily raise satisfaction if they are present. E.g. the ability to change a password within the system could be a Dissatisfier.
- **Indifferent:** These features have no impact on customers one way or another. Since customers are indifferent to them, we should try to eliminate, minimize, or defer them.

## PLANNING VALUE

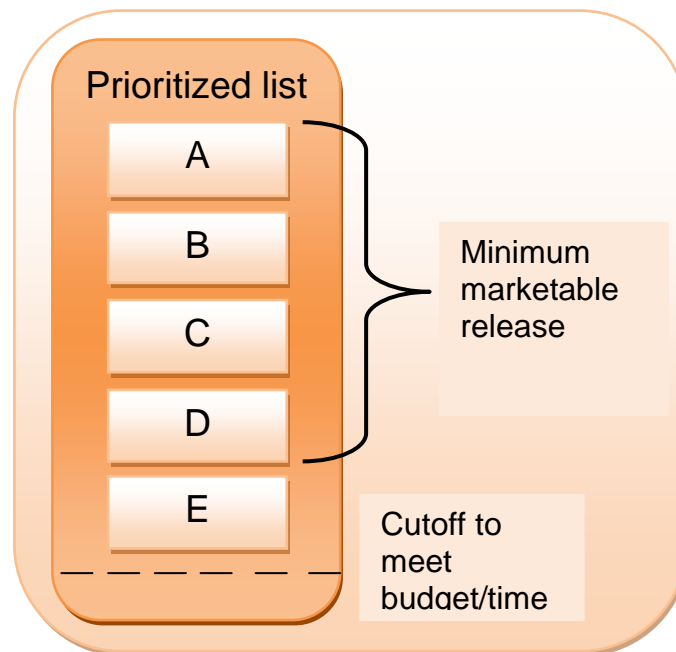
### CUSTOMER-VALUED PRIORITIZATION CONTD

**REQUIREMENTS PRIORITIZATION MODEL:** This is a more mathematically rigorous method of calculating priority than the other schemes. With this approach, the benefit, penalty, cost, and risk of every proposed feature is rated on a relative scale of 1 (lowest) to 9 (highest).

Customers rate both the benefit score for having the feature and the penalty score for not having it. Developers rate the cost of producing the feature and the risk associated with producing it. All the numbers for each feature are then entered into a weighted formula that is used to calculate their relative priority.

## PLANNING VALUE

**RELATIVE PRIORITIZATION / RANKING:** A simple list (whether it is created in a spreadsheet or in an agile requirements management tool) removes the categories that people tend to fixate on and allows the priorities to become the focus.



**Fig. 3: Simple Priority List**

## PLANNING VALUE

### RELATIVE PRIORITIZATION / RANKING CONTD

In figure 3 above, the features are listed in order of priority. The items at the top of the list – features A through D – are part of the defined minimum marketable release. If scope needs to be cut to meet the budget and schedule objectives, it's clear by looking at this simple list that adjustments should be made to E.

Relative prioritization also provides a framework for deciding if and when to incorporate changes.

A single prioritized work list is useful for simplifying the view of all the remaining work; rather than having separate “buckets” of work that represent change requests, defect fixes, and new features, a single prioritized list combining these items gives everyone a clear, complete view of everything that remains to be done on the project.

# PLANNING VALUE

## PRODUCT ROADMAP

A product roadmap is a visual overview of a product's releases and its main components. It is a communication tool that provides project stakeholders with a quick view of the primary release points and intended functionality. There is no one set way of depicting a product roadmap, but story maps, are a commonly used approach.

## STORY MAPS

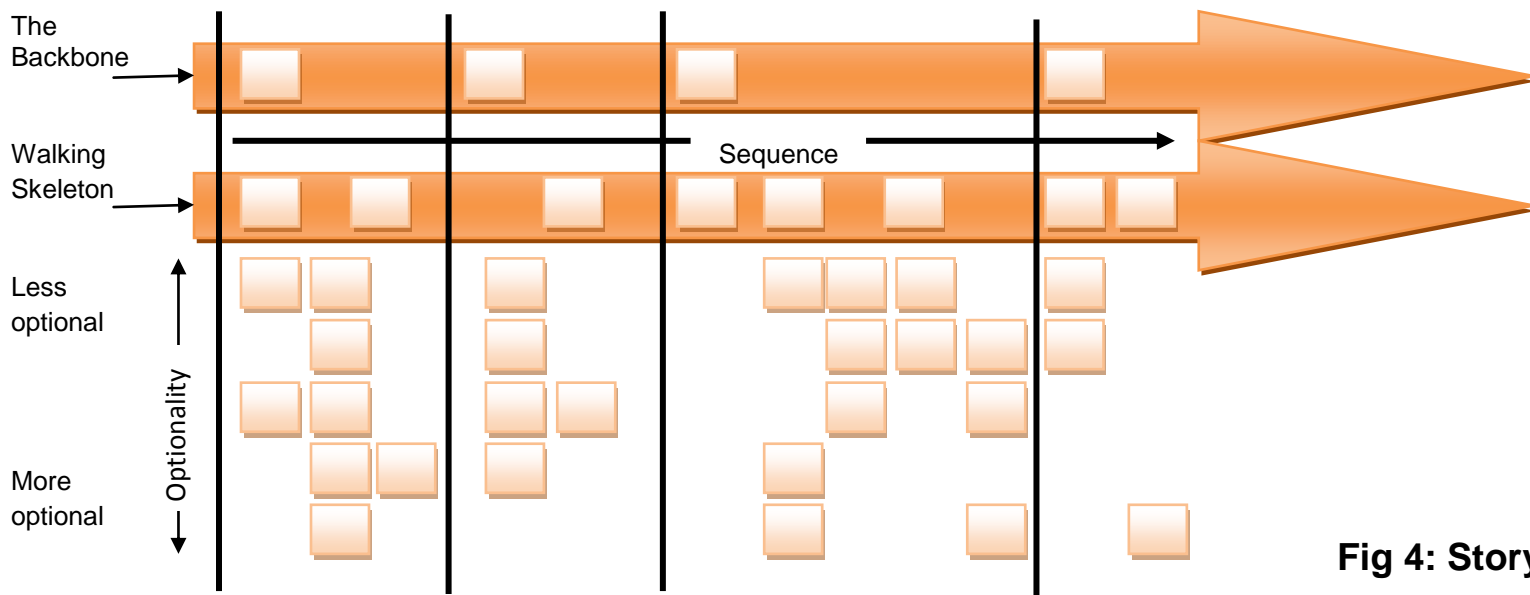
Story maps help select and group features for a release. These diagrams show the sequence of the features and indicate their importance to the project by classifying them as “backbone”, “walking skeleton”, or additional features.

The backbone is the essential functionality needed by a product or system. The walking skeleton describes the smallest system that could possibly work. Finally, any remaining features hang below the walking skeleton and are prioritized by their importance to the system.

## PLANNING VALUE

### PRODUCT ROADMAP CONTD

Using this approach, the final product roadmap consists of collections of story maps and shows what will go into each release. It depicts what is absolutely essential for the system (the backbone) and necessary for the solution (the walking skeleton), along with the other features that will comprise later releases.



**Fig 4: Story Map**

## PLANNING VALUE

### RISK-ADJUSTED BACKLOG

Risk management may seem like a traditional, process-driven project management effort that would not work well in an agile environment. However, agile methods are great risk reduction vehicles and are actually very well suited for rapidly identifying and reducing risks.

The iterative nature of agile projects tackle high-risk areas of a project sooner rather than later. This approach gets problems out in the open while there is still room in the schedule and budget to work on them. It also reduces the amount of effort invested in work that may end up being scrapped.

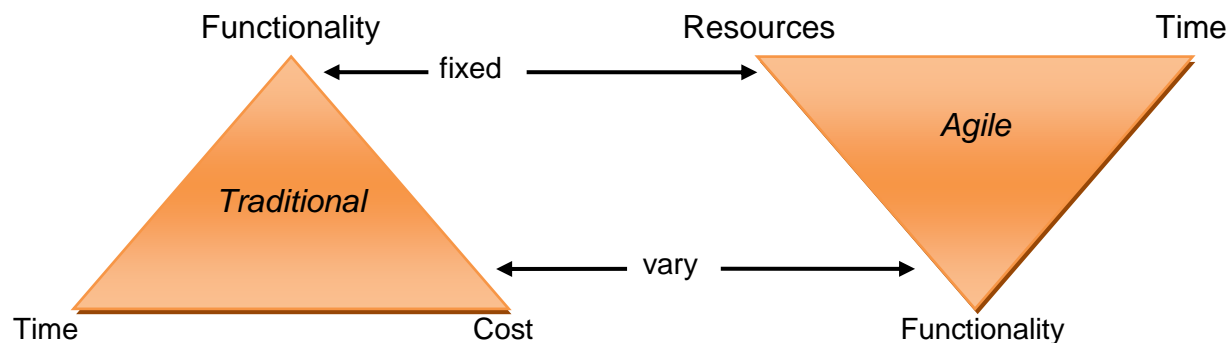
Assigning value to project risks help us focus on coming up with general, justifiable numbers that have consensus from the project stakeholders to use as a basis for prioritization.

Using this approach, we can rank the project risks to produce a prioritized risk list. For the risks that can be proactively tackled, however, the response actions can be prioritized alongside functional features. When combined, we will have a risk-adjusted backlog.

## PLANNING VALUE

### AGILE CONTRACTING

While agile methods provide great flexibility and allow us to manage changing requirements and priorities, this adaptability and scope flexibility can create problems when defining acceptance criteria for contracts and when outsourcing work.



**Fig 5: Inverted Triangle Model**



## PLANNING VALUE

### AGILE CONTRACTING CONTD

From figure 5, we see that agile projects attempt to fix resources and time (which are key components of cost) and vary functionality to achieve the highest-priority, best-quality product possible within those constraints. In contrast, when functionality is fixed, there is risk that the project will run out of money or time – or even worse, produce a poor-quality outcome.

The goal of agile methods and agile contracts is closer cooperation between the project team and the business or customer. This cooperation helps redirect the team's efforts toward delivering value-adding features. This goal is represented in the third Agile Manifesto value that ranks “Customer collaboration over contract negotiation.”

## PLANNING VALUE

### AGILE CONTRACTING CONTD

An agile approach also requires the business to be more involved in providing feedback on iterations, reprioritizing the backlog, and evaluating the value of change requests against the remaining work items.

There are different ways agile contracts can be structured:

**DSDM CONTRACT:** The DSDM contract was commissioned by the DSDM Consortium. This contract focuses on work being “fit for business purpose” and passing tests, rather than matching a specification.

**MONEY FOR NOTHING AND CHANGE FOR FREE:** This contract was presented by Jeff Sutherland. It is a standard fixed price contract that includes time and materials for additional work, but with a “change for free” option clause.

## PLANNING VALUE

### AGILE CONTRACTING CONTD

The customer can only use this “change for free” clause if they work with the team on every iteration. Failure to be engaged in such a way voids the clause, and the contract reverts back to time and materials.

The “money for nothing” concept is also only valid if the customer plays their part in the agile project. It allows the customer to terminate the project early when they feel there is no longer sufficient ROI in the backlog to warrant further iterations. E.g. The seller might allow termination of the contract at any time for 20% of the remaining contract value.

Using the “money for nothing” approach, the customer gets their top-priority business value, projects always finish early, and good relations are maintained with the seller.

## PLANNING VALUE

### AGILE CONTRACTING CONTD

**GRADUATED FIXED PRICE CONTRACT:** These kind of contracts are promoted by Thorup and Jensen as another type of agile contract. Here, both parties share some of the risk and reward associated with schedule variance. Thorup and Jensen suggest using different hourly rates based on early, on-time, or late delivery. For example:

Project Completion	Graduated Rate	Total Fee
Finish early	₦1,100/hour	₦920,000
Finish on time	₦1,000/hour	₦1,000,000
Finish late	₦900/hour	₦1,120,000

Under this arrangement, if the seller delivers late, they will get paid for more hours, but at a lower rate. If the seller delivers on time, they get paid for the hours worked at their standard rate. If they deliver early, they get paid for fewer hours – but at a higher rate.

## PLANNING VALUE

### AGILE CONTRACTING CONTD

**FIXED PRICE WORK PACKAGES:** Fixed price work packages mitigate the risks of underestimating or overestimating a chunk of work by reducing the scope and costs involved in the work being estimated. E.g. Breaking down statements of work (SOW) into individual work packages, each with its own fixed price. Then as the work progresses, the seller is allowed to re-estimate work packages in the SOW based on new information and new risks.

This approach allows the customer to reprioritize the remaining work based on evolving costs, and it gives the contractor the ability to update their costs as new details emerge, removing the need for the contractor to build excess contingency funds into the project cost.

## PLANNING VALUE

### AGILE CONTRACTING CONTD

**CUSTOMIZED CONTRACTS:** These different agile approaches to contracts can be pieced together to create a customized contract that benefits both the customer and the seller. With such contracts, the customer retains flexibility to reprioritize work and the seller is not penalized for sharing information about increased costs. They also remove the incentive for the seller to add large amounts of contingency costs to the project price. By combining elements of a graduated fixed price contract and fixed price work packages and incorporating the concepts of early termination (money for nothing) and reprioritization (change for free), we can create a contract that protects both parties and encourages positive behavior.

On agile projects, procurement has always been particularly challenging, since the scope is often not fully defined early in the project. In addition, the intangible nature of products like software can make it difficult to evaluate and get acceptance for the work. Any type of procurement – whether it's for agile or traditional contracts – works best when both parties want successful results that lead to future work.

## DELIVERING VALUE

The goal of delivering value is emphasized throughout the execution of agile projects. To accomplish this goal, the team should apply the lean concepts of maximizing value-delivering activities and minimizing “waste” or compliance (nonvalue-adding) activities.

For example, activities that may be necessary for an organization but are not directly focused on delivering value, such as time tracking and reporting, may be reduced or transferred while a project is being executed.

## DELIVERING VALUE

When aiming to maximize value, it is useful to look into the Poppendieck's seven software-related wastes.

Waste	Description	Example
Partially done work	Work started, but not complete; Partially done work can entropy	<ul style="list-style-type: none"> <li>➤ Code waiting for testing</li> <li>➤ Specs waiting for development</li> </ul>
Extra processes	Extra work that does not add value	<ul style="list-style-type: none"> <li>➤ Unused documentation</li> <li>➤ Unnecessary approvals</li> </ul>
Extra features	Features that are not required, or are thought of as "nice-to-haves"	<ul style="list-style-type: none"> <li>➤ Gold-plating</li> <li>➤ Technology features</li> </ul>
Task switching	Multitasking between several different projects when there are context-switching penalties	<ul style="list-style-type: none"> <li>➤ People on multiple projects</li> </ul>
Waiting	Delays waiting for reviews and approvals	<ul style="list-style-type: none"> <li>➤ Waiting for prototype reviews</li> <li>➤ Waiting for document approvals</li> </ul>
Motion	The effort required to communicate or move information or deliverables from one group to another, if teams are not co-located, the effort may need to be greater	<ul style="list-style-type: none"> <li>➤ Distributed teams</li> <li>➤ Handoffs</li> </ul>
Defects	Defective documents or software that need correction	<ul style="list-style-type: none"> <li>➤ Requirements defects</li> <li>➤ Software bugs</li> </ul>

Where we see activities on our projects that are wasteful, we can aim to eliminate them.



## DELIVERING VALUE

### TASK AND KANBAN BOARDS

The technical features that Gantt charts provide can actually present some disadvantages on agile projects. Scheduling software can illustrate very deep hierarchies of tasks, support task dependency integrity checks, and calculate interesting metrics, such as slack, subassembly costs, and resource utilization. Yet the technical sophistication of these tools is the principal reason they are not ideal for agile methods. The math, statistics, and quantitative reports that can be produced with these tools disguise the volatile nature of what is being analyzed – project tasks and estimates. In addition, sophisticated scheduling tools can alienate the team and discourage whole-team collaboration.

When we use tools that perform scheduling calculations and forecasting, two problems arise: data accuracy perception increases, and barriers for stakeholder interaction are created.

## DELIVERING VALUE

### TASK AND KANBAN BOARDS CONTD

These observations are reflected in Donald Reinertsen's book, "Managing the Design Factory", in which he warns of sophisticated models.

In contrast, agile planning and tracking tools employ a low-tech, high-touch approach. As implied by the name, these tools are simple, such as cards and charts, and are easy for all team stakeholders to manipulate by doing things like moving the cards, reordering lists, etc. By adopting these deliberately primitive techniques, we avoid a tool-related data accuracy perception and allow more people to update the plans as appropriate for the reality of the project.

## DELIVERING VALUE

### WIP LIMITS

Work in progress (WIP), also sometimes known as “work in process” or even “work in play”, is the term given to work that has been started but has not yet been completed. Having excessive levels of WIP is associated with a number of problems, including:

- WIP consumes investment capital and delivers no return on the investment until it is converted into an accepted product. It represents money spent with no return, which is something we want to limit.
- WIP hides bottlenecks in processes that slow overall workflow (or throughput) and masks efficiency issues.
- WIP represents risk in the form of potential rework, since there may still be changes to items until those items have been accepted. If there is a large inventory of WIP, there in turn may be a lot of scrap or expensive rework if a change is required.

## DELIVERING VALUE

### WIP LIMITS CONTD

Because of these problems, agile approaches generally aim to limit WIP. A common way to apply WIP limits on agile projects is to use Kanban boards that restrict the amount of work in the system and help ensure that WIP limits are not exceeded.

Agile teams use tools such as Kanban boards with WIP limits to help identify and remove bottlenecks so they can keep the process running efficiently with optimal levels of WIP. As a result, these tools help reduce the risks of tied-up capital, rework, and waste on the project.

Without limits on WIP, a project team may be tempted to undertake too many different pieces of work all at once.

## DELIVERING VALUE

### WIP LIMITS CONTD

The aim of WIP limits is to optimize throughput of work, not to optimize resource utilization. This is often counterintuitive to people at first. We tend to think team members should be busy working at all times, and anything else is laziness or inefficiency.

Limiting WIP helps identify bottlenecks and maximize throughput on a project, just like limiting the number of cars on a road helps traffic flow faster. On software projects, WIP limits equate to the number of features that are being worked on but are not yet accepted by the business.

## DELIVERING VALUE

### INCREMENTAL DELIVERY

Incremental delivery is another way to optimize the delivery of value on the project. With incremental delivery, the team regularly deploys working increments of the product over the course of the project. If we deliver the “basic” version of a product or service while working on the more complex elements, we have an opportunity to start realizing the benefits of the product and get an early return on investment.

Incremental delivery reduces the amount of rework by finding issues earlier and therefore, contributes to the delivery of value on the project.

## CONFIRMING VALUE

Agile methods are often used on projects that are intangible (e.g., designs, software, etc.). The intangible nature of these projects means it is all that much more important to validate that what we are building is, in fact, on the right track and seen as highly valuable by the business.

Because having a gap between requested and delivered features may lead to rework, it is important to discover these differences early. Therefore, confirming value – meaning verifying the team is building the right thing and that it works as desired rather than as first described, which could be different – is a key practice when using agile methods.

## CONFIRMING VALUE

### CUSTOMER-VALUED PRIORITIZATION

Working with the customer to prioritize the remaining items to be done helps ensure the work is progressing toward the target of the project, which may itself be moving. Together the team and the customer answer the questions of, “Have things changed?” and “Do we still want to move on to item X next?” These reprioritization sessions, which are typically done at the end of each iteration, serve as important checkpoints for agile projects. New and evolving priorities are then captured in the user story backlog, which is revisited again at the next session.

By asking the business representatives or customer what their top-priority features are, we learn about their motivations, risks, and acceptance criteria. Projects that do not engage in customer-valued prioritization are likely to miss out on identifying critical success factors.



## CONFIRMING VALUE

### PROTOTYPES, SIMULATIONS, DEMONSTRATIONS

Demonstrations of functionality are critical to confirming success on software projects. The term IKIWISI (I'll Know It When I See It) is often used in software development, because the true requirements may only emerge once the product is demonstrated and used.

In addition to helping clarifying requirements, demonstrations can uncover the need for new features.

When teams demonstrate functionality, two things occur. First, we learn about the differences between what was asked for and what was interpreted and built (the gulf of evaluation). Second, we learn about new or adjusted functionality (IKIWISI).

So requirements evolve with prototypes, simulations, and demonstrations. Giving people a chance to evaluate and use something helps uncover the true business requirements.

## TRACKING AND REPORTING VALUE

In an agile environment, it is important to monitor the rate at which features and value are being delivered to make sure we are on track to complete the project as agreed upon. Once we determine the project's status, we need to communicate that status to stakeholders.

There are several tools and techniques that we can use to track and report value, including earned value, cumulative flow diagrams, risk burn down graphs, and task and Kanban boards.

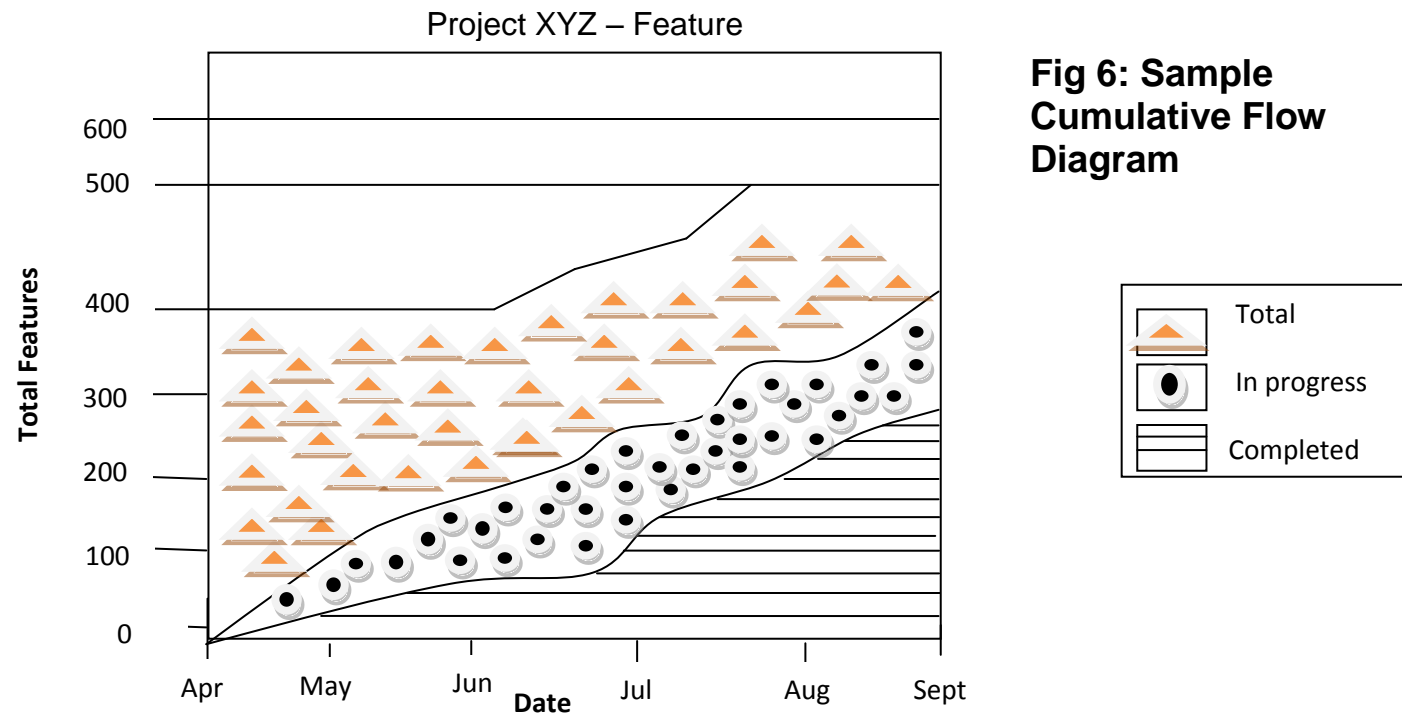
### AGILE EARNED VALUE

One of the key benefits of earned value is that it is a leading indicator. Perfect rear-view vision is not much use to us. EVM looks forward, trying to predict completion dates and final costs. After all, imperfect leading metrics are generally more valuable than perfect trailing metrics, since the leading metrics give us the opportunity to replan and change our approach.

# TRACKING AND REPORTING VALUE

## AGILE EARNED VALUE CONTD

Another benefit of earned value is that it is visual. Visual depictions help people discuss information and work collaboratively, since they can point, annotate, and extrapolate more easily with pictures than with words or numbers.



## TRACKING AND REPORTING VALUE

### CUMULATIVE FLOW DIAGRAMS (CFDs)

Cumulative flow diagrams are valuable tools for tracking and forecasting agile projects. CFDs can help us gain insight into project issues, cycle times, and likely completion dates.

Figure 6 above shows the features completed versus the features remaining for a fictional project that is still in progress. The orange triangle area represents all the planned features to be built. The dotted section plots the work in progress, and the striped section shows the total number of features completed on the project.

**LITTLE'S LAW:** This is a concept introduced by Donald Reinertsen that can be used to analyze queues from CFDs.

# TRACKING AND REPORTING VALUE

## RISK BURN DOWN GRAPHS

Risk burn down graphs are excellent tools for showing status and trends. They are essentially stacked area graphs of cumulative project risk severity. The severity scores for each risk are plotted one on top of another to show the project's cumulative severity profile. When risks and their history of severity are displayed in this format, it is much easier to interpret the overall risk status and trends of the project.

Escalating risks and new risks are also easy to spot on risk burn down graphs.

Risk burn down graphs quickly inform stakeholders if the risks are moving in the right direction (downward), or if they are escalating. These graphs provide an easy-to-interpret way to check the health of your project and can be quickly produced in tools like Microsoft Excel.

# TRACKING AND REPORTING VALUE

## TASK/KANBAN BOARDS

Tools like task boards help with tracking and reporting value, and their results are often shown on information radiators, which are large charts that are displayed in prominent places.