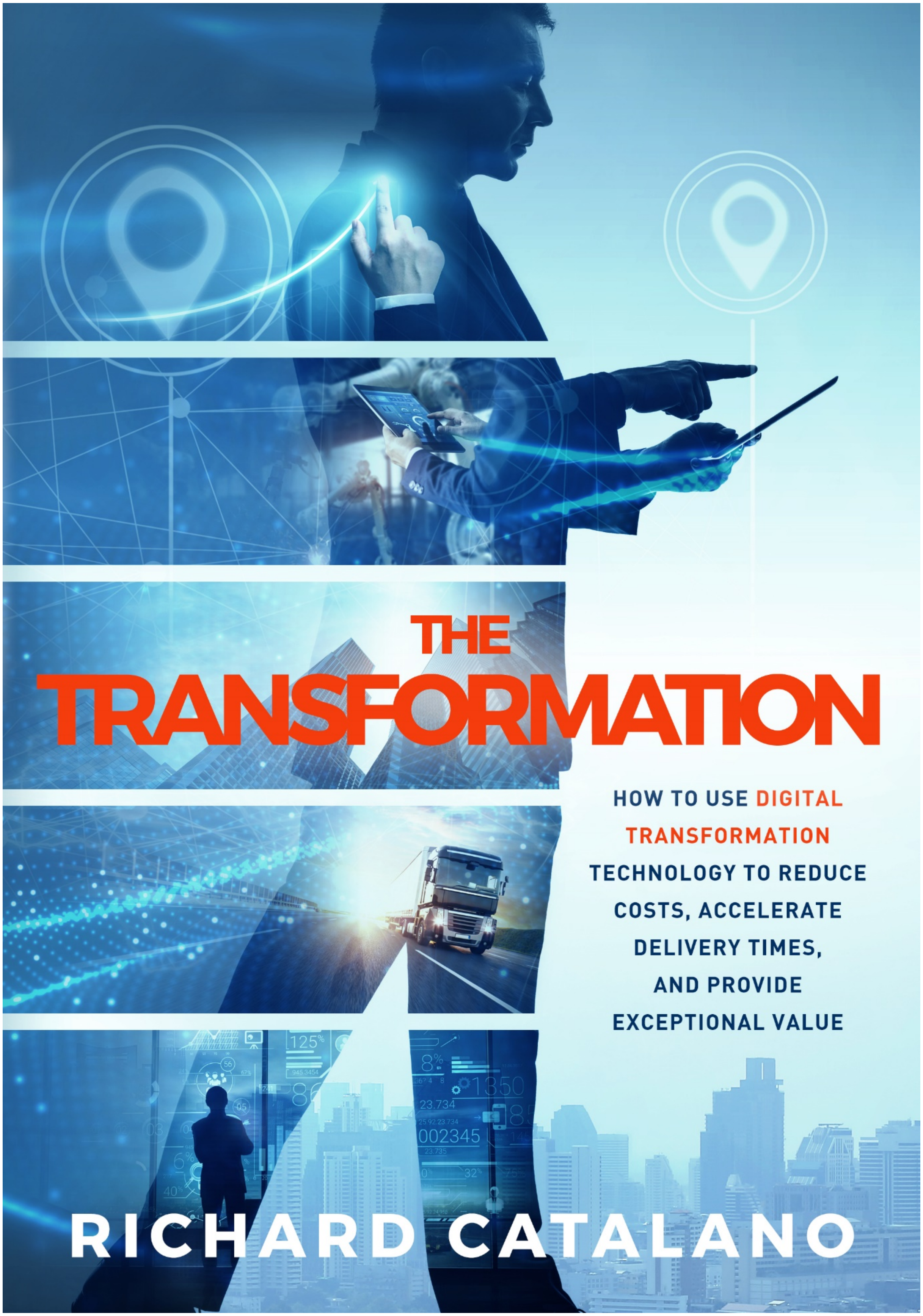

THE
TRANSFORMATION
HOW TO USE DIGITAL TRANSFORMATION TECHNOLOGY TO REDUCE COSTS, ACCELERATE DELIVERY TIMES, AND PROVIDE EXCEPTIONAL VALUE
RICHARD CATALANO

Chapter 4: Teams (Extended Version)

Author Note: In order to make the book a more concise read, there were certain areas where we needed to cut out some good stuff. This extended version of chapter 4 goes into much greater detail on the overall governance model and program organizational structure.

Mike was getting names and information on the team members he would have at his disposal. They were pulling some of the top people from various departments all over the company to spearhead the initiative, but right now, he was simply overloaded by the flood of names, bios, and specialties.

A few names were jumping to the top, recommended by superiors, or championed by one silo or another for their own purposes. Mike didn't know some of the people personally, others just by reputation, but he knew many of the names. Chet Osborn was one of them—a guy Mike had worked with before. Chet was intelligent, could be pretty aggressive, and had once been on a similar trajectory to Mike himself—before his demeanor caused a personnel problem with people below him. Chet's raw potential could make him a real asset, but his reputation also told Mike he might be more trouble than he was worth.

Another name he knew was Gina Paterno, a woman he had worked with on the Workseam Project. Mike liked Gina; she was smart, had good intentions, and was generally well-liked. But she could also be a little slow to make decisions and adapt to new situations. He wasn't sure how well she would do since the whole transformation initiative was by definition new.

"How you organize your teams can help you determine how you use your resources," Jonas explained. He and Mike were walking side-by-side through the

Extron building about a week after their time in the coffee shop. Mike's workload had started shifting as he reassigned most of his prior work commitments and started shifting the majority of his daily efforts to his new role and began the planning.

Mike and Jonas had been talking about the importance of getting off to a well-ordered start. "That's the first item of your integrated core, isn't it?" Mike verified. They hadn't really talked about it specifically since, but Jonas had already begun proving himself very valuable. His presence was stabilizing, and he was quickly becoming a friend as well.

"Exactly," Jonas confirmed. "Organization and resources is one of the fundamentals that you will use to successfully drive our digital transformation initiative all the way to completion."

"Do you have any tips on how to get all the data organized?"

"When setting up a digital transformation, I typically look for seven core organizational areas: process teams, technology teams, data teams, organization transformation teams, program leadership team, advisory council, and steering committee team. You don't get to pick the advisory council and steering committee team of business process owners, executive stakeholders, the program sponsor, and other executive-level members, so I always suggest starting with what you have the most say over."

Together, they entered Mike's office, which had become a de facto meeting place for him and Jonas. "So business process teams first? Why them first?"

“As a rule of thumb, your processes drive all the other aspects of scope and are usually a good place to start laying your foundation. As you define your program’s charter and scope, you’ll see how the options determine which processes are in scope for your program. So far, from what I have gathered, some key strategic objectives need to be accomplished with this program. From a customer perspective, we need to relook at our value proposition and identify new ways to add value for our existing customers and connect our brand to new customers who are looking for alternative energy suppliers. This looks like there are opportunities to re-envision the market-to-opportunity process to connect our brand to new and existing customers as well as the meter-to-cash process, along with our smart grid upgrade and Internet of Things integrations, to enhance the overall experience a customer has in dealing with Extron.

“There are great opportunities to engage with customers on a whole new level to not only enhance their personal portal experience on your website, but even going so far as to send individuals text messages for power outage updates and real-time messaging on off-peak and peak energy usage. We can even automatically run their dishwashers and washing machines at the correct times of day where energy supply is high and demand is low, thereby lowering the customer’s energy consumption costs.

“From a supply chain perspective, we need to do a better job of reducing costs and building efficiencies throughout the entire supply chain. That’s typically looking for opportunities in your procure-to-pay processes, including vendor management, product/materials management, transportation management, and warehouse/facilities management. There are some great opportunities in the world of artificial intelligence to allow Extron to optimize its inventory to save the company hundreds of millions of dollars in inventory costs. We will need to investigate a little further if there are other nuggets of opportunity in the supply chain that can help bring value to the organization and satisfy leadership's strategic objectives.

"If I recall, the final objective was the need to get better controls on our assets. We need to be able to know exactly where every piece of equipment is throughout our geography. We also need to be able to better predict when maintenance is required on each piece of equipment and actually fix problems before they happen. This will most likely require a complete overhaul of our enterprise asset management processes, as well as developing AI-driven predictive analytics capabilities that will allow us to perform big data computations on each and every smart part sensor you have on each piece of equipment, along with its corresponding functional location coordinates, across your entire geography."

"So, who do you suggest we put on these teams?"

"Well, there are various business process team roles: the process owner—an individual who is in senior leadership who owns the business process for your entire organization.

“Business process team leads—the person who is the on the ground day-to-day leader of your business process. They are responsible for defining and delivering the new business processes in their area for the organization.

“Business sub-process team leads—these are the people assigned by the business process lead to lead the design and development of the sub-processes within the core business process. The scope of a business process is pretty wide, and because of the level of granularity you need to run your organization, you have to break it down into more manageable components. There are a variety of ways to break down your sub-processes within your process area. In my experience, I usually see anywhere from three to six subprocess leads within a business process.

“Business process analysts—now, your business process analysts traditionally report into your business subprocess team leads. These are the people that you pull from the business who have a solid understanding of how the business operates and have what I call a functional/technical acumen. In other words, they understand something about process, how process flows work, how data flows through the processes, and they also know a little bit about the current technology and the systems used to make the processes work in your organization’s current state.

“These resources (as well as your subprocess leads and process lead) should be the ‘top shelf’ resources in your organization for their process area because these are the folks that you’re putting in the driver’s seat to define your company’s future vision for their process area. They are the ones who will be driving the transformation to their peer group once you are ready to go live.

“By the way, the business process analysts, subprocess team leads, and process lead should be dedicated full time to your program.”

Mike nodded. “That’s why we’re pulling them from their current roles and back-filling with the temps, right?”

“Exactly,” Jonas confirmed. “They need to be able to focus on the task at hand—your future operational processes. If these resources still have responsibilities within the organization while trying to do their job on your digital transformation program, your chances of success are significantly hampered.”

“Are those all the business process team roles?” Mike asked.

“Not quite. You’ve also got business process subject matter experts—individuals who have a peripheral relationship to the program. They are the individuals who remain with the business to handle the day-to-day operations of the business while the digital transformation program team defines the new way to do business.

“They are, however, a very important part of the process. These individuals are brought into design workshops for their specialized knowledge on the key processes they know within the organization, as well as what I call their ‘tribal knowledge’ for how things really work. They are also brought in for process reviews, conference room pilots, and testing throughout the program so they will start getting a feeling of what their new world will look like prior to being officially trained on the new solution. These are also going to be the future ‘hearts and minds’ you will be transforming as you are re-envisioning the new way of doing business, so you want to keep them engaged, at least at a peripheral, at what you are doing by showing the prototypes and demos of what the future processes, technologies, and data will look like as you are developing it.

"Then you've got consultants," Jonas added. "Business process consultants (system integrator/independents)—these are specialists who know your industry very well, know your industry process area very well, and know how to use the core technology to be used as part of your digital transformation to make your business process work within your industry. These are the guys that you will come to rely on to help guide you through the challenging road ahead in defining what your future-state organization will look like and how your business processes will operate within your environment."

Mike saw how the various groups of the business process team worked, but he knew it would still take a while to help define how all the resources would be used. "And that's just the first category!" he exclaimed.

Jonas smiled and nodded back, taking another drink. "It's all in that prep kit[1] I gave you, which I jokingly call 'If Jonas were King,'" Jonas said with a big grin. "In all seriousness, the prep kit I gave you sort of lays out my philosophies on how I do this type of work based on a lifetime of experiences solving these types of problems. It should come in handy from time-to-time.

"Next comes technology teams. Although digital transformation initiatives are chartered to re-envision new ways to do business and solve those business problems, there are a lot of technology-related components needed to make it all work. So, when setting up the technology area of your digital transformation

[1] Please visit www.platinumpmo.com/thetransformationbook to access the prep kit.

organization, you will need to consider your technical organization for these areas. If you are using a packaged software application as part of your transformation (likes Salesforce.com and SAP), you are going to have a configuration team with expertise in the software package as well as one or many of the business processes that are within the scope of your program. You will also have multiple development teams depending on what they are developing. For example, custom coding within the package (for functionality you need that does not exist within the package). Business analytics that are a combination of report writers, data scientist, and dashboard developers that help take complicated information and present it in a way to help your leadership make more informed decisions. Then you have your infrastructure team, which is responsible for all of the hardware, networks, and performance of the software. Then you have the security/governance, risk, and compliance group, which is responsible for protecting your software and data and building the appropriate controls so that resources cannot abuse the processes and procedures while keeping your leadership out of jail."

"Jail?" Mike blurted.

"Yes. Don't worry about it, though. But before we get too carried away, I want to mention enterprise architects—those individuals within your current organization who know the current technical architecture of your legacy application environment and are heavily relied upon for working with your digital transformation technical architects to ensure you have a solid future state-integrated architecture.

"Lastly, legacy application teams—these people are a really important part of your digital transformation implementation as you will be working with them a lot when it comes to interfaces and conversions and system retirement strategies. Engage with them early and make sure they know what your future architected solution will look like. Their insights, along with your enterprise architects, will be invaluable."

"And if you consult with them too late, you may not understand the entire image of what we want the company to be after the transformation is implemented," Mike added. "So data teams would be next if I remember correctly."

Jonas smiled. "Right on both counts. So, there are a couple of data team options you should consider: Each process team can have its own data part, or the data team can be its own organization."

Mike asked, "Is one better than the other?"

"Well, 'better' might not be the right word," Jonas said carefully, "but I prefer that the data team be its own entity with its own staff that works hand-in-hand with its process counterparts. If set up properly, your data management team under your data organization can and should be treated like a process team because data management is just as important as order to cash or record to report from an organizational process perspective. But that's not all there is to say about data teams. You need to look at data from a perspective of data migration as well as on-going data governance."

"Data migration teams are primarily focused on the one-time event of getting information out of our legacy environments and into our digital transformation solution environment, right?" Mike asked.

“Yes. There’s a tremendous amount of effort that goes into extracting the millions and millions of records out of dozens if not hundreds of legacy application environments, transforming that data so it meets the rigorous standards necessary to operate in your new environment, and then properly loading that information into your new environment.”

Mike whistled. “Sounds like a tall order.”

Jonas was getting excited. “Now, to properly perform these actions, you are going to need folks who really know their data as well as the business processes that use that data. But at the same time, you are going to want to put some processes in place to ensure that once you convert that data and it is all cleaned, transformed, and ready-to-go in your new environment. You want that data to stay clean and continue ready-for-use in your environment. That requires what is called data governance, and the types of folks who will help you with this really know their data as well as the business processes that use that data. With me so far?”

Mike gestured for Jonas to go on. “So, this group starts off in a dilemma: How do both data migration work and data governance work at the same time? This can even be further complicated depending on the service providers that your organization has contracted to do the work. I’ve have been on clients where they had one vendor responsible for the processes and configuration of the application, three different vendors responsible for the extract, transformation, and load aspects of the data migration, and yet another vendor responsible for establishing and developing the master data governance. That is really complicated stuff!”

“That’s a lot of vendors!” Mike said, understanding what Jonas was getting at.

“The other complication I have traditionally run into is that the organization doesn’t really have many people that really understand how the data is used within the organization. So, the internal selection pool to staff this part of your digital transformation organization is often limited. But I have some good news: A lot of the work that is done to get your organization ready for data migration is a significant subset of the same work that is needed to establish the cleansing and transformation rules for data governance. You may want to consider combining your data migration and data governance into one team. Now, this does have some risks, but after some thorough risk analysis, it may still be your best course of action.”

Mike thought for a moment. “I’ll get back to you when we have a decision.”

“Excellent,” Jonas responded. “So next, is organizational transformation. Depending on your structure, you can have three or four different teams under one umbrella: The communications team is responsible for processes and coordination of program-related internal and external communications. The training team is responsible for creating the program’s end-user training, which could take on several forms, including traditional classroom training, webcast training, and some newer adult techniques called ‘microlearning’ through knowledge as a service (KAAS) solutions. One service that comes to mind is a company called Ringorang.[2] Ringorang is an incredible platform that allows you to take the key concepts taught throughout your traditional classroom and webcast end-user training and positively reinforce the key transformative learning through gamification. It is really a neat concept.

[2] For more information on this tool, please visit www.ringorang.com

“The organizational transformation team is responsible for preparing the organization for fundamental changes that will be taking place as a result of your digital transformation such as organizational structures, process changes, and knowledge transfer. The transformational benefits team—which I have seen as either a sub-team under the OCM track or as a program track all on its own—is responsible for granularizing the benefits identified during the business case, identifying ‘owners’ within the organization who will sign up to be accountable for delivering those benefits, and then working with those leaders to help them transform their organizations.

“Now, these transformation activities are usually integrated with the OCM activities and are designed to not only support the new business processes and technologies but to put these leaders in the best position to achieve the benefits for which the program was chartered and to which they are being held accountable.”

The two just sat in silence for a moment. Jonas finished his water, and Mike looked off into space, processing what Jonas had told him. It was all in Jonas’ prep kit, which he had read—mostly—but to hear this man explain it all from memory was impressive. And a little intimidating. Once again, Mike was struck by the fact that he was finishing in deep waters. However, he also had a guide who really knew his stuff.

Governance Group

“So we’ve talked about a lot of other groups, but what about the leadership?” Mike asked.

“Okay, let’s cover that. The governance group, also known as your digital transformation leadership group, is traditionally made up of three distinct teams: the program management team, the advisory council, and the executive steering committee.”

Mike nodded. “What does the program management team do?”

“The program management team is the group that defines and maintains standards for the program. It strives to standardize and introduce economies of repetition in the execution of the many projects within your digital transformation program. It also maintains the program management office (or PMO), which is the source of documentation, guidance, and metrics on the practice of program management and execution.”

“Is there a standard way of breaking the team into roles?” Mike questioned.

Jonas nodded enthusiastically and took a quick drink to wet his throat, cluing Mike in that he was about to unload a lot of information. “It should have people assigned to the following roles:

- Business Lead—Individual Responsible for ensuring the future business processes are aligned with the organization's (future) strategy and that the ongoing transformation activities are working in concert with that future vision

- Technology Lead—Responsible for ensuring the technology is in place that aligns with the organizations future strategy and supports the future business processes and is positioned to grow into the future as the initial solution matures
- PMO Lead—Architect of the program management office. Responsible for the delivery and management of all program management processes
- Program Financial Advisor—Manages the program budget. Manages vendor contracts.
- Program Administrator—Ensure program policy and procedures are being following
- Program Analytics—Performs program reporting, analytics and health checks of the program
- Project Managers—Individuals managing the streams of work within the program
- System Integrator Partner—Key leadership members of your system integrator team that are helping you deliver the program

“Okay, that makes sense,” Mike said, jotting some notes. “Is that all?”

“Yes, I think so,” Jonas answered, moving on. “Now, the members of the advisory council are traditionally comprised of the business process owners and other executive stakeholders within your organization. They usually:

- Work with the business process teams to resolve complex issues during the program.
- Determine business unit and department readiness.
- Ensure completion of business implementation and roll-out preparation activities.
- Resolve critical business implementation and roll-out issues.
- Oversee data cleansing activities.
- Understand and communicate contingency plans.”

“Excellent, so how do they differ from the Executive Steering Committee?” Mike asked.

Jonas replied, “The Executive Steering Committee is usually made up of the program sponsor, who is usually the CFO or CIO, and other senior executive members of the organization. Now a good start to a program is if the program is actually sponsored by the CEO, COO or CFO and not the CIO, and having your CFO, Arthur Jonesboro, as your sponsor really helps.

“This is in no way meant to slight Charles, his abilities or his clout within the organization, but remember, your digital transformation is a business implementation, so some feel it should be sponsored by a business executive. Traditionally speaking, there is usually a “knee jerk” reaction that your digital transformation is “just another IT project” if it is sponsored by the CIO.”

“So, if it is the CIO, it might get some pushback from other areas of the company?” Mike asked.

“It’s possible,” Jonas conceded. “Digital transformations are not IT implementations! Your CIO should be an integral part of the executive steering committee, but if possible, not the sponsor of the program.”

Mike nodded. “Okay. So you mentioned some of the other c-level members?”

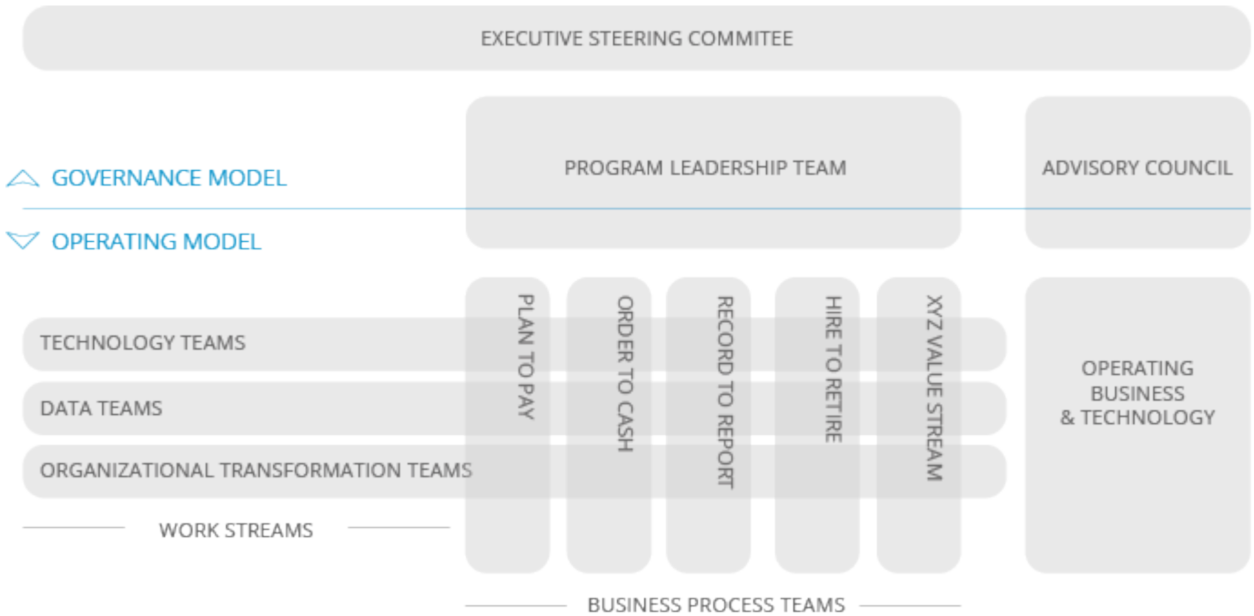
“Yes,” Jonas said, moving away from who should be the sponsor. “other executive-level members of the organization (CXO's and other executive stakeholders) need to be on this team so they can:

- Validate and approve the program charter
- Provide project sponsorship and overall leadership
- Develop broad scope definition and set overall direction, objectives and deadlines
- Provide direction and decisions on key issues, policy, strategic direction and guidance on the program risk mitigation

- Approve project changes that require revising project scope, budget or timeline
- Provide appropriate level of resources required for the project success
- Provide access / bridge to other projects that are aligned to your Digital Transformation Initiative
- Sign off on overall program readiness
- Make go / no go determination
- Approve contingency plans
- Communicate corporate messages to the digital transformation program team members that may have impact on the program

“So how do all these teams and roles work together?” Mike asked.

Jonas pulled out a chart that he had in his folder.



"Here is a good example of what I have used in the past that really has been effective," Jonas answered. "the digital transformation programs that we've seen as being most effective usually use a matrix model for their organization structure.

"From a hierarchy perspective, we split the digital transformation organization into two models, the operational model and the governance model. The operational model is responsible for the day-to-day operations of the program where the governance model is responsible for governing the overall program. From bottom to top, the higher you move up the hierarchy, the more removed you are from the day-to-day operations and the more strategic your lens is when looking at the digital transformation initiative from an organizational perspective."

Mike quickly took notes. He prompted, "I think I am getting it. Please continue Jonas."

"On the Operational side of the equation, you will find our business process teams running vertically in the matrix and our enabling work streams (like our technology, data, and OCM teams) running horizontally across the business process teams. This type of relationship provides integrated continuity within each business process team on how to approach technical conversations such as interface design, data conversations around data migration, and governance and organization conversations around transformation and training. An added bonus is that the enabling teams have a unique perspective of being in the details with each process team thereby having the initial insights at a granular level, should they see inconsistencies on how the business process are executing their work and can identify misalignments that can be reported up to the program management office."

"So, are there any that kind of cross over from one area to another?" Mike asked.

Jonas nodded. "You will also find some of the groups that have their toes in both the operational and governance side of the equation. For example, the program leadership team, which straddles both the governance and operational models, governs the work that the process teams and enabling workstreams performed on a daily basis and usually help drive about 90% of the decisions made for the program.

“Another group that straddles the fence is the advisory council. The process teams are dotted line to advisory council because the business process owners, who are part of the individual business process teams, are also part of the advisory council and use that council to engage their stakeholders back into the organization.”

“So,” Mike mused, “am I guessing correctly that the program leadership team also works closely with the advisory council and consults with them on critical decisions that they have authority to make?”

“Yes,” Jonas confirmed. “And for those decisions where the program leadership and advisory council feel need executive consultation (which should less than ten percent of time), those decisions will be brought to the executive steering committee.”

Jonas got up, tossing his empty plastic bottle into the recycling bin. “We can work more on this later. I know you’ve been with the organization for some time and know a lot of the key personnel. You’ll probably have recommendations for people you’d like to have on the program. You’ll want to work with their bosses to try to secure their services, and it will be a tough sell because the bosses don’t want to lose these resources because they depend on them for the day-to-day operations. This is where the decision board, steering committee, Arthur, and Charles come in to help influence the decision to get the right people for the program—it's the company's future.[4]

"If you'll spend the time to front-end load the organizational structure of your teams," Jonas said as he headed for the door, "you'll reap many rewards later."

"The influence of the decision board, the sponsor, and the CIO will be the key helpers in making this happen and will help navigate the politics. And I have a clear directive," Mike said as he walked Jonas out. "Thank you again—the info you sent is really useful, but when we go over it like this, it really comes together."

Jonas gave him a single, affirming nod. "My pleasure, Mike. See you tomorrow?"

"Bright and early."

***

Mike spent the next few weeks carefully selecting his project leaders and trying to thread his way through the minefield of how the program would ask for qualified people to step out of their existing positions to help the initiative for the company. He knew he'd get pushback because no one wanted to lose their top people. But he knew that if he didn't get some of Extron's best into the program, his task would only be harder.

Despite his volatility, Mike did end up selecting Chet Osborn; the potential upside was just too great for him to ignore. However, Mike was mindful that "potential" was a compliment when one was younger, and the older a person got, the less of a compliment it became. However, he hoped that Chet's full potential could still be realized and that he would be able to harness the man's best for the duration of the transformation.

Chet would lead the integration team. He knew the legacy environments like few others, and he understood integrations. Mike was counting on Chet to bring a lot of technical brilliance to the program and that he would be able to handle the other man's potentially less-developed skills, like communication.

Chet's reputation was for being very smart but not the best with people. His intelligence and work ethic had brought him up quickly, but as he had to handle more and more leadership responsibilities, his rise had faltered due to his abrupt, abrasive personality and poor leadership style. Mike, however, was hopeful that a new opportunity would give Chet his chance to learn from his mistakes and get his career going again.

"I guess we'll find out," Mike said to himself.

Mike had known of Gina Paterno since he started with Extron. She had been around for a few years before he got there, but while Gina was competent and seemed to like her job, she'd never seemed too interested in advancing her career. Her performance reviews seemed to reveal a woman who was afraid to take risks and chose to play it safe. She was consistent, dependable, and deferential.

Mike quickly became convinced that Gina's trouble wasn't vision—it was confidence. He'd tapped Gina to play a pivotal role in the data team, leading the data subject matter experts through the migration and governance aspects of data. She'd been an end-user and in customer service for most of her career, and she knew

Extron’s systems inside and out. Gina had been in the supply chain area of the corporate office for the last few years, so she understood supply chain and manufacturing extremely well. Her role would help ensure that Extron’s customers had a positive experience after the transformation. She’d never been part of a major project, but since she had so much tribal knowledge, Mike thought she’d be a great asset to the program.

He was also hoping that this role would be a steppingstone for what he wanted her to do toward the end of the program, running the operational readiness team. Because Gina came from the business and would learn along the way what program and project life were like, Mike was thinking that she would be a great asset in helping the program prepare for the digital transformation cutover that was eventually going to come down the road. When he sat down to talk with her about it, however, her lack of confidence was pretty evident.

“But I’ve never been on a project before,” she said hesitantly when Mike explained the situation.

“Well, this is a ‘program,’” Mike corrected gently.

“Okay,” Gina replied. “I don’t know how to execute at a program level. I’ve never done this before.”

“I understand,” Mike told her. “I’ve never been a program manager before, either.” He smiled at her. “But I’ve seen your work. I think with a little help, your tribal knowledge could really be of value in leading our data SMEs through the data migration and governance work. Then when the time is right, and we both agree it would be a good fit, I would like to consider transitioning you to leading the organization through our cutover."

"And you want me to do this job?" she asked.

Mike nodded. "Gina, you do great work. And I'm going to help you. Personally. I honestly think you simply need a boost to your confidence in yourself. Anyone who knows our business operations, data, and people as well as you do is going to be a real asset to this team. You just need do what you're good at. I'll help you with the rest—if you're willing."

"You're really offering me this job?" she asked reluctantly.

Mike laughed a little, good-naturedly. "Yes, I really am!"

"And you'll help me?"

"I will," Mike replied.

Gina considered for a moment and then said, "Can I think about it?"

Mike smiled. "Definitely."

She thought about it for two days, and then Mike began a new journey as a leader by starting to mentor Gina. He made it his point to build her confidence, and over the weeks that followed, he began to see a transformation coming over the shy, competent woman.

Mike met with Gina regularly over the ensuing weeks. She was good at her job, no doubt about that, but she was hesitant and unsure of herself. She would defer or bog down on decisions, and Mike guessed that she was probably a high

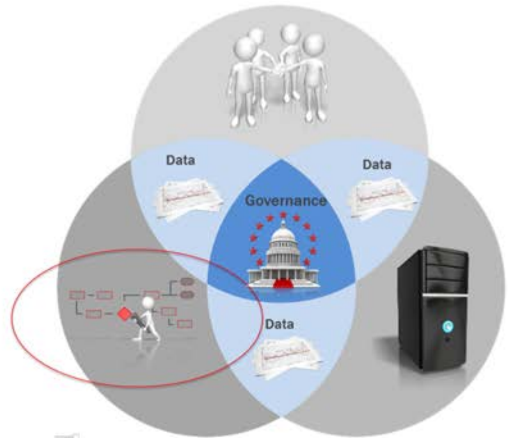
"S"—steadiness—on the DISC personality profile, with some "conscientiousness" thrown in for good measure. A week later, he was proved right when they did the DISC personality profiles as a leadership team-building exercise, and Mike's understanding of what she needed to make decisions grew. He started listening to her differently, and his coaching evolved to fit his new understanding of her personality, motivators, and strengths.

His leadership team was coming together.

Chapter 4: Organization & Teams

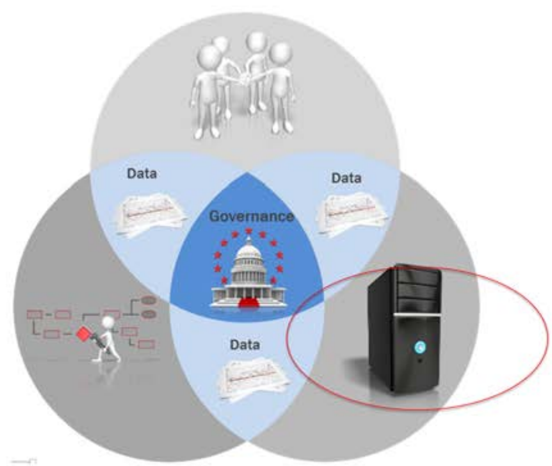
Business Process Team Roles

- Process Owner
- Process Leader
- Sub-Process Leads
- Process Analysts
- Subject Matter Experts
- Consultants



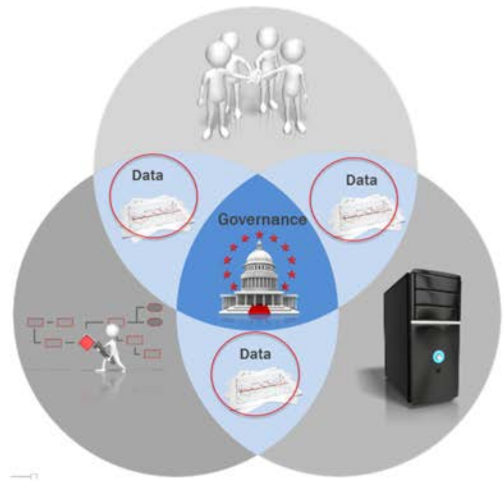
Technology Teams

- Functional/Configuration
- Multiple Development Teams (RICEFWM)
- Business/Data Analytics
- Infrastructure
- Security
- Enterprise Architects
- Legacy Application Teams



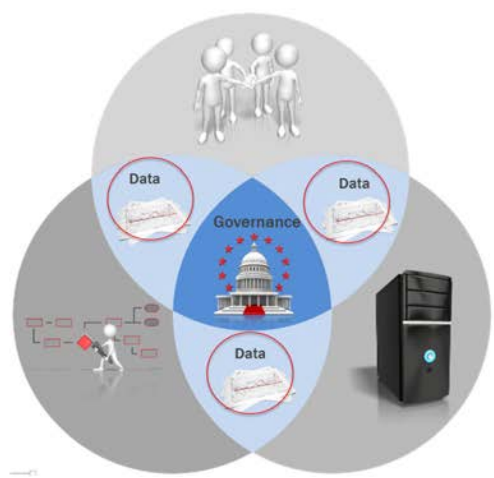
Data Migration Team Roles

- Three Separate Streams of Work:
  - Extract Team (Legacy)
  - Transform Team (Data Consultants)
  - Load Team (Target Transformation Environment)
- Each of these teams follow a similar role structure:
  - Data Steward
  - Data Manager
  - Data Subject Matter Experts
  - Functional Requirements Designer
  - Technical Requirements Designer
  - Developer



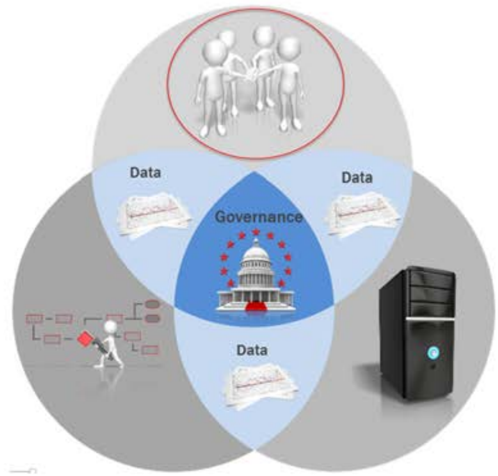
Data Migration Team Roles

- Each of these teams follow a similar role structure:
  - Data Steward
  - Data Manager
  - Data Subject Matter Experts
  - Functional Requirements Designer
  - Technical Requirements Designer
  - Developer



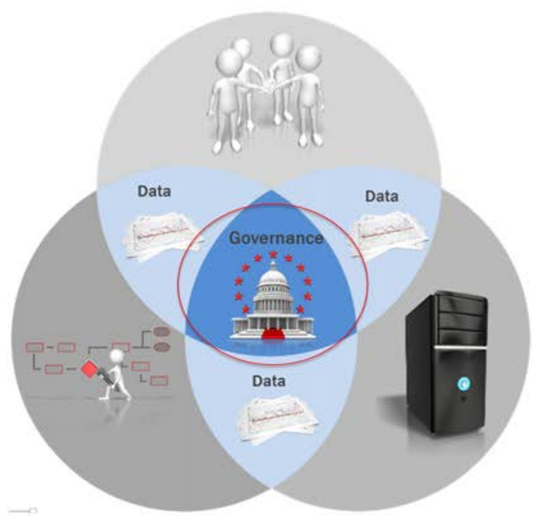
Organization Transformation (People)

- Communications Team
  - Communications Manager
  - Content Developers
- Training Team
  - Training Manager
  - Training Developers
- Organization Change Management Team
  - OCM Manager
  - Knowledge Transfer Agents
  - Organizational Design Liaison
  - Change Leads
  - Transformation Advisors
  - Benefits Realization Leads

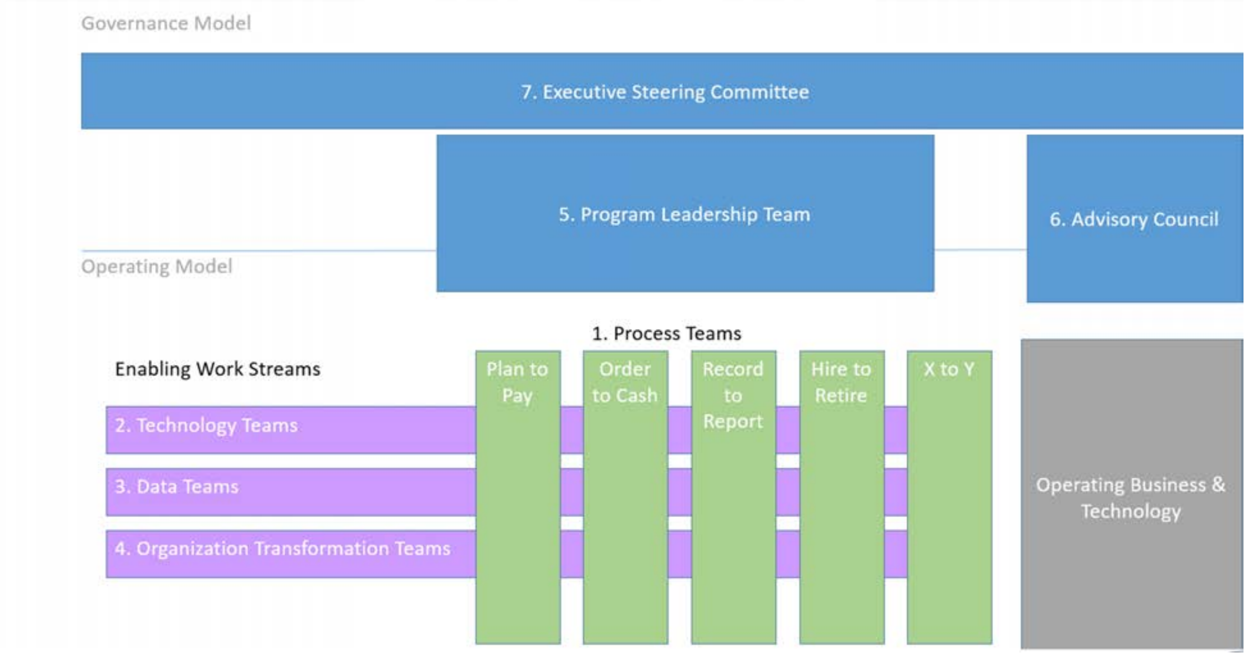


Program Leadership (Governance)

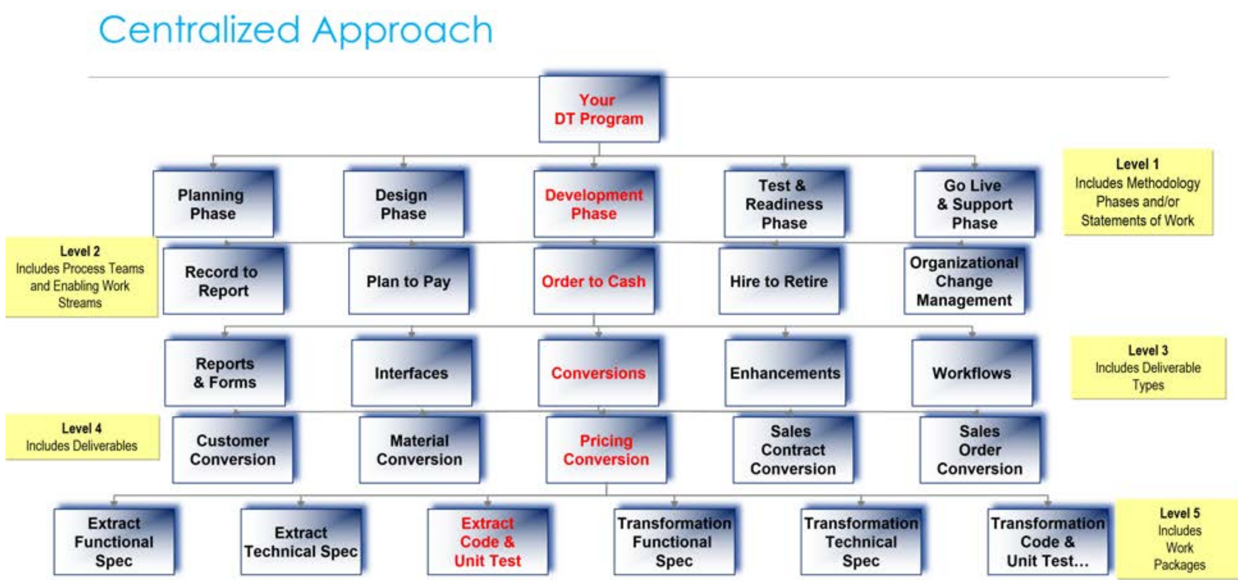
- Program Management Team:
  - Business Lead
  - Technology Lead
  - PMO Lead
  - Program Financial Advisor
  - Program Administrator
  - Program Analytics
  - Project Managers
  - System Integrator Partners
- Advisory Council:
  - Business Process Owners
  - Executive Stakeholders
- Executive Steering Committee:
  - Program Sponsor (CFO or CIO)
  - CXO's
  - Executive Stakeholders

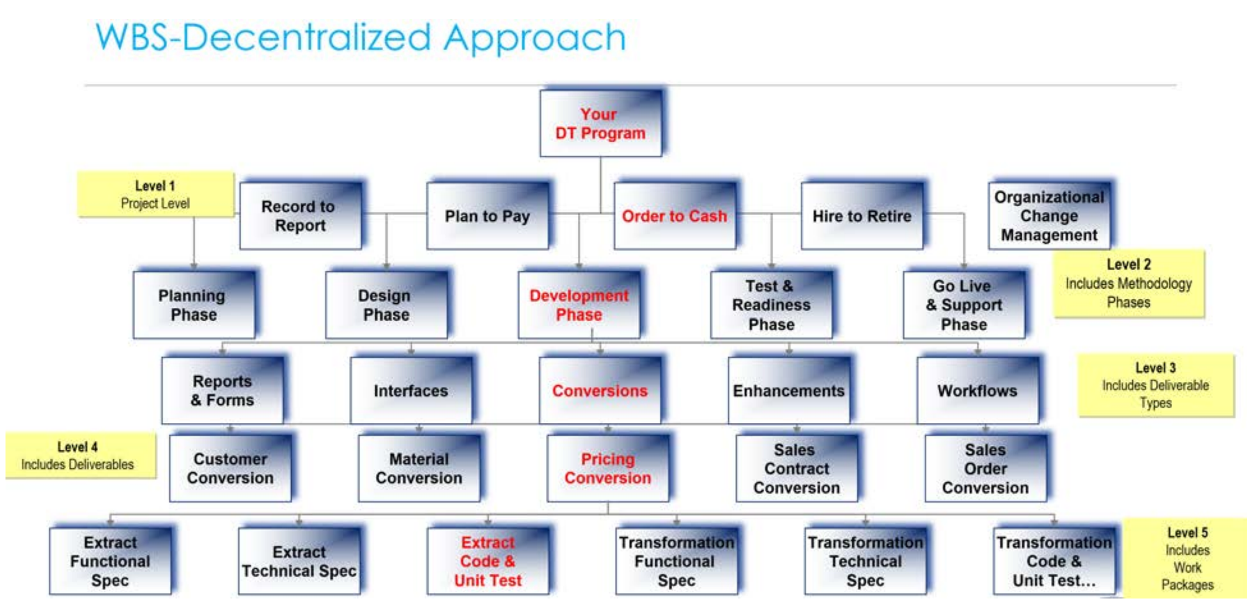


Digital Transformation Program @ Work



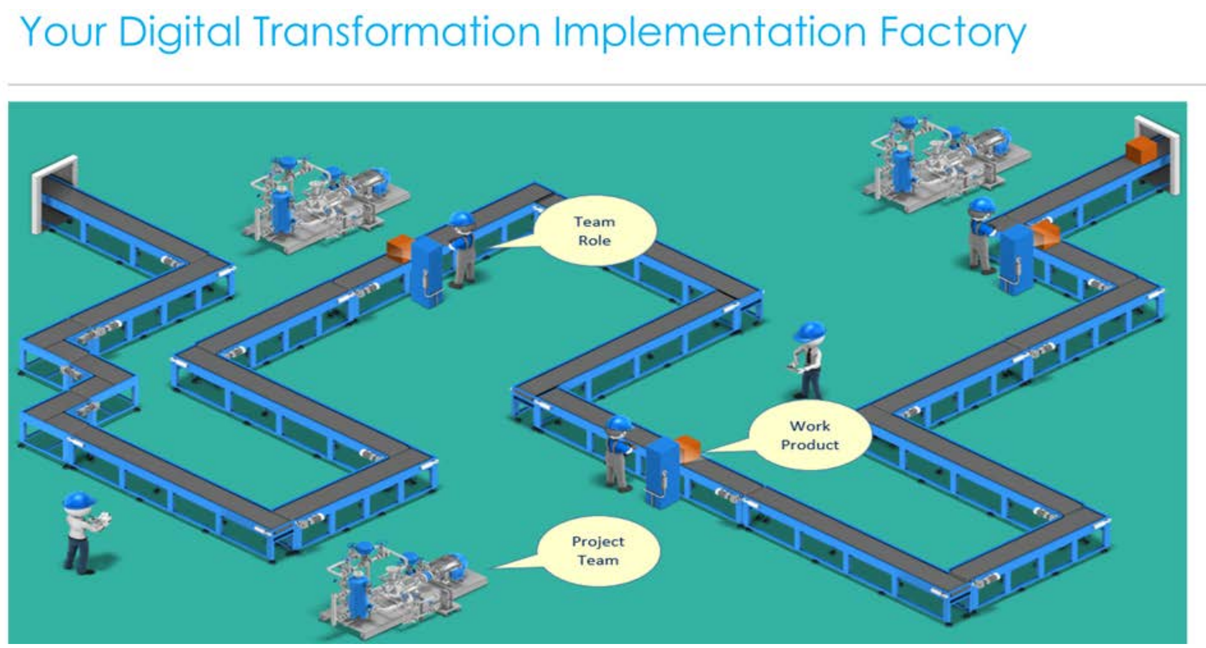
Chapter 6: Centralized vs Decentralized

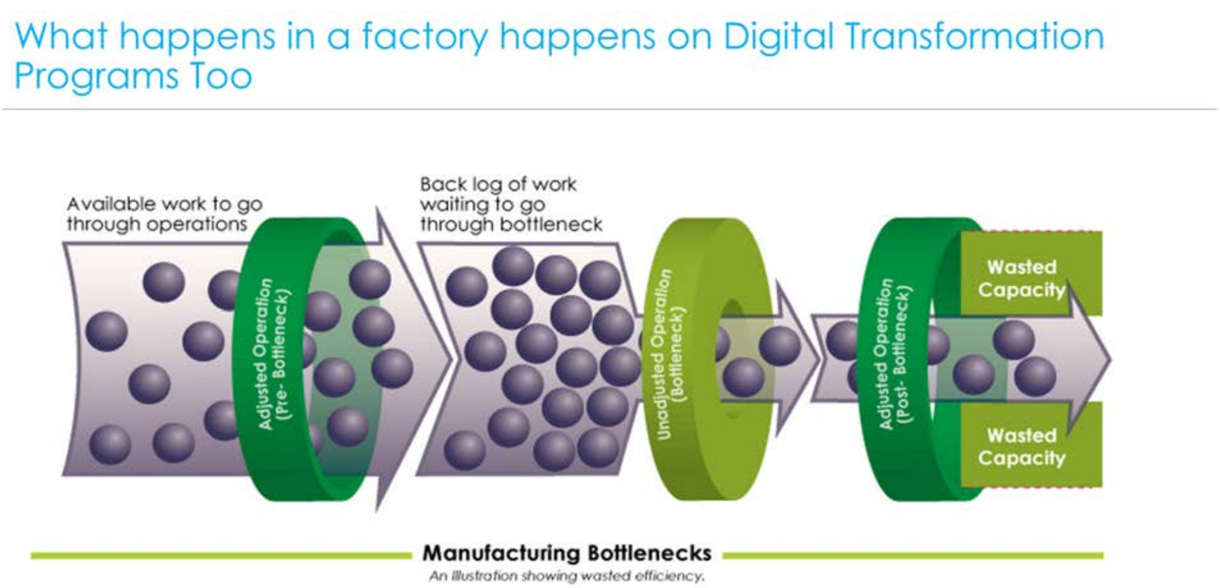




Chapter 7: The Critical Chain

Run Your DIGITAL TRANSFORMATION Program Like You Would Run a Factory





Let's look at this illustration here and assume that the little balls are our widget, the circular items are our work centers.

As you can see in the middle of this diagram that second work center cannot keep up with the flow of product that its predecessor work center is providing causing a backup in production.

This is causing wasted capacity in the third work center which is causing wasted production to the entire operation and company.

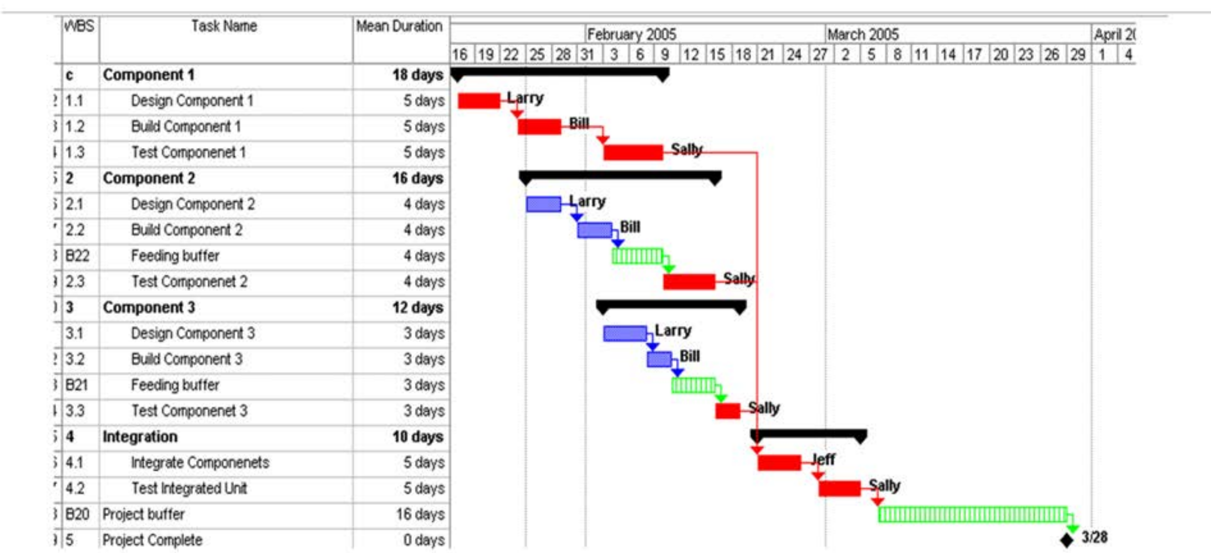
A product can only be produced as fast as its slowest process or machine.

So, as the Superintendent, what would you do?

You would find a way to somehow improve the throughput of that second work center so that it is running as fast as the next slowest work center.

You will continue to do this exercise until you get to optimal performance which is the perfect balance of speed, throughput, cost and quality.

Critical Chain Example



Let's look at a simple example of a project plan:

Using the Critical Chain Technique, you can quickly see the Critical Chain of this Program and that chain runs through the resource named Sally.

Like our machine number 2 in our manufacturing plant, Sally is the most overloaded resource on this program and the program can only complete as fast as Sally can produce. You can tell that Sally is the bottleneck because her name shows up in red and there are no lines connecting her task indicating that there is a physical work package dependency. The delays to the project are strictly based on the fact that Sally cannot do three things at once.

So, as a Program Leader I am going to do everything I can think of to rearrange Sally's work so that Program can perform at an optimal capacity: completing faster, delivering quality scope and not costing any more than what was budgeted.

Now here is the best part, now that you know which roles or resources your critical chain goes through, you as the program leader, and associated work package Accountables can do everything in your power to protect those resources and ensuring that they are delivering "focused effort" on their work packages.

With you and your Project Managers help, you will remove the obstacles of status reports, unnecessary meetings and all the dozens of other non-value added things that seem to consume our days and help your bottlenecked resources focus on the important things---delivering the critical chain.

Chapter 7a- Watergile

Author Note: This chapter did not make the final version of the book, however, I really liked it and thought it would be of some value to help you better determine when to use waterfall, when to use agile and when to use a hybrid approach that I refer to as “Watergile”.

“The next day, after Mike had finished digesting everything Jonas had taught him about program structures and critical chain, he had complete forgot about another topic of conversation that Jonas had mentioned weeks ago, which he called ‘Watergile,’ and how that fit into the program structures and critical chain philosophies, so he decided to give his old new friend Jonas a call.

“Hey, Jonas, it’s Mike. Got a few minutes to web-chat?”

“Anything for your Mike,” Jonas replied.

“I was thinking about our conversation yesterday and something you had brought up a few weeks earlier on the topic of ‘Watergile’ and how—if it does at all—it fits into what we discussed yesterday about program structures and critical chain project management.”

“Ah, great question. I am glad you thought to ask about it,” Jonas replied. “So, for most of the last forty years or so, most software development was performed using an approach referred to as the waterfall approach to systems development.”

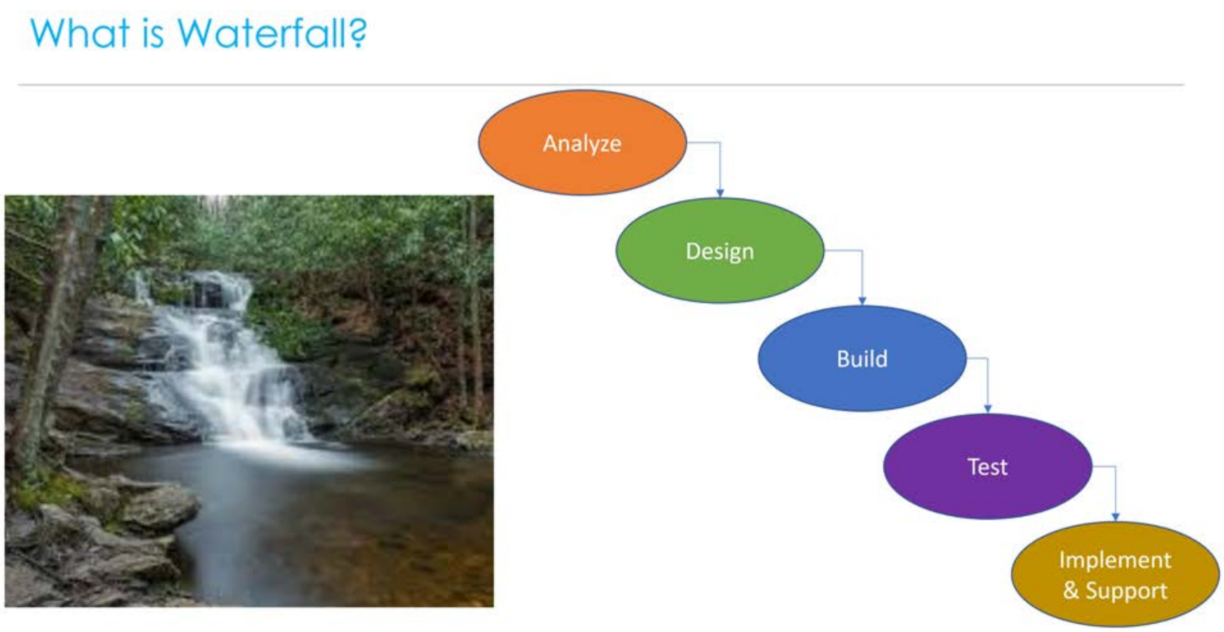
“Right, I’m sort of familiar with waterfall.”

“Okay, well, let me tell you what I know to see if it’s helpful. So, the waterfall software development process was first introduced by Dr. Winston W. Royce in a paper published in 1970. The waterfall model emphasizes that a logical progression of steps be taken throughout the software development life cycle (SDLC), much like the cascading steps down an incremental waterfall."

"I guess 'waterfall' sounded better than 'stairway,'" Mike joked.

"Right!" Jonas said, laughing. "Up until the last fifteen years or so, most software-enabled projects and programs followed the waterfall approach to their software development. They traditionally have an upfront set of contiguous phases that take your through the development and implementation of your software product. Here, hold on a minute, let me bring something up."

Jonas shared his screen as he googled something. Mike could see what he was talking about as Jonas went on,



“The ‘analyze phase’ represents the work performed by analysts to gather the requirements and requirement specifications for the future software product.

“The ‘design phase’ represents the work performed by the design team to take those requirements and start modeling them into technical specifications that will be used to develop the future software product.

“The ‘build phase’ represents the work performed by the software developers to take those technical specifications and develop software that performs work in accordance with those technical specifications.

“The ‘test phase’ represents the work performed by the test team and future users to take the results of the developed software and test to ensure that the developed software works in accordance with the requirements agreed to in the analysis phase.

“And, finally, the ‘implement phase’ represents the work performed by the implementation team to prepare the organization for the new software, migrate from and decommission outdated software, and to start the execution of the new software in the organization.”

“Okay,” Mike said, “I see how the various phases of development work. What about after the software goes live?”

Jonas answered, “Once the software is released to the organization, there is ongoing support of the product which can be considered a phase, albeit a continuous one.”

“Seems like this approach has been the industry-standard approach for software delivery for decades and has worked very well for those decades and the people that have used it,” Mike commented. “So why do you think we’re moving away from it to Agile in some areas?”

“It does have its challenges,” Jonas conceded.

“Like what?”

“For example, if the requirements change after the analyze phase, then this may have an exponential effect to the rest of the phases. Let’s use the example of building a house. You have provided your architect with your specifications for your master bathroom. Your architect shows you the design the master bathroom in your house from his blueprint. There is something you don’t like with your original requirements and you ask the architect to change the blueprint. That is a fairly inexpensive change. Now, let’s say they have started construction and have roughed in the plumbing and now you change your requirements and want the shower moved—well that is a very expensive change. Now, imagine doing the same thing at your final house walkthrough before your close on your house—major-major change and major-major expense.”

Mike winced at the thought of that change. “Would not want to pay for that!”

“The same concept occurs in software development projects using the waterfall method. The people defining the requirements did the best job they could in developing their requirements based on the information they knew at that moment in time. As the project matures, they learn more and start understanding things better, and therefore their knowledge is better, which will prompt changes to their requirements. It can become a very expensive proposition.

"Most people who work on software projects would agree that requirements rarely stay fixed for the duration of their program or project. This means that either the process is interrupted to account for the new or changing requirements (which causes annoyance and delays to the program or project) or that the business agrees not to make any requirements changes for the duration of the program or project. This is typically an unrealistic approach because of the ever-changing dynamics of a business and its market conditions."

Mike agreed, "Yeah, I can't see how that would work. It would be like developing the program with your mouth taped shut."

"Adapting to changing business dynamics (and subsequent software requirement changes) can have a significant positive impact on the original business case and return on investment for the program. Unfortunately, what happens is those requests are made by the business without a complete understanding of the downstream impacts to those changing requirements. These changes in turn can cause major delays to the implementation date, which may or may not reduce or eliminate the additional return on investment."

"I'd think it would also cause a lot of problems between the company and their developers," Mike said thoughtfully.

"It causes inherent friction between the business and the software teams when the goal should be working as one team, working together to achieve the best possible outcome."

“That’s a pretty major drawback.”

“And that’s not all. Another challenge with the waterfall approach is that the bulk of the defects and issues are not identified until you reach the testing phase.”

“Seems like that would put your launch date a risk.”

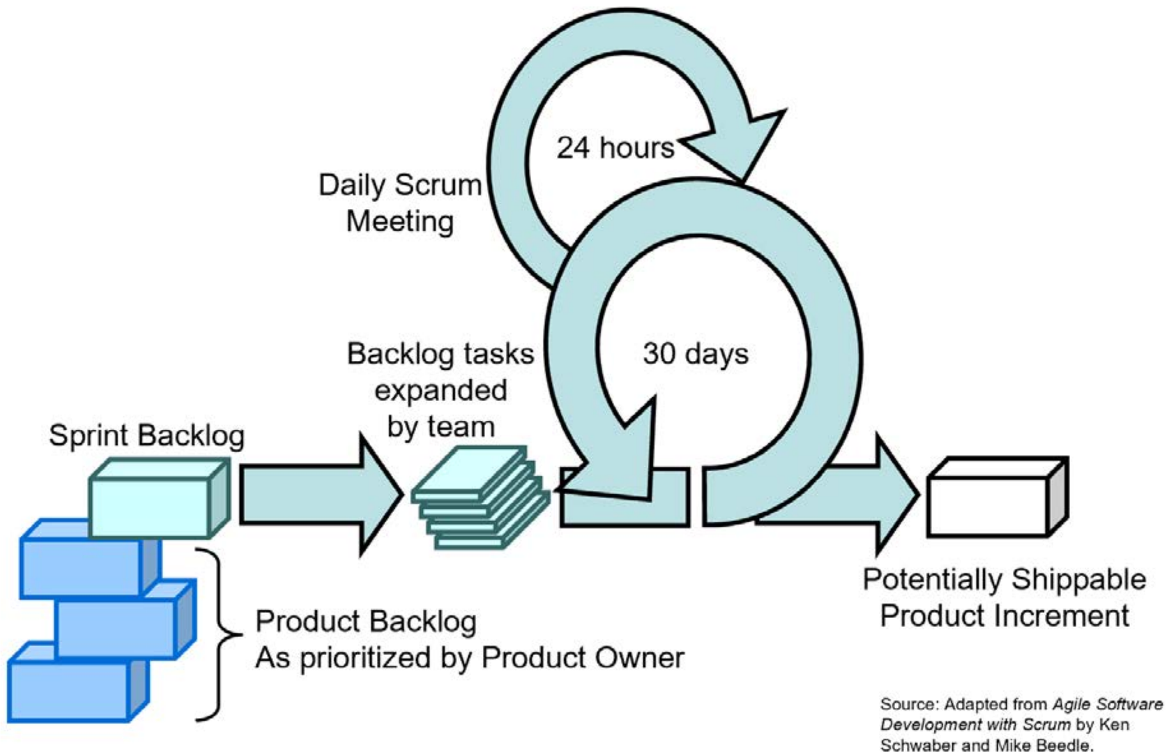
Jonas nodded. “More time is required to fix defects and retest the solution than what was anticipated in your program or project plan. It is really difficult to forecast the number and complexity of the defects that will be encountered in this phase—but speaking from experience it’s usually underestimated. This traditionally leads to high stress, long hours, infighting, and overall low morale.”

“So, how long did it take people to create alternatives?” Mike asked.

“After decades of running programs and projects this way, some folks made it their mission to try to find alternative ways to remove these impediments to software delivery. A variety of methods started appearing in the late 1990’s under what is now referred to as the Agile approach to software delivery. These thought leaders started coming up with ways to iterate the software development lifecycle into smaller chunks of value instead of one big bang.”

Mike said, “I have a lot more experience with agile, but if you don’t mind, I’d like your perspective on the differences.”

“Sure,” Jonas said. He typed on his keyboard and brought up a graphic for the agile approach.



“As opposed to upfront phases with lots of upfront requirements gathering, iterative development contains mini phases of requirements, design, build, test, and possible implementation within a number of weeks. This approach allows the organization to release a few features of the software early in order to start gaining a return on their investment. They also get to discover some potential issues early in the process and change requirement far more often than the waterfall approach.”

“I think that working in iteration also reduces the probability of bad multi-tasking because there are not as many activities going on at the same time, thereby allowing resources to focus on fewer things at one time,” Mike said. “So are there multiple versions of agile?”

Jonas nodded. “There were a variety of these agile methods and frameworks popping up in the late 1990’s including: XP, Crystal, DSD, FDD, and Scrum. In 2001, a group of thought leaders (including the developers of Scrum) got together and formed a group called the 'Object Mentor Group.' They gathered at Snowbird ski resort in Utah to try to come up with some key underlying principles for all of these methods. The result was the Agile Manifesto."

"Sounds impressive," Mike quipped.

"The Agile Manifesto reads as follows: 'We are uncovering better ways of developing software by doing it and helping others do it. Through this work we have come to value:

- Individuals and interactions over processes and tools
- Working software over comprehensive documentation
- Customer collaboration over contract negotiation
- Responding to change over following a plan

Jonas brought up a list of the principles they'd agreed to on the screen. "This manifesto was accompanied by a set of twelve principles:

1. Customer satisfaction through early and continuous software delivery – Customers are happier when they receive working software at regular intervals, rather than waiting extended periods of time between releases.
2. Accommodate changing requirements throughout the development process – The ability to avoid delays when a requirement or feature request changes.

3. Frequent delivery of working software – Scrum accommodates this principle since the team operates in software sprints or iterations that ensure regular delivery of working software.
4. Collaboration between the business stakeholders and developers throughout the project – Better decisions are made when the business and technical team are aligned.
5. Support, trust, and motivate the people involved – Motivated teams are more likely to deliver their best work than unhappy teams.
6. Enable face-to-face interactions – Communication is more successful when development teams are co-located.
7. Working software is the primary measure of progress – Delivering functional software to the customer is the ultimate factor that measures progress.
8. Agile processes to support a consistent development pace – Teams establish a repeatable and maintainable speed at which they can deliver working software, and they repeat it with each release.
9. Attention to technical detail and design enhances agility – The right skills and good design ensures the team can maintain the pace, constantly improve the product, and sustain change.
10. Simplicity – Develop just enough to get the job done for right now.
11. Self-organizing teams encourage great architectures, requirements, and designs – Skilled and motivated team members who have decision-making power, take ownership, communicate regularly with other team members, and share ideas that deliver quality products.

12. Regular reflections on how to become more effective – Self-improvement, process improvement, advancing skills, and techniques help team members work more efficiently.

“That seems like a really effective foundation to work from when you put it that way,” Mike said.

Jonas agreed, “So over the past decade or so the agile approach to software and product development has gained a lot of traction—so much traction that it is as if it is the only way for a company to be successful with their software or product implementations. It’s as if everything that the industry did with waterfall development before agile was useless.”

“Judging by your tone,” Mike said, “I’m thinking you’re about to say something different.”

“You know me too well!” Jonas said with a laugh. “In my humble opinion, this mindset is like throwing the baby out with the bathwater. You shouldn’t just abandon all of the principles that guided the industry for more than forty years. There are still some very important situations or initiatives, like yours, Mike, where waterfall methods are still very appropriate to use.”

“You’re not just saying that because you’re old school, are you?” Mike asked with a laugh.

“Well, old school is the best school,” Jonas joked, “But guys like me who have been around the block a few times have seen firsthand how organizations have struggled adopting an agile approach to large-scale Enterprise-wide digital transformations like you have at Extron. I saw the same thing a Tuterro, and I watched how that program struggled."

"So what do you suggest? Going back to waterfall?" Mike asked skeptically.

Jonas shook his head. "Being the curious guy that I am, and always wanting to stay on the forefront of my profession, I really started researching the effects of pure-play agile approaches on large enterprise transformation initiatives. I also reached out to a lot of my colleagues who were going through the same challenges that I saw the team at Tuterro going through, just to get some perspective and potential advice.

"They all said pretty much the same thing—agile is not a one-size fits all solution. It's great for small and medium size projects, but the larger the project, the more interconnections that are occurring on a project, the more difficult agile becomes. So, taking that feedback I dug around some more to see if I can find some concrete research that backs up what my colleagues had indicated and what I experienced at Tuterro, and I found some interesting stuff."

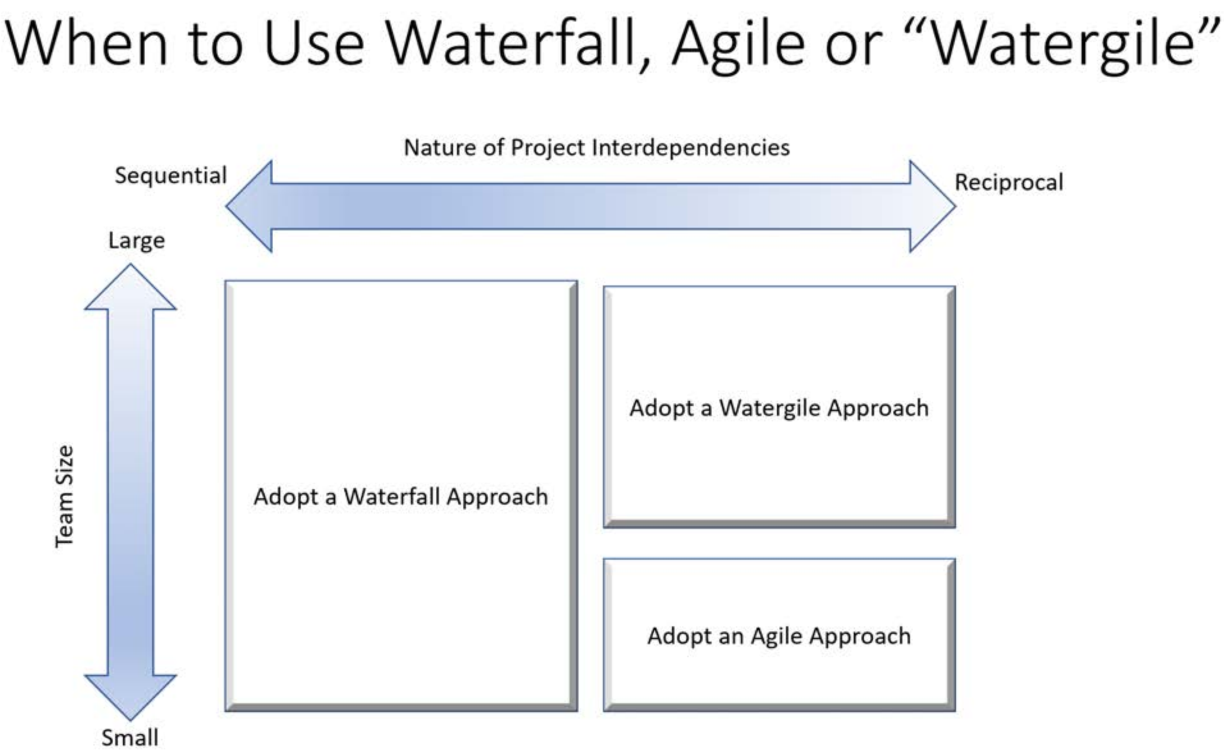
"Like what?"

"There was a research study headed by Dr. Jordan Barlow on the effects of agile development on large organizations, and their research concluded findings similar to me and my colleagues' experiences in the field. Give me a moment and let me share it with you."

Jonas brought up some material for Mike to see on his screen. "There was this chart that they described in the study that gave the reader a pretty good idea as to when they should use a waterfall approach, when the should use an agile approach, or when they should use a hybrid approach of the two, which I refer to as a ‘watergile’ approach.

“Ah, now it comes out—this is what I was wondering about,” Mike said. “So how do you know when to use what?”

“Take a look at this chart,” Jonas offered.



Jonas went on, “According to Barlow’s research, when most project or program interdependencies are sequential regardless of team size, the program should adopt a plan-driven (waterfall) methodology and invest in technologies that will support the planning process.

“When the nature of project interdependencies is reciprocal, meaning that program and subsequent project teams need to communicate and interface closely with each other, your choices are then derived based on the size of the program or projects.”

“Makes sense,” Mike said.

“When project team or program size is small, then you can use an agile approach. Agile methods, with iterative cycles and frequent communication among team members and stakeholders, are well-suited to small teams with highly reciprocal interdependencies.”

“But we’re not small.”

“No, you’re not,” Jonas agreed. “When your project team or program is large, then you should use a ‘Watergile’ approach where you will use plan-driven techniques for any program modules that have mostly sequential interdependencies but agile techniques for most of the project components that have reciprocal interdependencies.”

“Do you think it will confuse everyone to use different approaches?” Mike asked.

“As we discussed before, talk with your vendors. Find out what they like to do. What is their recommended methodology? I am all for hybrids because they allow you to put your players in the best position for success and not force them into an approach where they cannot provide you maximum value.”

“So, what do you recommend for our transformation specifically?”

"Personally speaking, I think there are some great attributes with agile. Communication is one of the big ones, because the teams are smaller, more focused and they working as a unit and should be communicating all of the time. However, when you have thirty to fifty teams all working to deliver one enterprise digital transformation operating model, that is going to require more structured planning, documentation, and governance."

"I can see that. We're going to have a lot of teams!" Mike commented.

"I also like the whole Scrum framework," Jonas continued. "Scrum takes its name from the analogy to Rugby where the team works together in a chaotic, changing environment to keep control of a ball. This can be compared to a team working in a chaotic environment of ever-changing requirements while keeping control of the project.

"Scrum consists of self-organizing, cross functional teams. These teams contain a group of people with different areas of expertise but work together for the same outcome. A Scrum team has three different roles:

1. The product owner is one and only one person and is responsible for maximizing the value of the product resulting from the work of the development team. The product owner is the sole person responsible for managing the product backlog.
2. The development team consists of professionals who do the work of delivering a potentially releasable increment of 'Done' product at the end of each sprint. Development teams are structured and empowered to organize and manage their own work. The resulting

synergies optimizes the development team's overall efficiency and effectiveness.

3. The Scrum master is responsible for promoting and supporting Scrum as defined in the Scrum guide. Scrum masters do this by helping everyone understand Scrum theory, practice, rules, and values.

"The teams work in iterations, which allows the business flexibility to change their requirements but still gives the development the certainty it needs to deliver a working piece of the product. This is one key thing that makes Scrum powerful, but it's something that needs to be carefully monitored and communicated across all of the other product owners and program leaders within the overall program."

"Otherwise it just gets crazy," Mike observed.

"Through transparency your team builds trust and defines what 'done' means. Once the team agrees on transparency, they agree to consistently check up on progress through inspection and make improvements on what they have seen. Adaptation is the team's ability to learn from one another and improve the overall process and delivery. Through this process, the team is improving time and time again before, during and after the release of a product. This is something that is very hard to do (maybe even impossible) with the waterfall method."

"You mentioned critical chain when you were talking about Watergile before. Can you tell me more about that?" Mike asked.

"In terms of critical chain, the critical chain approach works great in all three circumstances. With highly structured projects, it can really help you project the number or resources that will be needed at the strategic times throughout the program to complete the entire program. When doing agile projects, you set up your sprint, perform a critical chain assessment on that sprint to give you a statistical probability that you have enough capacity to actually finish the sprint. If not, you take something out of the sprint. If you have too much capacity, you go back to your product backlog and add something else to your sprint and recalculate your chain to get the right balance of throughput to realistically accomplish your sprint.

"Does this clear things up for you a bit, Mike?" Jonas finished.

"Yes, thank you, Jonas. I really appreciate it," Mike said. "I'd like to schedule a meeting with the program leaders to help roll this out. I'd like you to attend. And can you send me links to these graphics you've shown me?"

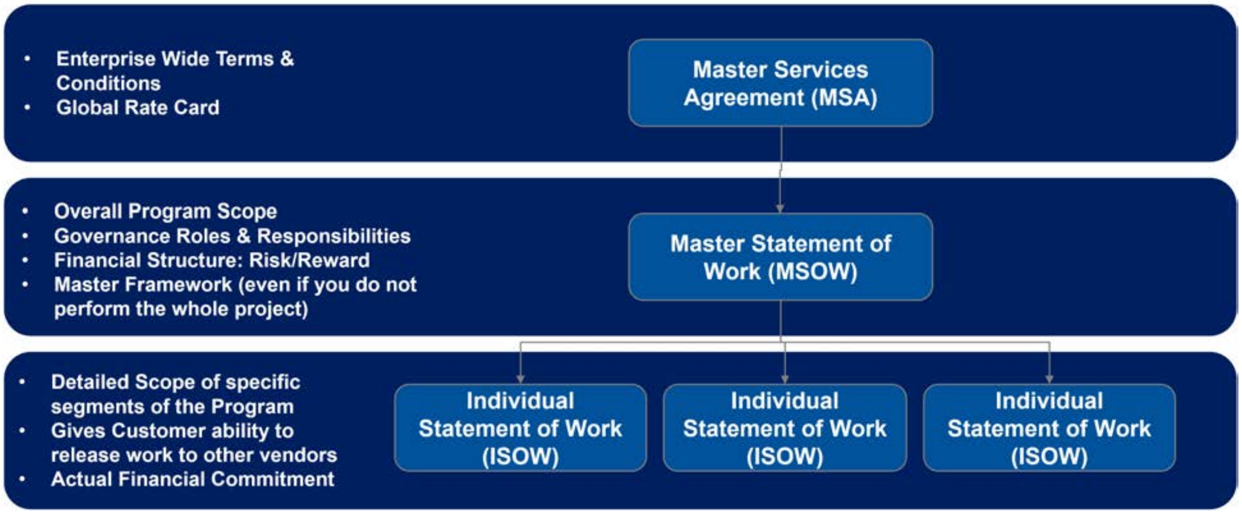
"Definitely!" Jonas answered. "I think that if you stay flexible and use a hybrid approach where its most beneficial, you'll see some big benefits."

"I agree," Mike said. "I'll see you tomorrow. Thanks again!"

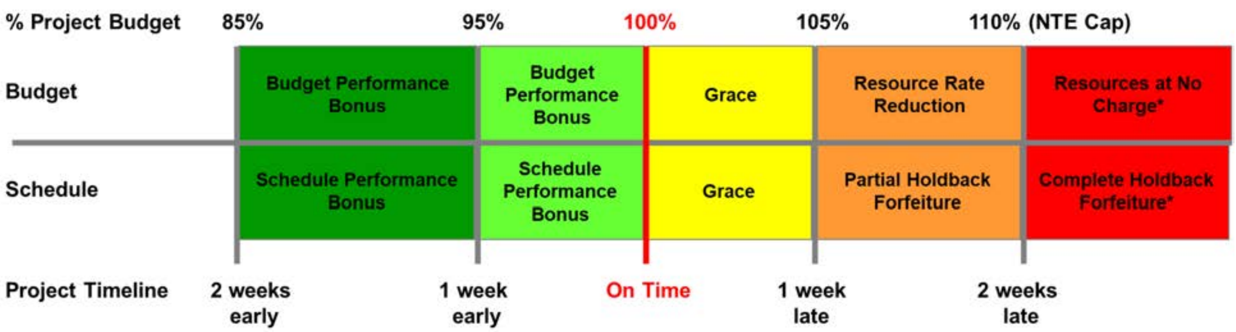
"Any time!"

Chapter 10: Risk/Reward Financial Models

Typical Agreement Structure



Shared Risk/Reward Financial Model (Example)



Incents Vendor to deliver early/ below budget for significantly higher profit

Protects Customer from significant budget overages

Budget Performance

Strategy	Description	Example
Project Budget Baseline with Not To Exceed (NTE) Cap	Time & Materials (T&M) Project Budget Baseline with a Not-To-Exceed (NTE) Cap based on the percentage of the Project Budget Baseline	Project Budget Baseline= $1,000,000 NTE Cap = 110% Maximum Project Fees= $1,100,000
Expense Cap	Expenses capped as a Not-To-Exceed (NTE) percentage of the Project Budget Baseline and billed on actual expenses incurred	Project Budget Baseline= $1,000,000 Expense Cap = 18% Maximum Project Fees= $180,000

Project Fees = Project Baseline with NTE Cap + Expense Cap

Schedule Performance: Risk/Reward Structure

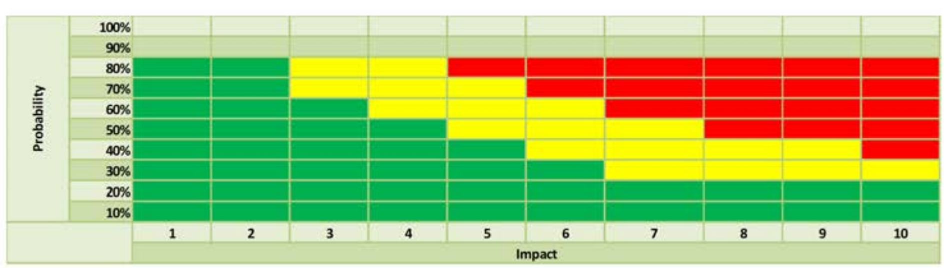
Strategy	Description	Example
Holdback	A percentage of each invoice would be held back and subject to forfeiture for late deliveries of milestones	Invoice Amount= $100,000 Holdback %= 10% Holdback Amount= $10,000
Risk: Holdback Forfeiture	Holdback amount that would be subject to forfeiture for late delivery of milestones based on the forfeiture schedule • Forfeiture amount may be earned back by getting project back on schedule and delivering subsequent milestones on time	Holdback Amount- $10,000 **Late** / **On-Time** / **1 wk late** / **2 wks late** Forfeiture Schedule: 0% / 50% / 100% Forfeited Amount: $0 / $5,000 / $10,000
Reward: Early Delivery Bonus	Bonus payment that Vendor would receive if the project is completed ahead of schedule	Project Schedule Baseline= On Time Early Delivery = 1wk early / 2 wks early Bonus Payment= Fixed Fee Amt / Fixed Fee Amt

Chapter 14: Risk Identification

Risk Identification: Breaking down the Big Rocks

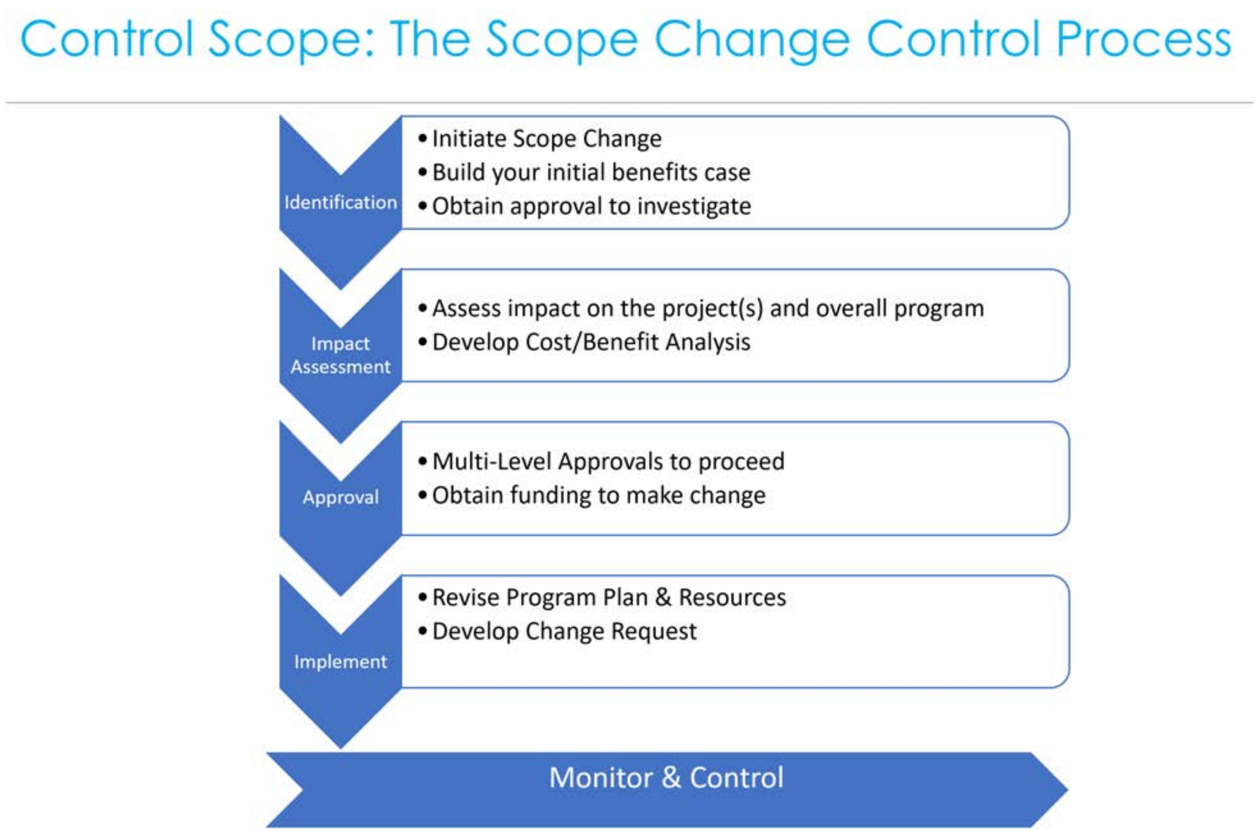
Risk Exposure Scoring

Risk Impact * Risk Probability = **Risk Exposure**



- **Risk Exposure**
  - Critical Risks (Red) should receive the greatest attention since these risks (if they occur) will have the greatest negative impact on the program
    - Require Mitigation and Contingency Plan
  - Major Risks (Yellow) should have a mitigation plan and should be monitored
  - Minor Risks (Green) should be monitored ongoing for potential increase in risk

Chapter 17: Change Control Process



Tensions Flare- Another Day at the Office

Author Note: Here is another chapter that did not make the final version of the book that I thought could be of value to you.

“What do you mean it’s not that important? My department can’t function without this!” Melissa Kotel was yelling at Jeffrey Allen as Mike walked into his office one dreary and rainy Tuesday morning. Marilyn Kelliher from Customer Service was standing right there with them, looking annoyed.

“You think that’s more important than maximizing our inventory turns and minimizing our inventory costs? Accounting is Accounting—I’m trying to save the company a fortune here!” Jeffrey retorted.

Marilyn Kelliher interjected, “Well, what about Customer Service—you know, the ones who actually interact with our customers? We need our enhancement to answer our customers inquiries real-time on the first call and get the answer right the first time, not bounce them around our call-tree until they finally get to a person who knows the answer off the top of their heads!”

“Well, good morning guys,” Mike said, deadpan. “Seems like we’ve all rolled out of the wrong side of the bed this morning. I assume you’re having this conversation in front of my office door waiting for me?”

Jeffrey looked slightly embarrassed, Melissa was still angry, and Marilyn just scowled at the other two. “Hi,” Jeffrey answered. “Yes, we need to talk. All of us need to talk about getting the things that we need into this final product.”

“Okay,” Mike answered, “well why don’t you all step on in and let’s have a civil conversation on the matter?”

Marilyn, Jeffrey, and Melissa walked into Mike’s office ready to plead their cases as to why their change had to be done or they would not be able to function in this new operating model.

“So, what seems to be the problem?” Mike asked.

“Let me start, Mike,” Melissa piped up right away. “Listen, I’ve seen the latest program financials and the remaining budget to get this program done, so I know funding is starting to get tight, but if I don’t get these enhancements to build these workflows in order to facilitate GAAP-required separation of duties, we run the risk of being out of GAAP compliance, which could have legal ramifications to Extron.”

Jeffrey interjected, “Mike, Mike, Mike, while I appreciate Melissa’s concerns about her gap workflows…”

“It’s GAAP, Jeffrey—G-A-A-P. GAAP!” Melissa cut in.

“My apologies, Ms. Kotel,” Jeffrey replied cooly. “May I continue?” She replied with something like a grunt. “While I appreciate Melissa’s concerns about her GAAP Workflows, I am talking about real savings with my inventory enhancement request. My department can save this company millions with this change.”

“Ah-hem,” Marilyn cleared her throat. “That’s all well and good, but none of that will matter if we don’t have any customers to serve. If we don’t get this AI knowledge center up and running, I’m afraid we run the risk of losing hundreds if not thousands of customers who will go to other providers who can provide the support they now demanding."

Mike held up his hand to forestall any more comments, which Melissa and Jeffrey looked ready to add. "Well, you all make compelling arguments, and I appreciate you all passionately expressing your points of view on each matter. Let me ask you a couple of questions before we go any farther. Are these requests entered in the change request log?

"Yep," Melissa replied quickly.

"I'm still working on it," Jeffrey said more slowly, followed by, "Me too," from Marilyn.

"Okay. Well, the first thing you guys need to do is get these requests into the change request log along with your business case as to why you feel that these changes are warranted at this stage of the program.

"Now, as Melissa mentioned, our program budget is burning a little hot right now, so there are not a lot of discretionary funds laying around to satisfy change requests. I've given the change control board some guidelines on how need to address changes from this point forward.

"In addition to the financial business cases you will be documenting in your change requests, we also need to look at each change moving forward from some fundamental perspectives."

Jeffrey looked like he was going to say something, so Mike held up his hand again and the other man kept quiet.

“First: Is the change due to a statutory requirement, which are those changes that have requirements which are applicable by virtue of law enacted by the government? Those have the highest priority moving forward.” Melissa looked vindicated and almost turned an “I told you so” look at Jeffrey.

“Second,” Mike went on, “Is the change due to contractual agreement with one of our customers, vendors or other key stakeholders? If so, then this will weigh heavy on the change control board decision.

“Third: Is the change just part of the nature of our business, or will it put our business at risk? And if it puts our business at risk, what is the probability of that risk occurring, and what would be the impact should that risk materialize? If you are building your case on this, I would highly recommend building out a risk in our risk register as well and tying your change request to that risk.”

Marilyn nodded at this last part as Mike went on, “Fourth: Is the change a customer desire? If so, document that desire, document the customer or customer group that is asking for the change. You may want to do some analysis on what value those customers bring to Extron. The higher the profit margin associated to the customers asking for the change, the higher the influence on the change review board.

“Fifth: Is the change due to an internal policy? If so, document accordingly. The decision board will have the weigh the pros and cons depending on the policy infraction.

“And finally: Is the change due to an internal preference? These changes need to be documented with the appropriate business case and value to the business as a whole. The change review board is meticulously evaluating all of the changes that are sitting in their evaluation queue, and they will make the best decisions they can—decisions they have been empowered to make."

"Thanks, Mike," Jeffrey said. "I appreciate you going over this with us, and we all appreciate what the program is going through."

"I know, Jeffrey. And, again, I appreciate your passion for your sides of the business. You all are championing for what you believe is the best for your line of business and for Extron. But here is the deal—build the case, make it as compelling as possible, and follow the guidelines I gave you. This program is a living, breathing, organism that will continue to grow and evolve for years to come. If we do our jobs right—and I know we will—there will be budget to fund enhancements to this solution for years to come. If we don't get to this now, there is a good chance we can get to it within the first year after our initial go live. Sound fair?"

He looked at each person in turn to see they agreed. Mike could feel the tensions in the group subsiding—Melissa especially seemed to have come down off her ledge and de-escalated dramatically. Jeffrey looked frustrated, and Marilyn seemed thoughtful.

"Now," Mike said, "if you'll excuse me, I need to get ready for a meeting. If you have more questions, feel free to ask me individually."

All three nodded, stood, and headed for the door. "Oh," Mike said to their backs, "and let's try to use our inside voices, shall we?"

Chapter 20: Quality Assurance Process