USING LARGE-SOLUTION SAFE IN A HIGH-PROFILE U.S. CIVILIAN PUBLIC SECTOR AGENCY

Abstract

While the Scaled Agile Framework (SAFe) is about 10 to 12 years old now, which is about 80 in dog years with the rapid rate of technological change, U.S. government agencies are flocking to SAFe for IT-intensive acquisitions in droves. U.S. Fortune 500 firms were some of the early adopters of SAFe and it is rapidly growing throughout the international community, including Europe, Africa, South America, and East Asia. There are over 500,000 SAFe certified professionals throughout the world and SAFe is rapidly eclipsing and is a serious threat to professional management and technical certification bodies that have been around for over 50 years. This threat is so real that traditional certification bodies stopped giving lip service to lean and agile thinking and made large investments in SAFe's competitors due to its proprietary nature. SAFe is in its golden age and is rapidly becoming the gold standard for lean-thinking, rapid business experiments, DevOps, user experience design, and even cloud development. SAFe is applicable to a wide variety of products and services that have yet to discover and apply its principles in addition to IT-intensive work. SAFe will eventually surpass Scrum which is now well-accepted in global non-IT operations domains.

There are over 170 U.S. government agencies responsible for spending and administering the annual $4 trillion U.S. economy. U.S. government agencies are neither bleeding-edge Silicon Valley startups nor profit-driven Fortune 500 firms. That is, they are slow on the uptake, organizational change takes decades, IT infrastructures are obsolete and can't keep pace, and they are slow to adopt modern information technology management practices such as Lean, Agile, Scrum, DevOps, SAFe, Cloud Computing, etc. Many in fact are still optimizing manufacturing era management practices from the 1970s such as process and document-intensive waterfall lifecycles, integrated master scheduling, earned value management, enterprise architecture, manual testing, on-prem data centers, and monolithic functionally organized autonomously budgeted silos with little to no collaboration from concept to operations. Traditional thinking emerged in an era when computers were primitive, slow, expensive, fragile, and scarce. Therefore, managers and engineers put 80% of their time, energy, and resources over-thinking, over-planning, and over-scoping their solutions before they considered even touching a computer, which teams were forced to share for only a few minutes at time.

However, high-profile government agencies are under intense scrutiny and pressure to quickly modernize IT practices from executive, legislative, and judicial branches, congressional committees, and the government accountability office. The budgets, acquisitions, and programs associated with new mission critical IT capabilities are shrinking, forced to do more with less, and often directed to apply lean and agile thinking principles, practices, and tools. Many, but not all U.S. government agencies now mandate the use of agile practices as part of their internal policies, contracts, and individual business unit methodologies. However, the bigger they are the harder they fall, and consistent application of lean and agile methods is yet to be achieved in even the most visionary U.S. government agencies. To many agencies, SAFe is still new, unfamiliar, and even controversial as an alternative to basic Agile Methods, Scrum, Kanban, DevOps, and even Cloud Computing initiatives. Its safe-to-say, pun intended, that only the boldest and bravest agencies have exclusively adopted SAFe for managing their IT acquisition portfolios, programs, and projects. However, even these are hit-and-miss, spotty, experimental, inconsistent, misapplied, controversial, or plain wrong!

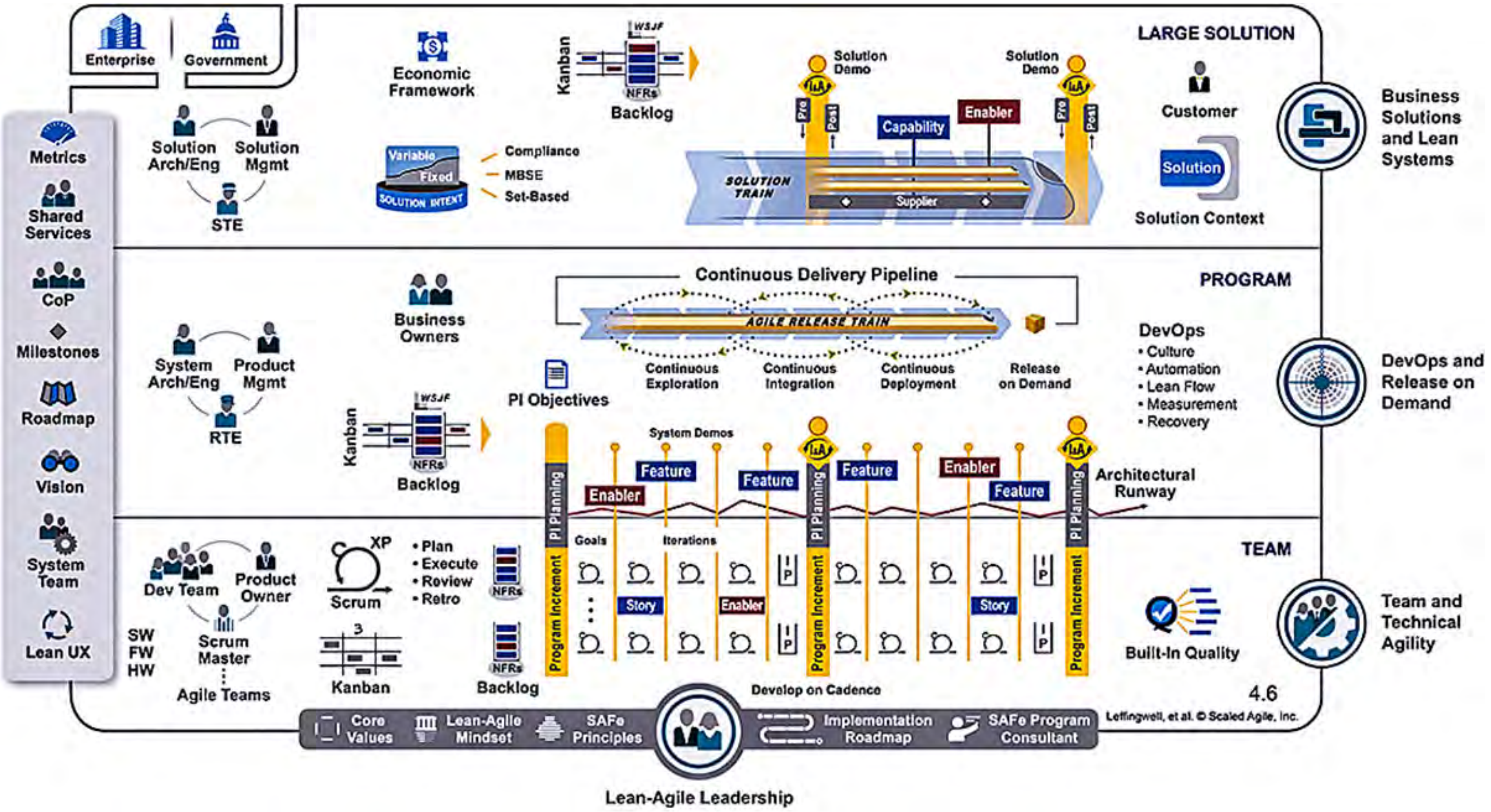
This is a case study of a U.S. government agency administering trillions of dollars constituting the entire U.S. economy applying SAFe to a major enterprise transformation. Its purpose was to migrate to public cloud services, create a virtual casefile system, and consolidate decades of legacy systems. It chose large-solution SAFe for its resemblance to traditional lifecycles, focus on system architecture, coordination of multiple agile release trains (ARTs), and availability of an application lifecycle management (ALM) tool. Lean and agile thinking promised more for less, this acquisition was four years late and overbudget, agile methods had a successful track record in this arena, and there was internal pressure to adopt basic agile practices. The SAFe implementation roadmap was followed, change agents were established, everyone was trained, ARTs were created, implementations plans developed, program increment (PI) planning done, and SAFe ceremonies ritualistically applied. Culture change comes last not first in SAFe, and this agency was steeped 1970s waterfall lifecycles, lean and agile thinking was in short supply, the program was over-scoped, expectations were high, time was short, and many linear processes and artifacts were baked into SAFe PIs.

Introduction

Although medium-sized in nature, this U.S. government agency allocates about 36% of its annual $11 billion budget to IT modernization, amounting to $4 billion per year. About 13% of its annual IT budget is applied to IT "modernization," also known as new product and service development, amounting to $500 million per year. It's not exactly clear what happens to the other 87% of its annual IT budget amounting to $3.5 billion annually—Perhaps operations and maintenance? Approximately 20% or $100 million per year is applied to its latest Large Solution SAFe transformation to migrate to public cloud services, implement an enterprise virtual casefile system, and consolidate 60 legacy applications into it. Prior to the SAFe rollout, approximately $85 million was spent over four years on just the enterprise virtual casefile system portion to no avail. New technology stacks were chosen on this go around to mitigate the earlier issues, including a commercial-off-the-shelf business process management tool, Amazon Web Services, and the latest information technologies such as containerized microservices among many others. Large Solution SAFe was also chosen to organize the restart of the four-year-old program along with a SAFe-based agile ALM toolset.

With a $4 billion annual IT budget, a very poor track record of IT acquisitions spanning decades, and the onerous responsibility of administering the entire $4 trillion annual U.S. government budget, this 70-year-old government agency has been under intense public scrutiny for many decades. So much so, that the U.S. government even allocated an extra $100 million in annual IT expenditures for federally funded research and development centers (FFRDCs) to oversee, manage, review, and audit its IT technical acquisition practices to no avail. Its personnel are among the best and the brightest the nation has to offer and many of their IT acquisition challenges are directly linked to antiquated IT acquisition practices from the 1950s, 1960s, and 1970s. It's enterprise engineering life cycle (ELC) is a direct knockoff of the U.S. Navy's MIL-STD-1679 (1975) and DoD-STD-2167 (1985), which saddle heavyweight military-grade documents on lightweight (agile) IT acquisitions. It ritualistically applies integrated master scheduling practices that emerged in 1958 and enterprise architecture practices from 1987. Every new IT acquisition must undergo intense system architecture scrutiny in order to integrate into its single monolithic enterprise architecture.

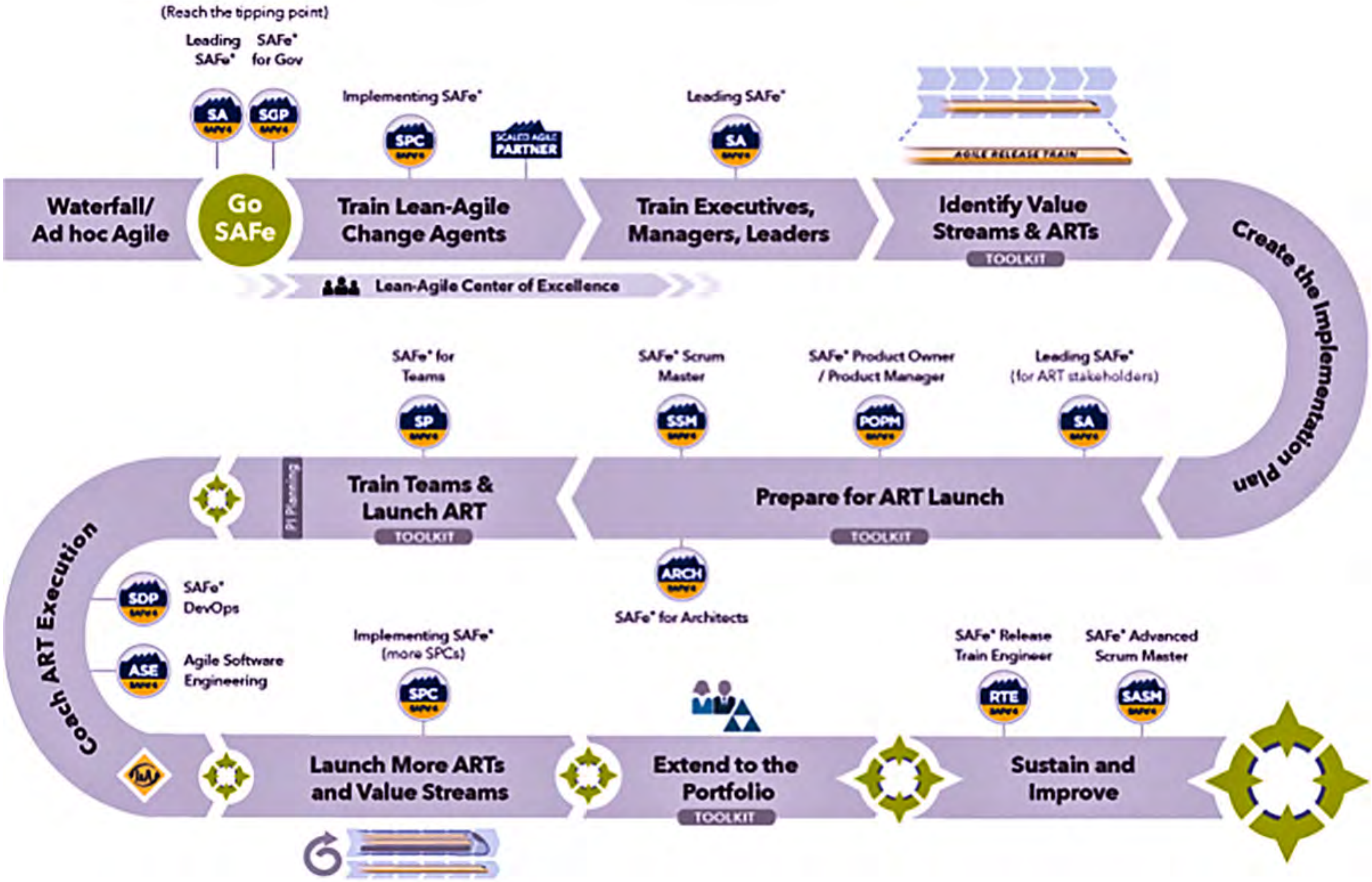
Lean thinking emerged around 1990 to cap off 50 year old best practices pioneered by Toyota, which inspired the creators of agile methods to adapt these practices to small IT project teams in the mid-1990s. SAFe emerged in the mid-2000s as a formalization of agile program and project management practices—That is, scale agile methods to multi-team programs with a smattering of multi-program Kanbans at the portfolio-level. In other words, SAFe's creators wanted to scale lean and agile thinking principles to large IT portfolios, programs, and teams. Small project agile methods such as Scrum recommended team sizes of seven to nine people. However, SAFe boasted one of its agile programs consists of 125 people, and its portfolios could contain many of these programs, called agile release trains (ARTs)! SAFe continued evolving, adding a large solution layer for architecting systems of systems, infusing even more lean thinking principles, DevOps, and even business experiments. Although SAFe's initial beginning was rather slow amidst competing models, Fortune 500 firms began adopting SAFe for its intuitive face validity (graphical model) and certification training programs. U.S. government agencies began flocking to SAFe from 2017 to 2019.



Footnote. *As illustrated above center, continuous exploration (business experiments), continuous integration (builds), and continuous deployment, and system releases can take place at any point during SAFe program execution*.

Program Increment Implementation

This agency was urgent to restart its four-year-old $85 million IT acquisition as a three-pronged $250 million enterprise transformation initiative. Given this crisis, journey to agile methods, and integrated master schedule culture, it followed the SAFe implementation roadmap to the tee (see below). It trained and certified its core leadership and management teams as SAFe Program Consultants (SPCs), identified its ART, created an implementation plan, prepared for ART launch, and trained its initial two teams in SAFe as well. This also included a battery of self-paced instructional modules on its SAFe-based ALM. It conducted a one-day PI planning event for its initial PI spanning eight weeks. The initial PI consisted of onboarding personnel in its immense administrative paperwork process, badging everyone, issuing laptops and network connectivity, establishing its architectural runway, and beginning work on key enablers. Rigorous integrated master schedules, roadmaps, and SAFe ceremonies were applied as well. These included Scrum of Scrums, Product Owner Sync, Release Management Meetings, System Demos, Iteration Planning, Daily Standups, Iteration Demos, Iteration Retrospectives, Inspect and Adapt, and PI-preplanning among many others.


(Reach the tipping point)
Leading SAFe®
SAFe® for Gov
SA
SGP
Implementing SAFe®
SPC
SCALED AGILE PARTNER
Leading SAFe®
SA
AGILE RELEASE TRAIN
Waterfall/ Ad hoc Agile
Go SAFe
Train Lean-Agile Change Agents
Train Executives, Managers, Leaders
Identify Value Streams & ARTs
TOOLKIT
Lean-Agile Center of Excellence
Create the Implementation Plan
SAFe® for Teams
SP
SAFe® Scrum Master
SSM
SAFe® Product Owner / Product Manager
POPM
Leading SAFe® (for ART stakeholders)
SA
PI Planning
Train Teams & Launch ART
TOOLKIT
Prepare for ART Launch
TOOLKIT
ARCH
SAFe® for Architects
Coach ART Execution
SDP
SAFe® DevOps
ASE
Agile Software Engineering
Implementing SAFe® (more SPCs)
SPC
SAFe® Release Train Engineer
RTE
SAFe® Advanced Scrum Master
SASM
Launch More ARTs and Value Streams
Extend to the Portfolio
TOOLKIT
Sustain and Improve

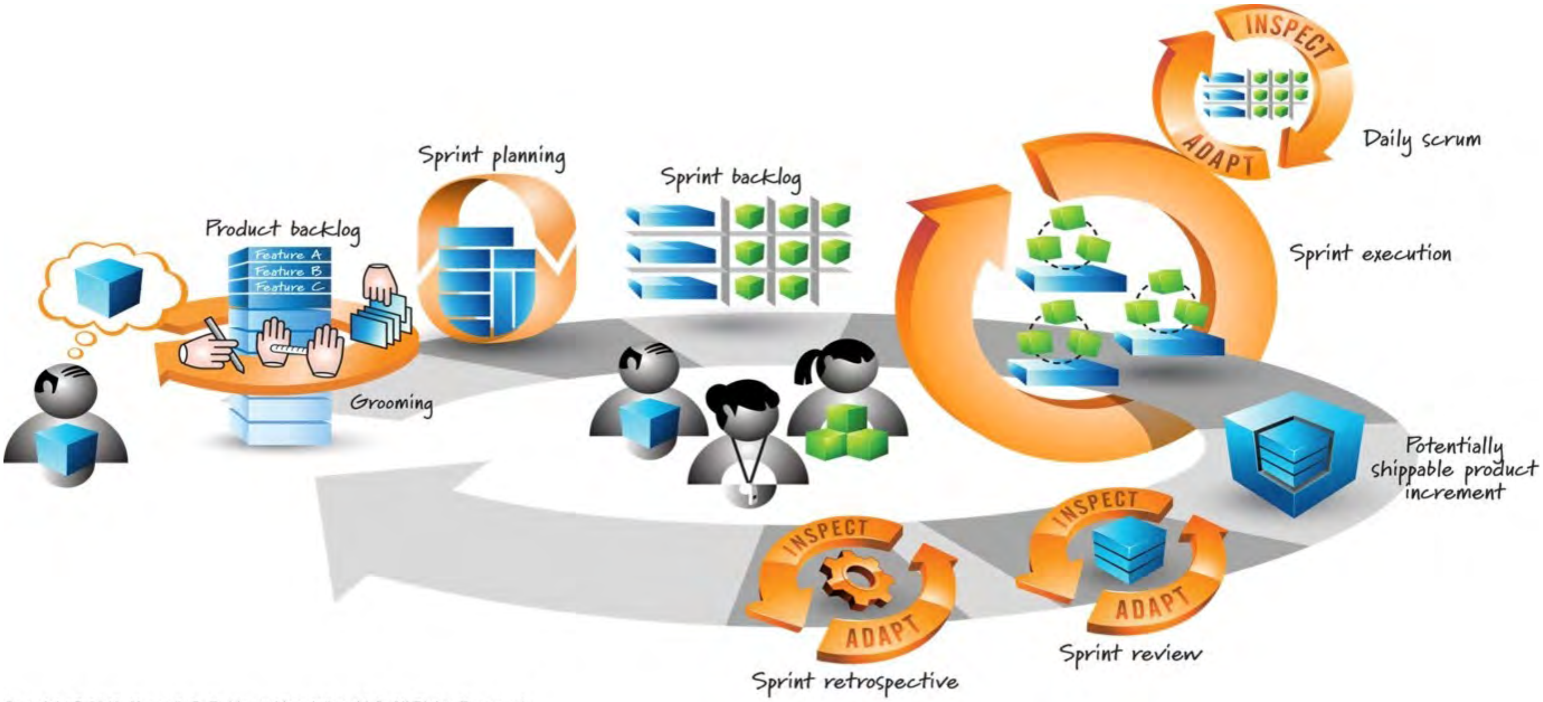
This transformation program was a commercial-off-the-shelf (COTS) integration or technology insertion project. It's not a custom software development project with a lot of homegrown programming. The initial four-year $85 million project stalled for use of a non-scalable casefile technology among many other factors. The restart involved selecting a scalable COTS solution that operates in a cloud environment. This was a three-pronged project to migrate the new enterprise casefile system to AWS and integrate and interface more than 60 legacy systems out of more than 200. A possible fourth prong was to upgrade its enterprise agile ALM to the latest version, merge its legacy ELC waterfall lifecycle into it, and configure, provide training, and use it to manage future large-scale SAFe acquisitions. Therefore, the program consisted of interfacing its network to AWS, acquiring a suitable COTS tool, installing and configuring it in a public cloud, interfacing legacy systems to the COTS tool, and establishing its latest agile ALM as the SAFe standard. Apart from obvious over-scoping, there were other system issues related to its functional organization, aggressive integrated master scheduling culture, and failure to establish centralized PMOs going back decades.

Immense effort went into using integrated master schedules and roadmaps for predicting the exact delivery date for its initial release following five 8-week PIs down to the minute and second. There was also an urgency to automatically generate SAFe roadmaps, visualizations, reports, and metrics from its SAFe-based ALM tool for performance management. The majority of the program's participants were nationally distributed, so few team members had ever met one another, save daily online meetings, phone calls, and other teleconferences. Given the agency's deeply engrained traditional thinking culture, it codified its waterfall ELC into the five major SAFe PIs for the first year of the restart. The first PI consisted of program startup and architectural runway activities, the second PI would begin system design, the third PI would consist of COTS installation, and the latter PIs of integration, operation, and certification. The government agency was functionally organized, so business owners, enterprise architects, methodology and tooling, testing, program management, and security engineering personnel belonged to separate departments. Other than the primary two feature teams, 80% of the program's participants belonged to functionally disparate divisions.

There was a lot of enthusiasm restarting the program with SAFe, and the core leadership and program management team was convinced they were going to succeed. There was little overall direct experience with SAFe, including Scrummasters and developers, however culture change comes last not first with SAFe, so that shouldn't have dissuaded people too much. There weren't any full-time SAFe coaches at the government level, although the delivery team contractors did have a few. However, most of the direction was coming from the government which overtly stated it didn't want or need any SAFe coaches. That being said, without direct SAFe experience nor understanding of lean thinking fundamentals, managing this program to a successful end was going to be a large hurdle to overcome. For one thing, the majority of the program's effort was in producing its historical waterfall documentation, which was heavier than normal. Of course, few PIs were actually dedicated to hands-on cloud, COTS installation, legacy system integration, etc. The cloud team was on a separate contract with little to no communication or synchronization. And, of course the government was convinced that the key to program success was full utilization of the core delivery teams.

Program Increment Execution

The first Large Solution SAFe program increment began in earnest with the use of Scrum for implementing its two-week sprints or iterations. Iteration planning, daily standups, sprint demos, sprint retrospectives, system demos, product owner syncs, RTE syncs, scrum of scrums, release management, PI preplanning, integrated master scheduling, roadmapping, agile ALM planning, and governance meetings began in earnest. Although not all program participants were required to attend every meeting, breaking up a working day with three or four meetings causes so much multi-tasking context switching that is humanly impossible to complete any productive thought, analysis, or work. Two more teams were added after the first PI planning event, an architectural runway and an elaboration (design) team. In fact, these latter two teams constituted half of the productive workforce. They often met multiple times per week in all day marathons to convince functionally organized enterprise and system architects to think in terms of vertical capabilities and features. Although the agency's architecture function agreed to follow SAFe's concepts of vertical capabilities and features, it was not easy for them to think in these terms, which caused a lot of internal friction.



The first PI along with rest of the program consisted of establishing a large integrated master schedule with dozens if not hundreds of tasks to track every possible administrative activity. This included acquisition, contracting, contractor administration, IT infrastructure and network access, badging, laptops, tooling, security and privacy, COTS lifecycle compliance, and the procurement of the COTS itself which was one of the major lynchpins of program success. Other components included setting up public cloud services, interfacing the agency's network to the public cloud, satiating the ELC waterfall and COTS lifecycle policymakers, setting up the agile ALM, working with functional architects, complying with usability and accessibility guidelines, and, of course, complying with security and privacy requirements. Dozens of people attended frequent congressional oversight, governance, roadmapping, release management, enterprise lifecycle, agile ALM, risk management, and integrated master schedule meetings interspersed among the already overpopulated SAFe and Scrum ceremonies or meetings to scrutinize every task on the integrated master schedule to look for signs of the slightest schedule slip, risk, or possibility of smoke, spark, flame, fire, or meltdown.

Traditional project managers believe in full utilization and overallocation of individuals and teams in order to complete aggressive schedules, over-scoped programs, unrealistic deadlines, underbudgeting, and, of course get their money's worth out of people. Lean mathematics dating back 70 years clearly show that the closer one gets to full capacity, the slower the schedule becomes, and productivity even comes a screeching halt. Traditional managers exacerbate this with multi-tasking, over-allocation, part-time resources, and multi-program management. That is, in order to achieve its goals of full utilization, they'll place many unrelated tasks on individual and team calendars. Unrelated tasks incur such a large penalty on context switching that is also slows and halts productivity as well. Many of the architectural runway, elaboration, and myriad of other meetings were used to jam more and more scope, capabilities, features, user stories, and tasks into existing program and team backlogs as possible to achieve full-utilization and overallocation in order to succeed on this program go-around. Enterprise business requirements had not even been added to the program's integrated master schedule nor backlogs, but, none-the-less this practice was certainly not helping matters much.

Things were getting tense quickly after the first couple of major iterations, although this work largely consisted of architectural runway, SAFe, Scrum, and other traditional ceremonies. People questioned whether SAFe was helping at all without automatic bottoms up schedule analytics. The program's scope was growing daily along with increasing pressure to work harder, longer, and faster. However, it wasn't clear whether there was actually a Product Owner, Product Management Team, and carefully prioritized set of capabilities and features. Apart from the immense amount of preplanning that went into restarting the second phase of the program in the first six months of 2019, construction of an integrated master schedule, and development of multiple high-level and detailed roadmaps, there didn't seem to be any SAFe program backlog grooming going on convert the program's scope into prioritized features. The job of the architecture and elaboration teams was to create manufacture more and more capabilities and features everyday for the upcoming PIs and the job of the primary two feature teams was to complete the current capabilities, features, user stories, and tasks. In other words, it wasn't quite clear who was responsible for the rest of the scheduled tasks.

The program manager tasked government and contractor teams to devise independent integrated master schedules. One team was asked to construct multiple integrated master schedules and run simulations to predict the shortest possible completion date. Although, it was clear to everyone that full realization of this AWS based enterprise casefile system would take years to fully bake, stabilize, and reach operational maturity, the first five SAFe PIs consisted of getting a basic COTS tool to the cloud with 60 legacy systems certified and operational in about 9 months. A technical manager commissioned multiple roadmaps to be manually and automatically generated to track the overall program's health, status, and ability to succeed. Few people were privy to either of these groups of integrated master schedules nor roadmaps. It was like a giant chess game between Greek demigods moving the pieces of human history around the eons to determine the ultimate fate of humanity. Running schedule and roadmap simulations to complete 5,000 tasks in 9 months with few technical resources didn't seem like too much lean thinking was going on. The following observations and recommendations were shared with the government's program and technical leadership teams.

Mid-PI Observations & Recommendations

Strengths	Weaknesses
• Enterprise uses hybrid Traditional, Agile, and SAFe lifecycles	• Traditional waterfall enterprise IT culture and mindset
• Enterprise moving towards Lean and Agile lifecycles/methods	• Hybrid waterfall/agile enterprise process from 1970s
• Enterprise application lifecycle management (ALM) system	• Resistance to SAFe from enterprise methodologists
• Some enterprise experience/success with Agile and SAFe	• Enterprise functional managers want to micromanage
• Some top-down enterprise/program commitment to SAFe	• Enterprise/programs lack SAFe proficiency/experience
• Enterprise/program has a sense of urgency to use SAFe	• Dependence on functional/linear enterprise suppliers/services
• SAFe is a critical success factor (CSF) to program success	• Suppliers/services dependent on traditional documents
• Management participants well-versed in SAFe ceremonies	• Enterprise suppliers/services unfamiliar with Agile ALM
• Top-level chief SAFe Program Consultant (SPC) coach	• Mistake all-day planning meetings for SAFe ceremonies

- Top SPC coach proficient in Lean-Agile terms and practices
- Lead SPC driving enterprise/program towards SAFe mindset
- Lead SPC committed to Lean, Agile, and SAFe ceremonies
- Managers served as SAFe Release Train Engineers (RTEs)
- Scrummasters well-versed in SAFe ceremonies/practices
- Formal, by-the-book top-down SAFe Rollout
- Program participants trained/certified in SAFe practices
- Rare, three-level Large Solution SAFe implementation
- By-the-book Solution and Program Roadmaps developed
- By-the-book SAFe ceremony proficiency and application
- Formal SAFe Program Increment (PI) planning executed
- Five or six SAFe Program Increments (PIs) planned
- SAFe Scrum team cadence synchronized and enforced
- Regular SAFe governance Scrum of Scrum ceremonies
- Program participant SAFe ceremony buy-in and execution
- Program recognizes importance of Lean-Agile Feature Teams
- Formal SAFe application lifecycle management (ALM) tool
- Formal SAFe ALM tool training and self-paced videos
- Top-down SAFe ALM chief engineer and tools coach

- Managers want to serve as surrogate RTEs to micromanage
- No clear distinction between user stories and tasks
- SAFe ceremonies are mistaken for Agile collaboration
- Suppliers/services use SAFe ceremonies for collaboration
- Large-Solution SAFe used for Big Design Up-Front (BDUF)
- SAFe used to create a highly interdependent architecture
- Architectural Runway used for Big Design Up-Front (BDUF)
- Teams developing requirements/designs vs. working software
- More managers and analysts than developers and testers
- Dependence upon SAFe ALM to capture explicit knowledge
- SAFe ALM user story/task burndown and analytics not used
- Program lacks focus on development of working software
- Use SAFe for plan and schedule-driven program execution
- Dependent upon full resource utilization and overallocation
- Program micromanaging enterprise suppliers/services
- Enterprise/program lacking Lean-thinking principles/practices
- Key enterprise/program managers are nationally distributed
- Enterprise/program not committed to DevOps/microservices
- Program participants lack some confidence with SAFe

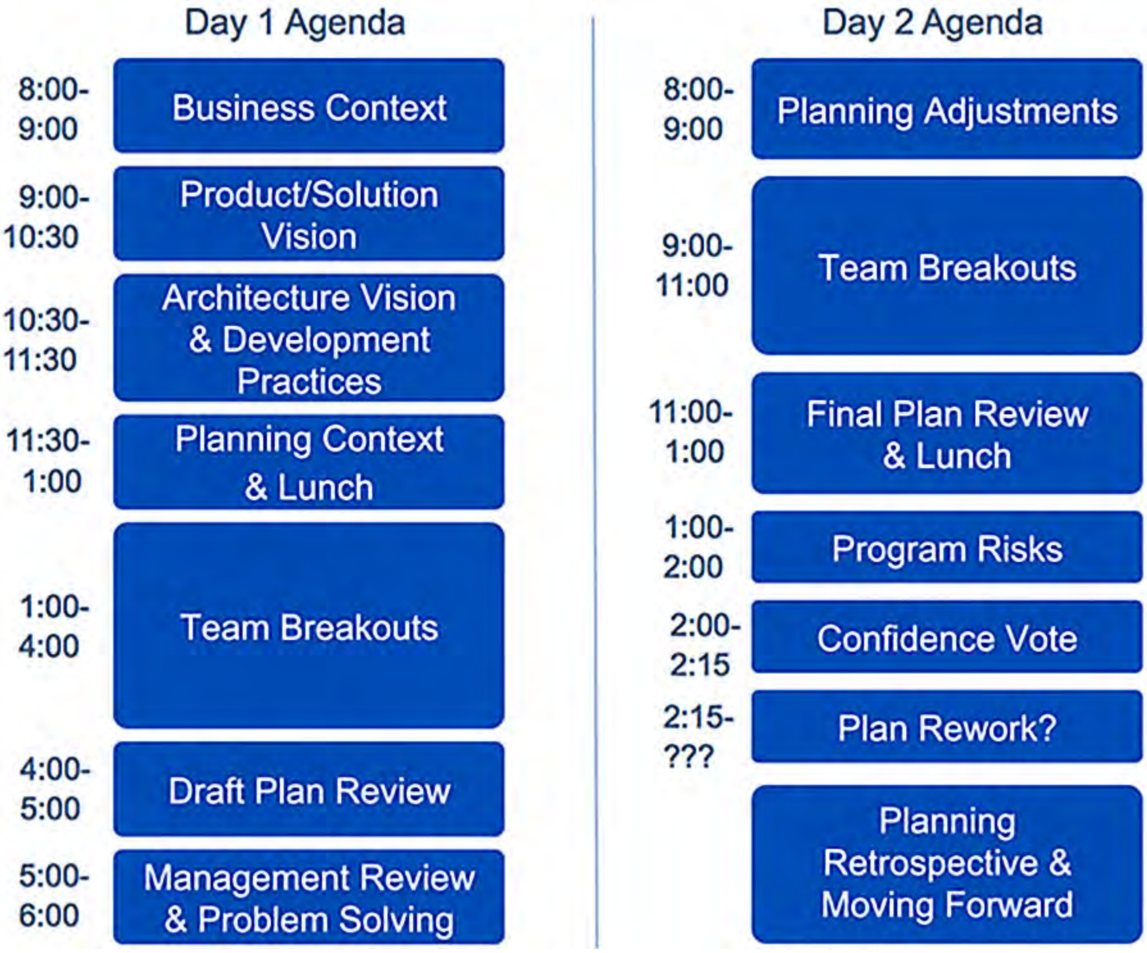
Mid-PI Recommendations

- Keep it small, simple, lean, mean, and streamlined
- Keep the team small, focused, and insulated from enterprise
- Keep the scope of the program focused on a very tiny/small number of capabilities
- Keep the team focused on system/software development (working systems/software)
- Ruthlessly apply lean thinking principles (limited WIP, excess capacity, one-piece workflow, etc.)
- Minimize the plans, documentation, processes, ceremonies, meetings, metrics, and other process elements
- Limit the size and complexity of the enterprise, architecture, design, component, and implementation dependencies
- Limit the number of interfaces to enterprise services, legacy systems, commercial tools and services, etc. where possible
- **Stretch goal**—Develop a small highly-modularized microservices ecosystem (vs. a single, large highly-interdependent design)

PI#2 Planning Event

The program grew in scope, size, complexity, and urgency by leaps and bounds throughout the first PI. As mentioned earlier, two more teams were added to the basic two feature teams from the first PI planning event. These included the architectural runway and elaboration (design) teams. At few more teams were to be added in the near future, closer integration with the public cloud team which was on a different contract, and a contract to add more developers and subject matter experts was planned. A third team was eventually created to synthesize the business process management workflows into even more capabilities and features for architecture and elaboration teams to solution. The scope was growing by leaps and bounds, teams were already operating at full capacity, Little's Law was in full-effect and wait times were getting longer not shorter. Therefore, the second round of PI planning began in earnest, a full two-day event was planned this time, and program participants would participate in a face-to-face big room event. The program manager was to give the overall vision, business owners were to give the enterprise perspective, architects were to talk about technical matters, and SAFe coaches and RTEs would describe PI planning ceremonies.

While the first PI planning event and PI were "by-the-book," the second PI planning event took SAFe to the next level of rigor. The big room associated with another of the agency's SAFe programs was seized, detailed PI pre-planning meetings and slides were prepared, all of the agency's leaders were engaged, and business owners were invited to participate. Multiple PI pre-planning meetings were held every week with every conceivable government leader to get their buy-in, approval, participation, and inputs. The output of this meticulous planning was an elaborately detailed 90-page full-color SAFe PI planning briefing printed for about 100 people. SAFe PI pre-planning is a laborious, stressful, and difficult process, but the delivery team made it seem extremely easy, which it probably wasn't. The delivery team consisted of its own contract, program, project, and agile project managers, including Scrummasters and technical specialists. To an outsider, it seemed as though they worked very smoothly, in-unison, collaboratively, and effortlessly in a highly professional manager. Looks can be deceiving, because SAFe planning is an arduously difficult process. In the end, they followed the basic recommended two-day SAFe PI planning format (as shown below).



Day 1 Agenda	
8:00-9:00	Business Context
9:00-10:30	Product/Solution Vision
10:30-11:30	Architecture Vision & Development Practices
11:30-1:00	Planning Context & Lunch
1:00-4:00	Team Breakouts
4:00-5:00	Draft Plan Review
5:00-6:00	Management Review & Problem Solving

Day 2 Agenda	
8:00-9:00	Planning Adjustments
9:00-11:00	Team Breakouts
11:00-1:00	Final Plan Review & Lunch
1:00-2:00	Program Risks
2:00-2:15	Confidence Vote
2:15-???	Plan Rework?
	Planning Retrospective & Moving Forward

This two-day PI planning event promised to be a great SAFe exemplar, and it didn't disappoint. Of course, it helps when your government customer has just as much SAFe experience as the delivery team and is super smart too. That is, SAFe PI pre-planning is much harder when you have to drag your customer kicking and screaming all the way which happens all of the time. In this case, the shoe was on the other foot and it was the government that was pushing the delivery team to apply by-the-book SAFe ceremonies. The meeting was a bit awkward at first, because all 100 people were in one room face-to-face for the first time. Everyone was extremely experienced—itinerant experts in their fields—and that was a lot of ego and brain power in one place at one time. The room was not intuitively organized, so it was difficult to tell where the front and the back were. In the old days, leaders sat in the front row, while worker bees sat in the back rows, however, we often find leaders sitting in the back of the room now where they can get a greater perspective and its safer. Unfortunately, this was a programming lab with a lot of fixed tables, monitors, little room to maneuver and collaborate, and there were few chairs—Not to mention hard to find and no directions were supplied.

The program manager kicked things off, although she didn't introduce herself by name, supply her title or position, nor clearly delineate the program's vision. The government SAFe RTE gave a brief description of the program's roadmap and upcoming capabilities exposing a glaring hole in the program—No Product Manager, Product Management Team, and prioritized Product Backlog. Perhaps the roadmap is the Product Backlog and the integrated master schedulers are Product Management? The business owners gave an elongated discussion of the high-level business process management workflows, stating that our transformation initiative would only address a small fraction of the enterprise's needs (if at all). The architects gave a terse, but highly contested description of the architecture vision and capability enablers—Perhaps, the architecture team is Product Management and the architectural runway is the Product Backlog? Finally, the government's RTE gave a brief description of the PI planning ceremonies to follow. It was a quiet, but tense PI planning kickoff, that was a lot of brain power and egos in one place, and no one knew exactly what to expect, because it was this program's first live face-to-face two-day PI planning event together as an entire team.

The PI planning venue had three large breakout rooms along with some smaller ones, so the two major feature teams had one of their own and the architecture runway and elaboration teams shared the large executive conference room. The latter two teams are joined at the hip, so it was only fitting they work together as one giant team—all 30 of them. Each of the four teams did their PI planning in advance, so the breakout time was used to confirm the stories they'd devised, elaborate detailed tasks, assign story points, and play Scrum poker to verify estimates. Some emergence did occur as people thought of new user stories and tasks and debated estimates. Scrummasters walked in with long lists of PI objectives in-hand from integrated master schedules and roadmaps, but wordsmithed them a bit with the teams based on their technical expertise. The delivery team project manager walked around a bit to ensure team backlogs were filled to maximum capacity. Government RTEs also observed from a distance. Once again, it was the first-time teams had been all together face-to-face in one location. Program managers, leaders, business owners, and other government managers also met for the first time and talked about overall program and business strategy together.

While breakout rooms enable a little more privacy for individual teams, they take away a little bit of synergy, excitement, and transparency from classical big room SAFe PI planning. However, technical developers prefer more privacy than open big rooms. Furthermore, each breakout room was a little tight, consisting of computer tables with fixed monitors, which wasn't very collaborative either. Each person had their own laptop and was head down staring at the agile ALM system, user stories, and tasks too, which subtracted from collaboration. I fail to mention again that Scrummasters came in with prefabricated PI objectives from integrated master schedules and roadmaps. So, much for bottoms up planning, synergy, brainstorming, and emergence? There was a short lunch break for everyone to go to the government's cafeteria, which also missed an opportunity for collaboration, because the government refused to supply lunch, coffee, or snacks. The teams returned and finalized their draft plans, user stories, tasks, and story points. Business owners even assigned business value on the first day! The draft plans consisting of hand-written PI objectives, capacity, and business value were presented, there was little discussion, and everyone was dismissed.

Again, the first day was an awkward moment for everyone, so let's just call that the forming stage where expectation and level setting were taking place. The government team stayed behind to commiserate about their over-scoped program, coordination frustrations, and how delivery teams were under-committed, under-utilized, and non-productive. Teams went straight to breakouts the next day. The program manager and her staff cornered each Scrummaster and gave them a spirited pep talk on increasing utilization, commitment, and productivity—Scolding them rather severely! Although somewhat disheveled by the severe shakedown, each of our Scrummasters are consummate professionals and returned to the breakout rooms, regathered their composure, and continued adding even more PI objectives, user stories, tasks, and story points to their final PI plans. We went straight to final plan review, risk ROAMing, and confidence votes and dismissed everyone for the day. It was an eye-opening experience for everyone, government, delivery team, SAFe coaches, Scrummasters, technical specialists, etc. In general, everyone was very pleased, and although somewhat tense and mechanical, everyone gained excellent insights into the program's overall status.

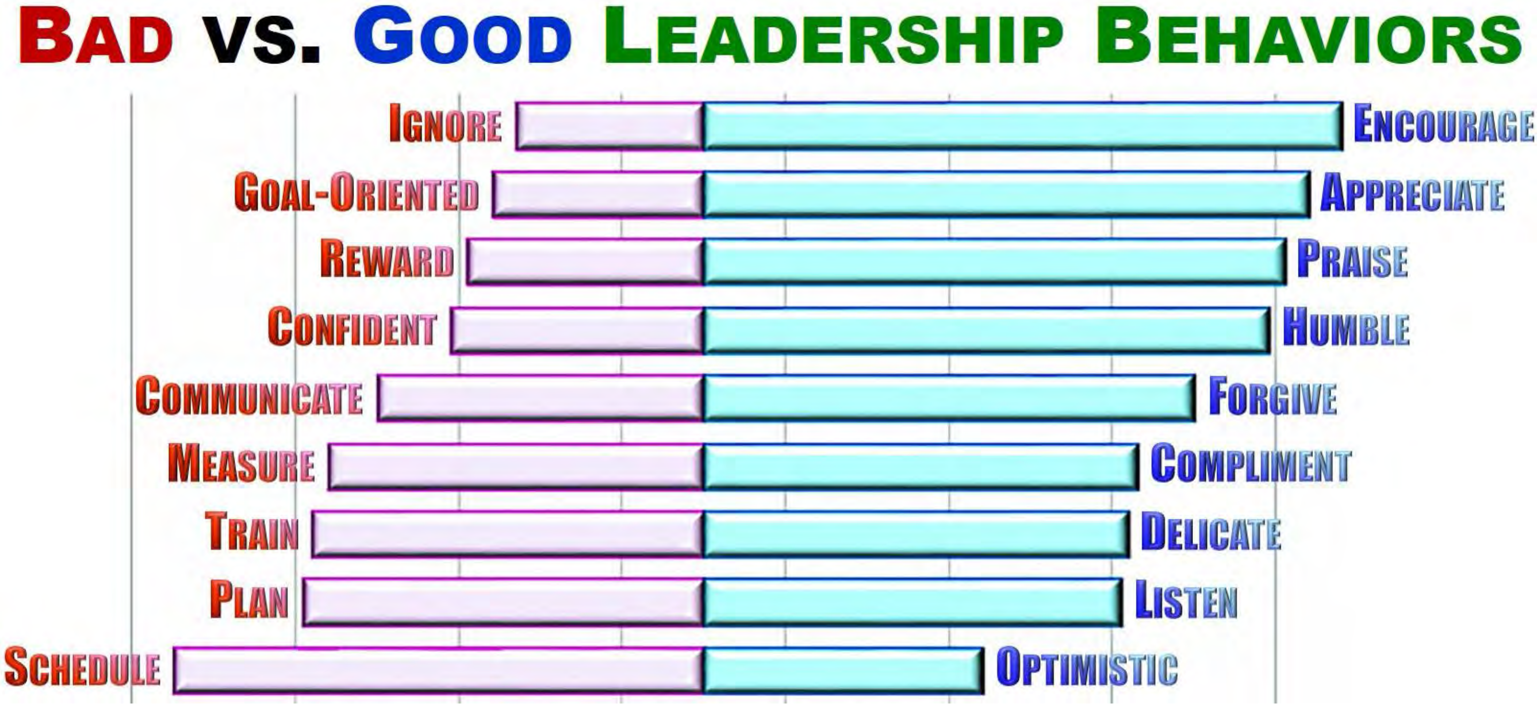
PI Planning Retrospective	
Good	• Everyone is highly talented, skilled, motivated, and leaders. • Very well organized, by-the-book two-day SAFe PI planning event. • Everyone together in one place face to face for the first time. • Good, rare engagement by leadership and business owners. • Rigorous draft planning, user stories, estimates, objectives, etc. • Nice risk ROAMing, confidence votes, and final retrospective. • Program managers and business owners received real-time inputs. • Unprecedented enthusiasm for SAFe ceremonies at government level.
Bad	• Densely packed computer tables, monitors, and poor sound quality. • No food, drink, snacks, or other refreshments provided for teams. • Visual program and iteration boards were underutilized during planning. • No explicit Product Owner, Product Management, and Product Backlog. • Too much pre-planning and not enough collaboration and brainstorming. • Too much heads down agile ALM tool data entry vs. visual F2F planning. • Program scope is basically one giant architectural runway (epic enabler). • Delivery team is a little less enthusiastic about the use of SAFe model.
Better	• Everyone needs training in lean thinking principles and mathematics. • Implement Product Management and dramatically reduce the scope. • Use large conference room without computer tables and breakout rooms. • Use visual planning techniques or state-of-the-art visualization tools. • Allow for more bottoms up emergence vs. top down schedules/roadmaps. • Government needs more training and skills in servant leadership thinking. • Everyone needs even more SAFe training, certifications, and coaching. • Apply more bottoms up self-organizing SAFe principles and practices.

Footnote: *See appendix for full elaboration of PI planning observations*.

Reflection and Analysis

Okay, let’s stop and reflect on what’s happening for a little while. Every government agency and SAFe rollout is different, unique, and has its own set of challenges. Oftentimes, its lack of leadership buy-in and participation, which is not the case here. At other times, its soft commitment to SAFe programs and ceremonies, which is also not the case here. Many times, it’s failure to follow the SAFe implementation roadmap, which is certainly not the case here either. And, yet, many times it’s lack of SAFe training and certification among the core buyer and supplier teams, of which this leadership team has enough. At other times, it’s poor SAFe PI pre-planning, PI planning, and PI execution, of which this program team is top-notch. A critical defect is lack of participation, input, and commitment by business owners, of which this program’s business owners deeply committed. Yet another major issue is lack of visibility, support, and oversight by business executives, of which congressional committees and other oversight agencies make up for in spades. So, what are some of the underlying systemic challenges with SAFe this program is facing. Well, let’s examine some of the key ones, because there still may be others that will emerge later on which is often the case.

- **Lean Thinking**. The largest systemic issue is lack of lean thinking in program inception, planning, and execution. At its root, lean thinking is about working smarter not harder. That is reducing scale, scope, size complexity, risk, uncertainty, cost, length, duration, etc. Technically speaking, lean thinking principles include taking an economic view, managing queues, exploiting variability, reducing batch size, applying work in process (WIP) constraints, controlling flow, getting fast feedback, and decentralizing control. Lean thinking does not include over-scoping, integrated master schedules, micromanagement by committee, filling backlogs to capacity, overutilizing and over-allocating people, endless meetings, and forcing people to work faster. It’s very reasonable to buy a commercial business process management tool, install it in AWS, connect it to legacy systems, and perform federated workflow management. What’s not reasonable is surrounding this simple MVP in a mountain of administrative bureaucracy, constructing endless roadmaps and integrated master schedules, feeding these into Product Backlogs, artificially setting near term milestones, and then forcing people to do the impossible while adding more administrative scope.
- **Servant Leadership**. The second largest system issue is the need for true lean-agile servant leadership vs. top-down command and control. On a social side, servant leaders encourage, appreciate, praise, act humble, forgive, compliment, tread lightly, listen, spread optimism, and most importantly establish a safe, enjoyable, and fun atmosphere. On a technical side, they promote lean thinking principles, provide ample resources and time, mentor and develop people, empower and treat people with respect and dignity, empathize and listen to people, retrospect and learn, admit they don’t know it all, build in extra capacity, limit WIP and batch sizes (scope), reduce vs. increase pressure, allow for emergence and creativity, and reduce complexity, pain, and discomfort rather than increasing it. Servant leadership is all about meeting people where they are, visiting with them, listening, and empathizing with them, identifying their pain points and impediments, and doing everything in their power to remove them rather than creating new ones. Most importantly, servant leadership is about treating people as equals, whether government or contractor, sharing in responsibility rather than shifting blame, and trusting and empowering people to contribute.



- **Emotional Intelligence**. This program needs a healthy dose of training, development, and coaching in emotional intelligence especially among government personnel. You can catch more flies with honey than with vinegar, an old saying that seems to have fallen by the wayside. Respect for people is at the heart of the lean thinking, which is not found in traditional methods that are all about scope, cost, and performance management. Respect (and dignity) for people comes first, and good program performance will follow quickly on its heels. Conversely, disrespect for people leads to poor program performance, not better results. Yelling, screaming, berating, coercing, forcing, and threatening people to work faster slows things down in the long run. You might get short term results from such behavior, but the long-term effects of radioactive behavior are a long, protracted, and damaging cold war. Wanna fix a bad situation fast? Say good morning, how are you, how was your weekend, how is your family, what’s new, and how are you feeling? But, mean it! Take the time to listen, empathize, and build an environment of trust, safety, and fun, not just for the blonde-haired, blue-eyed golden child. Don’t just say, good morning, where’s my report?
- **Cooperation**. Most importantly, government programs need to establish cooperative, collaborative, egalitarian, and power-sharing relationships with their suppliers. That is, high-power-distance between buyers and suppliers is the quickest way to poor IT acquisition performance. Buyers must bring suppliers into the family of trust, share information and power with them, and extend equal decision-making rights. One of the most powerful tools in the lean arsenal is continuous process and product improvement vs. full utilization, overallocation, rigid roadmaps and integrated master schedules, ALMs, metrics, and simply telling people to work faster, harder, and longer. So, first of all, programs must invest in doing good retrospectives, involve everyone in retrospectives including suppliers, and allow suppliers to share in the selection of process and product improvement actions. The greatest asset an IT program manager has are the brains of his or her workforce, not just their backs, fingers, and programming skills. Tap into their brains, empower people to think, and trust them with the authority to take action on their thoughts. Lowering power and status barriers between buyers and suppliers is the quickest ways to IT acquisition success.
- **Expectation Management**. One of the most powerful tools an IT program manager can apply is proper expectation management, especially among high-profile, intensely scrutinized, and multi-pronged transformation initiatives. There is just too much of a tendency to over-scope, over-promise, and inflate people’s expectations by spending large volumes of money to solve everyone’s problems. Expensive, over-scoped public sector IT acquisitions are notorious for poor program performance, cost and schedule overruns, and outright failure. Over-scoping is a violation of basic lean principles, because it leads to longer wait times, lead times, cycle times, cost, complexity, uncertainty, risk, defects, failures, etc. To add insult to injury—overpromising and underdelivering—IT acquisition managers will then start gathering up everyone’s cats, dogs, business requirements, and legacy systems, and throw them into the mix too, inflating everyone’s expectations beyond any hope of reasonable satisfaction. The key to using SAFe is to apply lean thinking, principles, and practices of establishing tight WIP limits, matching capacity to demand, and scoping people’s expectations within the throughput constraints of the delivery team’s capacity.
- **Product Management**. A key and often missing or underperforming function of Lean, Agile, Scrum, and SAFe is sound Product Management. Product Managers and Product Owners are the CEO of IT programs. They’re responsible for interfacing to stakeholders, empathizing, comprehending, and managing their needs, requirements, and expectations, and converting these into a FEW high-priority epics, capabilities, features, and user stories. More importantly, product managers need to apply lean thinking principles more than anyone, understand the team’s capacity and WIP limits, and funnel a few tightly scoped work-items into the value stream—Not to mention buffering teams from overzealous stakeholders. They should cooperate with teams as servant leaders to roll up their sleeves and lend a helping hand to balance workflows—Shortest wait, lead, and cycle times, and highest quality, usability, and customer satisfaction. The wrong thing to do is for a traditional program manager or product owner to wrap a simple MVP in an over-scoped ball of business requirements, convert this into an integrated master schedule, jam its tasks into Product Backlogs, micromanage minutes and seconds, and force everyone to work longer and harder.
- **Visual Management**. SAFe, like the Toyota Production System (TPS), began as an interactive big room, creative, emergent, and visual process of creating bottoms up rolling wave incremental release plans. The result of this event was an Obeya or war room with visual posters representing value streams, work items, processing stations, WIP limits, queues, risks, issues, and other features of successful product and service development (manufacturing). Leaders, program managers, and other key stakeholders could instantly SEE the VISUAL status of the Obeya’s physical plans on the war room wall at any time. It was a like a giant chessboard, the work items were the chess pieces, and moving them around was like playing a global battle between East and West for economic power. Once, again, keys to Lean, Agile, and SAFe thinking are empowerment, equality, collaboration, shared decision-making authority, face to face communications, and VISUALIZATION. Yes, one might argue that meeting a teammate in a breakout room while jamming more user stories and tasks into an agile ALM is a form of collaboration, but it certainly is a poor substitute for an Obeya, SAFe PI Planning, and the use of highly interactive visualizations.

There are many strengths and weaknesses associated with this program’s SAFe implementation. For one, it’s a rather tightly scoped MVP—Buy a commercial business process management tool, install it, migrate it to the cloud, tie in a few legacy systems, and evolve it over a period of years. What can be simpler than that? Secondly, the government chose Large Solution SAFe to leverage the power of lean and agile thinking for successfully completing these mission critical transformation capabilities. Third, it was an extremely well-structured and highly organized SAFe rollout that rigorously followed the SAFe implementation roadmap. Fourth, it ritualistically applied the myriad of SAFe and Scrum ceremonies and practices. Fifth, the team is highly motivated, talented, skilled, intelligent, and very analytical—They're great integrated master schedulers. Sixth, government leadership, program managers, business owners, and other functional specialists are proactively involved in participating in the SAFe ceremonies, including PI planning, which is VERY rare! Seventh, everyone is working very hard to get this MVP signed, sealed, and delivered within 9 months! However, the devil is in the details, solutions seem obvious, and small improvements can go a long way (see below).

Philosophy	Rollout	Operation
☐ Lean Thinking	☑ Leadership Buy-In	☐ Minimal Meetings
☐ Innovation	☑ Management Participation	☑ Apply ALM Tools
☐ Visual	☑ SAFe Experience/Certification	☐ Apply DevOps Tools
☐ Collaborative	☑ SAFe Pre-Planning Team	☐ Expect/Allow Maneuverability
☐ Transparent	☑ SAFe Training	☐ Excess Capacity at All Levels
☐ Product-Oriented	☑ MVP/Product Visioning	☐ Product/Scrumteam Collaboration
☐ Results-Oriented	☐ Near-Term Feature List	☐ Three-Week Iterations
☐ Measurement-Oriented	☐ Vertical MVP Feature Teams	☐ Small Feature Teams
☐ Emotional Intelligence	☑ Scrum Maturity	☐ Collaborative Pairs
☐ Empowerment-Oriented	☑ Cadence/Synchronization	☐ Self-Selected Tasking
☐ Creative	☑ Big Room Venue/Event	☐ Pull vs. Push Work
☐ Experimental	☑ Full Participation	☐ Burndown Working Stories
☐ Fluid	☑ Visual Planning	☐ Don't Micromanage Minutes
☐ Improvisational	☑ Dynamic Collaboration	☐ Create/Burndown Daily Plans
☐ Conversational	☐ Emergent Visual Planning	☐ Demo Software vs. Status
☐ Knowledge Sharing	☐ Safe/Enjoyable Atmosphere	☑ Granularized User Stories
☐ Emergence	☐ Creative/Improvisational	☐ Frequent Deployments
☐ Freedom	☐ Working Software Focus	☐ Sliding vs. Fixed Scope Planning

Footnote: *Blue checkmark is GOOD; Yellow checkmark is ADEQUATE; Pink open box is MISSING*

Summary

So, what did we learn from this brief case study of a large IT-intensive government acquisition program applying SAFe to rapidly deliver a cloud-based enterprise virtual casefile system? SAFe is rapidly eclipsing both traditional, lean, and agile paradigms for IT portfolio, program, and project management, especially in the public sector. SAFe is the embodiment of modern lean thinking values, principles, and practices, which should not be fed by over-scoped roadmaps and integrated master schedules. SAFe is a model to tap into the creative power of frontline IT specialists by inviting and empowering them to create bottoms up rolling wave PI plans through highly collaborative, visual, and interactive visualizations. Servant leaders must create and maintain an environment of trust, safety, and enjoyment in order for creativity to remain high, problem solving to ensue, and success to be achieved. Good SAFe program rollouts using its recommended implementation roadmap are necessary. Culture change comes last not first in SAFe, so people have to be very patient when applying it. Sound Product Management is often a missing element of SAFe success, which needs to relentlessly master and apply lean thinking to balance workflow and achieve success. **What's the bottom line**? Apply lean thinking, lean thinking, and more lean thinking from concept to operation!!!

Overall Program Retrospective (To-Date)	
Good	• Unusually talented and hyper-motivated client and delivery teams. • Customer management is very proactive, involved, and responsible. • Great insight using SAFe to manage large-scale high-risk program. • World-class, extremely well-organized SAFe rollout and structure. • Excellent day-to-day use of SAFe for coordination and management. • Good idea to use enterprise SAFe ALM tool to capture work status.
Bad	• Client structured as functional departments vs. projectized team. • Fully distributed customer and delivery team slows collaboration.

	• Program codified 1970 waterfall in SAFe Program Increments (PIs). • Program is 3 transformations in one to make bricks without straw. • Program is over-scoped with too much WIP and is on short deadline. • Large Solution SAFe used to make heavy BDUF Architectural Runway.
BETTER	• Descope program by an order of magnitude and plan realistic goal. • Make centralized and projectized program management office (PMO). • Identify vertical hi-value feature slices and form feature teams. • Invest in training and coaching for lean concepts and principles. • Delay use of waterfall enterprise life cycle (ELC) and documents. • Localize using small and fewer feature teams as much as possible.

Footnote: *See appendix for full elaboration of SAFe program observations*.

Further Reading

- Campbell-Pretty, E. (2016). *Tribal unity*: *Getting from teams to tribes by creating a one team culture*. Richmond, Victoria: Pretty Agile.
- Campbell-Pretty, E., & Wilson, A. L. (2019). *The art of avoiding a train wreck*: *Practical tips and tricks for launching and operating SAFe agile release trains*. Pretty Agile Pty.
- Dalton, J. (2019). *Great big agile*: *An os for agile leaders*. New York, NY: Apress Media.
- Hopp, W. J. (2008). *Supply chain science*. Long Grove, IL: Waveland Press.
- Kane, L. (2018). *SAFe PI planning*: *A step-by-step guide*. Torrance, CA: Agile One Media.
- Kaye, B. & Jordan-Evans (2014). *Love em or lose em*: *Getting good people to stay*. San Francisco, CA: Berrett-Koehler.
- Knaster, R., & Leffingwell, D. (2018). *SAFe distilled*: *Applying the scaled agile framework for lean enterprises*. Boston, MA: Pearson.
- Leffingwell, D. (2007). *Scaling software agility*: *Best practices for large enterprises*. Boston, MA: Pearson Education.
- Leffingwell, D. (2011). *Agile software requirements*: *Lean requirements practices for teams, programs, and the enterprise*. Boston, MA: Pearson Education.
- Leffingwell, D. (2018). *SAFe reference guide*: *Scaled agile framework for lean enterprises*. Boston, MA: Pearson.
- McConnell, S. (2019). *More effective agile*: *A roadmap for software leaders*. Bellevue, WA: Construx Press.
- Pearson, C., & Porath, C. (2009). *The cost of bad behavior*: *How incivility is damaging your business and what to do about it*. New York, NY: Penguin Group.
- Pound, E. S., Bell, J. H., Spearman, M. L. (2014). *Factory physics*: *How leaders improve performance in a post-lean six sigma world*. New York, NY: McGraw-Hill Education.
- Reinertsen, D. G. (2009). *The principles of product development flow*: *Second generation lean product dev.* New York, NY: Celeritas.
- Rico, D. F. (2018). *Lean & agile contracts*: *21 principles of collaborative contracts and relationships*. Retrieved June 29, 2018, from http://davidfrico.com/collaborative-contract-principles.pdf
- Rico, D. F. (2018). *Lean & agile organizational leadership*: *Establishing vision, guidance, & trust to unleash enterprise competitiveness*. Retrieved February 20, 2018, from http://davidfrico.com/rico18g.pdf
- Rico, D. F. (2018). *Using SAFe 4.5 to transform a $200 million U.S. healthcare portfolio*. Retrieved November 19, 2018 from http://davidfrico.com/safe-case-study-ii.pdf
- Rico, D. F. (2019). *32 attributes of successful continuous integration, continuous delivery, and devops*. Retrieved September 27, 2019, from http://davidfrico.com/devops-principles.pdf
- Rico, D. F. (2019). *Business value of lean leadership*: *22 Attributes of successful business executives, directors, and managers*. Retrieved April 27, 2019, from http://davidfrico.com/rico19a.pdf
- Rico, D. F. (2019). *Business value of lean thinking*: *Capitalizing upon lean thinking principles to rapidly create innovative products and services*. Retrieved November 11, 2019, from http://davidfrico.com/rico19d.pdf
- Rico, D. F. (2019). *Evolutionary architecture*: *24 principles of emergent, organic, and highly-adaptive design*. Retrieved September 3, 2019, from http://davidfrico.com/evolutionary-architecture-principles.pdf
- Rico, D. F. (2019). *Lean & agile enterprise frameworks*: *Using SAFe 4.6 to manage U.S. government agencies, portfolios, and acquisitions*. Retrieved June 12, 2019, from http://davidfrico.com/rico19b.pdf
- Rico, D. F. (2019). *Lean leadership*: *22 attributes of successful business executives, directors, and managers*. Retrieved August 28, 2019, from http://davidfrico.com/lean-leadership-principles.pdf
- Rico, D. F. (2019). *Piloting the scaled agile framework (SAFe) in a top 10 u.s. energy firm*. Retrieved May 15, 2019, from http://davidfrico.com/x-safe-case-study-iii.pdf
- Rico, D. F. (2019). *Top SAFe videos*, Retrieved April 7, 2019, from http://davidfrico.com/top-safe-videos.htm
- Rico, D. F. (2020). *32 principles and practices of highly-successful SAFe implementations*. Retrieved February 16, 2020, from http://davidfrico.com/safe-principles.pdf
- Rico, D. F. (2020). *Business value of CI, CD, & DevOps*(*Sec*): *Scaling up to billion user global systems of systems using containerized microservices*. Retrieved January 26, 2020, from http://davidfrico.com/rico20a.pdf or http://youtu.be/qrWRoXSS9bs
- Rico, D. F. (2020). *Business value of lean thinking*: *Capitalizing upon lean thinking principles to rapidly create innovative products and services*. Retrieved January 29, 2020, from http://davidfrico.com/rico20b.pdf or http://youtu.be/wkMfaPAxO6E
- Rico, D. F., Sayani, H. H., & Sone, S. (2009). *The business value of agile software methods*: *Maximizing ROI with right-sized processes and documentation*. Ft. Lauderdale, FL: J. Ross Publishing.
- Rothman, J. (2016). *Agile and lean program management*: *Scaling collaboration across the organization*. Victoria, Canada: Prac. Ink.
- Smith, P. G. (2018). *Flexible product development*: *Agile hardware development to liberate innovation*. San Fran., CA: Jossey-Bass.
- Soukath-Ali, M. M. (2017). *Get SAFe now*: *A lightning introduction to the most popular scaling framework on agile*. Chennai, India: Ali.
- Thompson, K. W. (2019). *Solutions for agile governance in the enterprise* (*SAGE*): *Agile project, program, and portfolio for development of hardware and software products*. Vancouver, CA: Sophont Press.
- Yakyma, A. (2016). *The rollout*: *A novel about leadership and building a lean-agile enterprise with SAFe*. Boulder, CO: Yakyma Press.

Appendix I—Overall Post SAFe Program Increment (PI) #2 Planning Retrospective, Observations, & Recommendations

- **Planning**. *Extremely well-organized and orchestrated SAFe pre-planning and planning.*
- **Facilities**. *Very nice facilities extremely well-suited for SAFe PI planning events.*
- **Synchronize**. *PI planning used to periodically collocate nationally distributed team.*
- **Attendance**. *Leaders, middle managers, team leads, and developers attend PI planning.*
- **Culture**. *Highly motivated, confident, extroverted, left-brained, analytical planners.*
- **Agile**. *Use Lean-Agile-SAFe frameworks to get more for less and better program outcomes.*
- **SAFe**. *Like SAFe because it most closely resembles plan-driven culture with agile outcomes.*
- **LSS**. *Like Large Solution SAFe, because of its historical big up-front architecture culture.*
- **Backlogs**. *Leaders want backlogs driven to full-capacity and utilization as overallocated IMS.*
- **Utilization**. *Mathematically speaking, full utilization slows, stops, and retards productivity.*
- **AgileFall**. *Traditional waterfall phases, documents, and process WIP are hard coded into SAFe PIs.*
- **Phases**. *Architecture and design are phased instead of applying evolutionary architecture principles.*
- **Topdown**. *Using SAFe to continue promulgating its strong, historical top-down driven planning culture.*
- **Scope**. *Program is simply over-scoped from hidden process, document, and status meetings point-of-view.*
- **Schedules**. *Strong culture of using integrated master schedule (IMS) for modeling over-scoped programs.*
- **IMS**. *Management wants to use Agile Lifecycle Management tool to forecast and track program performance.*
- **Disconnect**. *There's a big gap between heavy traditional IMS vs. lighter Agile ALM from Scrum/SAFe teams.*
- **Product Planning**. *Managers should collaborate in product planning vs. using PI planning for this purpose.*
- **Objectives**. *Scrummasters give PI objectives to SAFe teams during PI planning for socialization and grooming.*
- **Experience**. *Customer and delivery teams have little experience with bottoms-up Lean-Agile-SAFe intent/thinking.*
- **Scrummasters**. *Talented and motivated with high face-validity and balance of left and right-brained thinking skills.*
- **Rewards**. *Scrummasters who drive teams to full-capacity and utilization are rewarded and perceived as more successful.*
- **Metrics**. *Some Scrummasters gaming SAFe metrics by overloading teams with PI objectives while underdelivering a little bit.*
- **Board**. *Teams should have draft program boards by end of day one and use them for draft plan reviews and management reviews.*
- **Paradox**. *Western culture is a hybrid of traditional push-driven planning vs. Eastern pull-driven democratic decision making.*
- **Social Skills**. *Technical developers are highly talented, competent, aware, involved, and collaborative in SAFe planning ceremonies.*
- **Demographics**. *More demographic diversity at SAFe development team level than at Scrummaster, middle management, and leadership.*
- **Mechanical**. *Robotic structure, pace, and organization, minimizing social interaction, collaboration, emergence, trust, and team building.*
- **Collaboration**. *Greater investment in teamwork, collaboration, team building, communication, social skills, and emotional intel needed.*
- **Enterprise**. *SAFe is not a standard enterprise Lean-Agile Framework, there are multiple Lean-Agile constituencies, and other promoted.*
- **Responsibility**. *Leaders are convinced the key to over-scoping, over-planning, and ambitious transformation is speed and full utilization.*
- **Delivery team**. *Scrummasters highly motivated and maintain a positive attitude in the face of high-uncertainty, pressure, and frustration.*
- **Misuse**. *Government using SAFe PI planning for top-down chastisement of delivery team contractors, negating bottoms up collaboration.*
- **Product Management**. *Government needs a healthy Product Management function to prioritize work, balance workflow, and buffer teams.*
- **SPC**. *Government SPC needs to identify roadblocks, bottlenecks, and impediments, identify solutions, and buffer teams from distractions.*
- **Servant Leaders**. *De facto SPCs are multi-hatted, suboptimizing, and spread too thin to identify and resolve program wide impediments.*
- **Metrics**. *It easy for people to suboptimize, game metrics, and protect team than solve program issues, challenges, risks, and impediments.*
- **Involvement**. *The good news is leaders and business owners are actively involved, which is rare in both traditional and lean-agile projects.*
- **Training**. *Leadership needs lean thinking, motivation, emotional intelligence, etc. (NOT processes, metrics, OKRs, IMS, EVM, ALM, etc.).*
- **Retrospectives**. *Leaders need coaching in process improvement, retrospectives, simple/visual lean thinking tools, and positive behaviors.*
- **Visualization**. *Scrummasters need training in interactive planning vs. heads down ALM data entry robbing SAFe of its power (iObeya).*
- **Top Heavy**. *Program is filled with too many leaders, managers, supervisors, technical leads, and Scrummasters (needs more developers).*
- **Leadership**. *Servant leaders needs to be identified, selected, acquired, trained, rewarded, managed, evaluated, developed, and optimized.*
- **Cage Match**. *SAFe PI planning is like a mixed-martial-arts (MMA) cage match or gladiator arena (when used by highly aggressive teams).*
- **Biology**. *Studies of cognitive biology, animal behavior, and insect societies are very helpful in understanding the dynamics of PI planning.*
- **Survival**. *The key to survival, like in evolutionary biology is teamwork, collaboration, cooperation, and heterogeneous symbiotic groups.*
- **Behavior**. *Understanding and mastering subliminal human behaviors is a key to identifying, mitigating, and managing natural conflicts.*
- **Distributed**. *Entire team is distributed hindering collaboration and synergy, so PI planning enables socialization and problem solving.*
- **Visibility**. *This is a highly visible national-level program under close congressional scrutiny, so PI planning data was a priceless asset.*
- **Levity**. *Facilitators need training and skills in eliciting levity, fun, trust, and openness, which are keys to highly effective PI planning.*
- **Cliques**. *There is strong evidence of natural self-organizing cliques, silos, and rivalries along demographic lines which is very common.*
- **F2F**. *Nationally distributed teams were elated to participate, meet people, collaborate, and problem solve in first formal PI planning event.*
- **Surprise**. *Leaders were surprised by the favorable feedback because they don't comprehend emergent bottoms-up PI planning principles.*
- **Synergy**. *Traditionalists don't comprehend the power and value of empowerment, socialization, collaboration, synergy, and emergence.*
- **Business**. *There were a lot of business owners present, giving leaders and business owners a unique opportunity for collaboration as well.*
- **Runway**. *The program is building the architectural runway as the plane is departing, it's just getting started, and expectations are high.*
- **Lean**. *It's true that "culture change comes last not first" in SAFe, however much more upfront lean thinking would have helped a lot.*
- **Paradigm**. *Traditional methods are FIXED scope, while lean-agile methods SLIDING scope (trick is to slide its scope to realistic features).*
- **Plans**. *Developers devise bottoms up plans to complete pre-prioritized features in PI planning (but product management layer is missing)!*
- **Others**. *Non-developers may also use SAFe frameworks, ceremonies, practices, and tools alongside developers (not just programmers).*
- **Coaching**. *Program should invest in lean-agile coaching to reach tipping point, disarm impediments, optimize performance, and succeed.*
- **Cooperation**. *There has to be far more cooperation, trust, teamwork, collaboration, and joint problem solving between buyer and supplier.*
- **Bottomline**. *It's a highly structured SAFe program, it's needed to manage risk, and everyone is highly motivated/talented which is great.*

Appendix II—Overall Post SAFe Program Increment (PI) #3 Planning Retrospective, Observations, & Recommendations

The prior SAFe Program Increment (PI) #2 got off to a rather fast start, in spite of the rather tense, mechanical, and somewhat ambiguous context for the PI#2 planning event described in the main body of this case study. The cadence consisted of 5 two-week iterations, although it was rather ambiguous whether the final two-week iteration was a bona fide Innovation and Planning (IP) iteration. Well, in fact, all five iterations were devoted to technical vs. innovation-oriented work, and the first two-days of PI#3 were used for PI#3 planning. Given that this program is rather overscoped and on a very tight schedule, conducting a bona fide IP iteration would have been a rather hard sell. Especially, as you recall, when the program manager sternly told (yelled at) the Scrummasters to work faster (make more bricks without straw). The good news is that the program religiously followed SAFe ceremonies throughout PI#2, many of which were observed first-hand (or, virtually at least as we'll discuss in more detail later). Individual SAFe teams followed routine Scrum ceremonies, such as iteration planning, daily standups, reviews, and retrospectives. Collective teams followed routine SAFe ceremonies such as a Scrum of Scrums (SoS) or Agile Release Train (ART) Synch meetings twice a week, system demos and retrospectives, inspect and adapt sessions, and a traditional program review every other week. In some sense, the program, SAFe teams, and Scrum teams sort of stabilized a bit, and there was considerably less consternation and pressure as was experienced during the pre-PI#1, PI#1 planning, and PI#1 timeframes. In other words, the SAFe program started to hum a bit during PI#2 execution.

However, systemic issues continued to dog this SAFe program and new insights and unplanned problems continued to emerge throughout PI#2. The first major problem was that the core business process workflow contract continued to stall and present a major risk to completing this program at all. Although, it's yet to be started, the excitement concerning that risk died down by the mid-PI#2 timeframe. The second major problem was the commercial cloud services contract was completely separate, so synchronizing its cadence with the main business process workflow contract will never be achievable, which is a darn shame. The third major problem was that the customer performed an enterprise-wide agile application lifecycle management (ALM) tool upgrade to improve SAFe support, metrics, dashboards, digital program boards, and other major management functions. The ALM upgrade didn't show any signs of stabilizing until the end of PI#2, although the customer pressured everyone to use the ALM system before PI#1 began (so the ALM upgrade was two PIs behind schedule, but everyone was still expected to continue making bricks without straw). The fourth major problem was that the government customer decided to award the primary business process services contract to an entirely different firm midway through PI#2 and terminate all interrelated contracts by the end of PI#2. Many times, a government program management team goes native and falls in-love with its prime contractor, only to have its contracting department abruptly select a different (lower) bidder, which seems to have happened in this case. However, I'm not sure the government wouldn't have desired a new services contract.

In spite of these major structural issues, other insidious and hard to see challenges also dogged this SAFe program. First of all, there were many government and contractor teams involved in this program, but only two were applying SAFe and Scrum ceremonies and the ALM tool throughout PI#1 and PI#2. However, the government wanted to use the ALM tool as a single source of measurement "truth" about the status of the program—surrogate Integrated Master Schedule (IMS)—although the two SAFe teams comprised less than 20% of the program's scope. Second of all, only ONE of the TWO SAFe teams was doing development, while the other was doing system architecture work, but not necessarily applying SAFe Large Solution ceremonies to identify a small set of capabilities or features with high business value using SAFe's Weighted Shortest Job First (WSJF) or other lean thinking principles and practices. So, there was only ONE SAFe team doing actual software development, and in spite of its brisk velocity, was simply a bottleneck and could not possibly implement all desired capabilities and features within the allotted time. Actually, both teams were bottlenecks, the government wanted to implement a million years of features (and a few more), the system architecture team wasn't prioritizing and defining SAFe capabilities and features fast because the government wanted them all, and, of course, the capacity of the sole development team wasn't great enough to do everything. In spite of these bottlenecks, the government resolved to create a separate tiger team to identify even more capabilities and features, but it was not required to follow SAFe ceremonies and plan and track its work in the ALM tool.

But, wait, there's more! Clearly the program needed some sort of lean and agile architecting function. It needed to map out its enterprise or system architecture in advance, perform impact mapping, and identify a feasible minimum viable product (MVP), which was not complete by the end of PI#2. Probably the most insidiously hidden problem was the large power-distance, lack of trust, and lack of collaboration between the government and contractors. There were many government decision-making meetings behind the scenes to which contractors were simply not privy.

Complete transparency, collaboration, egalitarianism, empowerment, and trust are absolutely necessary between buyers and suppliers in all situations, be it traditional or agile (and SAFe cannot function without it). Therefore, apart from the routine SAFe and Scrum ceremonies conducted by the two teams, there was very little communication, collaboration, and "meetings" between the government and contractors. There was some turnover, changeover, and inconsistency in key government leadership positions during PI#2, which also caused some major perturbations. Contractor-to-contractor relationships also suffered from lack of collaboration, communication, cooperation, and trust, even within individual contractor teams. This is pretty normal in government contracting, but it took two PIs for this insidious problem to create some real problems for management and technical productivity. Contractors often begin by playing nicely, quickly fall into silos, and then later collaborate more at program's end. No contractors were exempt from this problem, and there is high levels of distrust and competition within individual contracts and firms.

Probably the most insidious problem of all hindering communication, collaboration, cooperation, productivity, and trust is that this program is nearly 100% nationally distributed. Although SAFe PI planning helps mitigate this risk by identifying epics, capabilities, features, stories, and tasks, computing team capacity and velocity, mandating gobs of ceremonies, and suggesting tools to track metrics, rigorous SAFe PI planning is simply insufficient for distributed cross-functional teams. Even individual SAFe teams are nationally distributed with a handful of people in each major city. This has some benefits as two or three people can collaborate locally face-to-face without interruption, but decisions get made locally and suboptimally, non-collocated team members are left out in the cold, and God help you if you don't work for the same firm (or your task spans government-contractor boundaries). Even government teams are distributed, locally suboptimize, don't apply SAFe and Scrum ceremonies nor use the ALM tool, and refuse to cooperate, communicate, or openly collaborate. Part of this is simply the new information age, where people often work part-time on multiple programs (multitask), live in a different city or part of the world, and often don't have time for more than an occasional email or single emoji by texting, Skype, Slack, etc. The 20th century was dominated by collocated face-to-face teams and communication, while the 21st century is NOT (and SAFe experts now demand collocated SAFe teams). Unfortunately, collocated teams, while clearly superior, do not solve all problems, especially due to the demographic diversity common in today's world presenting almost insurmountable collaboration risks.

Okay, so let's briefly talk about SAFe PI#3 planning! Due to the nature of the contract recompete, communications between government and contractors and contractor-to-contractor came to a screeching halt. PI#3 pre-planning was done by a small collocated team in a hurry and in the dark. There was actually some government concern whether a PI#3 would even happen at all (e.g., would the existing contract continue, or could the system architecture team work fast enough to feed features into the sole development team in-time). Of course, the government took no responsibility for the performance of its own tiger team formed at the beginning of PI#2 to serve as a liaison between the business analysts and the system architecture team (which was a bottleneck). Tensions were high, the PI#3 deck came together late, was not printed in advance, nor sent to the participants until PI#3 planning started. This was actually pretty good, as the PI#2 deck was only printed by not electronically distributed, so the PI#3 planning deck was emailed during PI#3 planning, because there was no time to print it. The program manager attended, but not the government lead RTE. This wasn't actually a bad thing, because the deputy-RTE stepped in and did a good job facilitating PI#3 planning. The government gave awards to the government staff, which is a typical sleight-of-hand to punish the contractors. They wouldn't even say the name of their prime contractor! This is a no-no—What was the government thinking? The program manager was more subdued and less aggressive than in PI#2 planning, which was a good thing. It's hard to say whether this was intentional or accidental—Probably due to the contract recompete!

There were no nametags, no one identified themselves when they spoke, there was no microphone, and the video screens were too small. Although, the program deserves credit for using a large central room with breakout rooms for teams to conduct PI planning, the room is LESS than ideal. First of all, the ceilings are low, preventing use of a large overhead projector so everyone can see the presentations, program boards, etc. Second of all, there are desks in the entire space with fixed computer monitors taking up space, cluttering the entire space, hindering collaboration, visibility, sound, etc. No one uses fixed computer monitors anymore, and everyone uses laptops, especially in civilian agencies. The desks and computer monitors should be donated to a museum. The most insidious problem is the government's insistence that no visualizations, program boards, post it notes, etc. be used for PI planning and that all planning should take place in the ALM tool. VISUALIZATION is a KEY to the Toyota Production System (TPS)—How or why does the government feel they can SUBTRACT VISUALIZATION from Lean thinking? PI planning takes place in advance, PI objectives are established, features and user stories identified, and everything is placed in the ALM tool the week before PI planning. The development team was done with PI planning after the first day and *conducted their Iteration planning as well. No visual program board was constructed illustrating the features of all teams and their dependencies. I understand the government's necessity to do automated SAFe tracking for distributed teams, but they seriously need to consider augmenting their ALM tool with a visualization technology like iObeya!*

The best part of PI#3 planning was there was ample food to eat as the government provided pizza for everyone on the first day and hosted a large potluck luncheon on the second day for everyone. The government's commissioner gave a stirring speech and the government's SAFe coach made everyone watch a motivational TED talk on the importance of teamwork very early in PI#3 planning (http://youtu.be/PY_kd46RfVE*), which emphasized the importance of valuing everyone's contributions—I'm sure government leaders came in loaded for bear, so the TED talk disarmed everyone immediately! Although the government's program teams did NOT participate in PI#2 planning and SAFe ceremonies, they did participate in PI#3 planning and presented their PI objectives, Program Boards, Features, Dependencies, Risks, etc. We'll have to wait and see whether they conduct routine SAFe and Scrum ceremonies and participate in the SoS or ART-Synch. You have to give everyone a great deal of credit for applying SAFe, sticking to it, and not giving up so easily (although I'm sure that's crossed everyone's mind)! There are far too many programs in the world that are not applying SAFe to this degree! That being said, the government is applying SAFe as just another program management technique to get their business requirements done better, faster, and cheaper (every darn one of them)! There's NO lean thinking in that and this is a big mistake! Second of all, most contractors, even the good ones, don't seem to know why they're applying SAFe—That is, they're applying SAFe because the government told them to use it! A little lean thinking goes a long way, and EVERYONE can find satisfaction in that from business analysts to testers!*

PI#2 & PI#3 Planning Observations

Strengths	Weaknesses
• *Smart highly motivated government-contractor teams.*	• *Using SAFe for traditional program management.*
• *Planning oriented government-contractor-teams.*	• *Everyone is not trained in lean thinking principles.*
• *Selected SAFe for use in this government program.*	• *High government-contractor power distance.*
• *Government team trained and certified in SAFe.*	• *High inter and intra contractor distrust-competition.*
• *Government driving use of SAFe vs. contractor.*	• *Nationally distributed government-contractor team.*
• *Applying well-structured SAFe PI pre-planning.*	• *Not applying SAFe lean portfolio management.*
• *Applying well-structured SAFe PI planning.*	• *Not applying SAFe system architecture practices.*
• *Applying routine SAFe and Scrum ceremonies.*	• *PI planning room not conducive to collaboration.*
• *Agile Release Train (ART) is synchronized.*	• *Government removed visualization from PI planning.*
• *Standard ALM tool with strong SAFe support.*	• *Small number of program teams using SAFe-Scrum.*
• *Use central "Big Room" for SAFe PI planning.*	• *Lack of clear objectives, alignment, and language.*
• *Leadership participation in all SAFe ceremonies.*	• *All contracts not centrally managed nor use SAFe.*

Recommendations

- *Continue applying SAFe principles, practices, and ceremonies (don't give up, they're getting better).*
- *Invest in generic lean thinking principles, practices, values, techniques, tools, and technologies.*
- *Also invest in Kanban-oriented training, principles, practices, values, techniques, tools, and technologies.*
- *Get trained and certified in SAFe lean portfolio management and especially SAFe architecture practices.*
- *Dramatically downsize the program's scope, select a few high-value epic-MVPs, and constrain expectations.*
- *Slow down and focus on a few high-value epic-MVPs, so the government can get more for less sooner than later.*
- *Success lies in excess capacity, lean narrowly scoped epic-MVPs, and DevOps vs. full-utilization and capacity.*
- *Don't abandon SAFe's lean visualizations to track nationally distributed teams in enterprise ALM tool.*
- *Use a better SAFe Program Increment (PI) planning venue (or substantially remodel the labs you're using).*
- *Create and emphasize a one-team culture (government-to-contractor, contractor-to-contractor, intrafirm, etc.).*
- *Create central PMO, align all program contracts under PMO, apply SAFe throughout, align, synchronized, etc.*
- *All program government and contractor teams must participate in PI planning, ceremonies, and ALM tool use.*
- *Take a hard look at SAFe 5.0 and its business agility principles, practices, layers, functions, and ceremonies.*

Rico, D. F. (2020). Business value of lean thinking: Capitalizing upon lean thinking principles to rapidly create innovative products and services. Retrieved January 29, 2020, from http://davidfrico.com/rico20b.pdf

Appendix III—Overall Post SAFe Program Increment (PI) #3 Execution Retrospective, Observations, & Recommendations

- **Multiple Contracts Blocked from Start**. *Multiple strategic and tactical acquisition contracts were blocked from the get-go! Although this acquisition was composed of multiple closely interrelated Epic-MVPs, the core functionality was a commercial-off-the-shelf (COTS) business process workflow automation tool. This agency had a long track-record of vendor lock (e.g., buy a tool once and use it forever to the tune of billions of dollars in payouts to COTS vendors over many decades). Core IT vendors understood this concept, and, although this acquisition was merely an experimental MVP of MVPs to lay a foundational infrastructure for later building, all vendors wanted in on the ground floor (otherwise they'd get locked-out of core business functions FOREVER!). This led the losing COTS workflow vendor to immediately protest, rendering this SAFe acquisition dead-in-the-water like a lame duck! It's unclear whether the agency really needed such an expensive and exclusive COTS workflow component to begin with, so selecting one from the start, vs. building a true-MVP may have been the lean faux paus created. If this was a true SAFe-driven MVP, then a separate contract to acquire the COTS tool creating an impenetrable RISK blockage or impediment may have been avoided.* ***Lesson learned****—Don't comprise a SAFe acquisition from multiple disparate cost-intensive acquisition contracts that will lead to obvious blockages from the start, especially if you're a highly-visible agency with a long history of COTS IT vendor-lockin that pays out billions in operations and maintenance fees over decades (cash cow)!*
- **Government Operating in a Vacuum**. *This agency's leadership and operational acquisition teams operated in a vacuum from cradle to grave, making key decisions for suppliers playing strategic and tactical roles. Although this agency was on-the-move to lean and agile thinking and practices and selected SAFe as its process model, it left out transparency, cooperation, and bottoms-up thinking altogether. A critically insurmountable success factor to large and complex public sector acquisitions, other than keeping them simple from the start, is historically high degrees of low-power distance hierarchies, flat networked decision-making governance structures, and unprecedented levels of egalitarianism, democratic decision-making, and good ole' bottoms up participation. This agency operated key governance decision making boards in the dark from sunup to sundown five and sometimes seven days a week, devised strategies and tactical plans without contractor input, and then flowed these traditionally-minded precepts down to unsuspecting contractors and suppliers instead of eliciting their input on strategic and tactical decisions as recommended by lean, agile, and SAFe principles and practices. Worse yet, when the wheels started coming off the wagon, this agency started directing its frustration towards the suppliers for its own ill-conceived decisions, sort of expecting contractors to offer unsolicited help for sunk decision-making costs (e.g., we wrecked into a tree, what do we do now?). This agency had a track record of attracting brilliant left-brained analytical thinkers, but chose infinitely complex solutions and implementation frameworks, which were completely antithetical to lean and agile thinking principles.* ***Lesson-learned****—Other than keep-it-simple, institute vicious transparency, involve everyone in bottoms-up strategic and tactical decision-making, and realize there's more wisdom in the crowd than in a few left-brained leaders with high IQs who will wreck the ship out of cognitive blindness and overconfidence in principles and practices associated with traditional thinking for composing infinitely complex acquisitions and resulting IT solutions.*
- **Not Everyone Using SAFe**. *This government agency wisely chose the SAFe model for exercising lean-agile thinking principles and practices to help rescue, restart, and successfully complete this four-year-old IT acquisition that had perceived immense business value to the enterprise. However, the agency's personnel itself were responsible for 80% to 90% of the work-in-process, some of it value and non-value adding, while relegating SAFe ceremonies, use of ALM tools, and value-adding IT product and service development to a few small contractor teams or suppliers. That is, the Pareto Principle was working well against this government acquisition from the get-go (i.e., the government used traditional ceremonies for 80% of the acquisition products and services, while key contractor suppliers used SAFe ceremonies for 20% of the value-adding work). To make matters worse, this government agency wanted the suppliers to plan, track, and help manage ALL of the acquisition activities, including traditional government and SAFe contractor work, using traditional principles and practices. So, some planning, tracking, and execution reports were represented using traditional instruments, while others were displayed using SAFe instruments (that were NOT easily reconcilable). This caused a great deal of early confusion—traditional parties didn't know what SAFe parties were doing and SAFe parties did not know what traditional parties were doing. Worse yet, this government agency tormented the few contractor SAFe teams for failing to report out 100% of the scope using SAFe instruments, traditional parties used this dual operating system as proof SAFe was failing, and everyone was dissatisfied from beginning to end.* ***Lesson-learned****—Everyone must use SAFe principles, practices, ceremonies, tools, reports, and instruments from cradle-to-grave to be successful with SAFe.*
- **Autonomous Traditional Contracts**. *This government acquisition was not a single integrated acquisition with all contracts subservient to a master contract with task orders, but was composed of multiple AUTONOMOUS contracts for IT program management office (PMO) services, IT integration services such as SAFe product and service development, Cloud Computing infrastructure design and development services, traditional integrated master scheduling (IMS) services, independent verification and validation (IV&V) services, and many more. Unfortunately, a key Epic-MVP contract was used for a Cloud Computing infrastructure (i.e., the COTS business process workflow automation suite was supposed to operate on a commercial cloud solution). However, a separate government department operated the cloud computing contract in the dark—presumably using linear waterfall principles and practices with long lead times, little to no integration, and very little transparency and insight into its plan, operation, and status. Therefore, the Cloud Computing team was not integrated into the main SAFe Agile Release Train (ART) nor main SAFe Solution Train, there was little synchronization, and it was nearly impossible to coordinate the development of software services to run on the cloud platform.* ***Lesson-learned****—Include key Epic-MVP enablers such as Cloud Computing IT infrastructure into your main contract, have them use SAFe, and certainly integrate them into the main ART or at least main SAFe Solution Train for proper alignment, synchronization, planning, tracking, coordination, and insight.*
- **Little-to-No Collaboration and Cooperation at Any Level**. *This government acquisition suffered from little-to-no collaboration and cooperation between the government and key suppliers, supplier-to-supplier, and within individual supplier contracts and teams. That is, everyone operated in a vacuum from the get-go, government teams competed with contractor teams for power and status, contractors competed with one another to look good and make the others look bad, contractor teams competed for power and status with their own subcontractors, and there was even high-degrees of competition between SAFe teams within individual suppliers. This is pretty typical in large government acquisitions leading to wasted resources in the billions of dollars. Often times, government and contractors begin cooperating at unprecedented levels once an acquisition runs into trouble, is in peril of cancellation, or simply runs out of money. Then everyone starts cooperating like they were old friends. Is this simply part of Bruce Tuckman's model of Forming, Storming, Norming, and Performing, or something worse like selfish suboptimization associated with economically-driven traditional thinking principles and practices leading to historically high IT failure rates across the globe in the trillions of dollars annually? None-the-less, lean-agile thinking*

principles, practices, and frameworks such as SAFe are designed to maximize collaboration and cooperation, minimize sub-optimization, and maximize successful IT acquisition outcomes! So, what's the problem? Well, one probably isn't applying SAFe principles and practices very well if no one is cooperating at any level! ***Lesson-learned**—Be diligent about applying SAFe principles and practices from the get-go, especially if you wanna be successful with SAFe, have a successful public sector IT acquisition, and experience unprecedented levels of cooperation and collaboration minimizing the degenerative effects of historically-high IT failure rates associated with traditional thinking*!

- **Blame-Game Operating Culture**. *This government agency suffered from a blame-game operating culture from cradle to grave. That is, this agency planned, managed, and executed billions if not trillions of dollars in complex traditional IT acquisitions over decades that that rarely succeeded. Therefore, it evolved a Gene or DNA structure of self-defense to blame everyone else for infinite complexity that could never be unraveled by any means, be it lean, agile, traditional, or SAFe thinking. Government departments blamed one another, contractors blamed one another, and individuals blamed one another from cradle to grave once the SAFe restart of this four-year-old IT acquisition began. Was blaming someone from day-one a motivational tool to enhance success or was it simply a method of acknowledging you'd already failed before starting an IT acquisition? As soon as individual government and contractor teams were assembled in the first operating SAFe ceremonies, the complaining began (i.e., he or she is incompetent, he or she is not doing their job, he or she is failing, he or she is not qualified, he or she doesn't look trustworthy, etc.)! Most of the IT acquisition workforce, be it government or contractor was nationally distributed, so the blame game continued during all ONLINE traditional and SAFe ceremonies. It created an operating environment of fear, uncertainty, and doubt. It was okay as long as someone else was being called a nincompoop, but it was inevitable that everyone's turn came sooner or later. It was worse if you weren't in the room or in all traditional or SAFe ceremonies, because the worst intense, dark, and hateful slander often took place behind closed doors! People were often judged on their gender, color, ethnicity, voice, background, status, and other superficial characteristics to the point of ridiculousness! Worse yet, the people who were especially good at the vicious blame game rose to the top or became government employees if they were especially vicious contractors! Well, modern social psychologists have determined that positive operating cultures lead to IT acquisition success and negative operating cultures undermine IT acquisition success (so much so, that lean-agile models now claim positive management attitudes as a critical success factor)! Well, if this government agency and contractor team thought SAFe would rescue them from a dark nihilistic blame-game oriented culture, they were dead wrong!* ***Lesson-learned**—Start being positive vs. negative, create an optimistic operating culture, compliment people instead of criticizing them, and for God's sake stop stabbing people in the back for creating a complex IT acquisition that's never gonna succeed*!
- **Immensely-Complex and Overly Optimistic IT Acquisition**. *When it comes right down to it, this government agency composed an immensely complex IT acquisition consisting of too many disparate moving parts that could never be reconciled in a million years. That is, there were too many requirements, too many legacy systems, too many contracts, too many people, too many epics, too many IT acquisition practices, etc. In general, this government agency, while NOT the largest and most well-endowed government agency, suffered from TOO-MUCH-MONEY syndrome (i.e., immense overconfidence and cognitive blindness associated with wealth, health, and longevity). Because resources were seemingly limitless, this government agency felt they could construct a new infinitely complex IT acquisition and resulting IT product and service, especially if they used SAFe (even just a little SAFe, like a miracle ointment on a large open traditional wound)! The answer, of course, was NOT just in adopting a lean-agile framework such as SAFe and bootstrapping it onto an immensely traditional government culture, IT acquisition operating environment, and impossible scope. The answer was not simply to execute SAFe ceremonies, but understand the core values and principles upon which SAFe is based (i.e., lean and agile thinking). Part of the lean paradigm is to keep-it-simple, reduce the scope and batchsize, build in excess capacity, limit work-in-process, and pivot and move from day to day, iteration to iteration, and program increment to program increment. A fundamental IT principle is the notion that IT itself is comprised of immense complexity, uncertainty, and unpredictability. Therefore, one must downsize the scope, use experimental methods to tease out the uncertainty, frequently measure outcomes, and continually pivot, move, and evolve the scope, direction, and even the vision as insidiously invisible tacit intangible immeasurable theories and hypotheses become visible quantifiable explicit facts, knowledge, data, and outcomes. One simply cannot predict an immensely complex scope, charter multiple disparate contracts, haphazardly apply traditional and SAFe practices, and jam all of this unpredictability into multiple integrated master schedules (IMS) renamed as roadmaps to make them sound more lean and agile, and then choose the most optimistic operating plan (well in advance of converting uncertainty to certainty)!* ***Lesson-learned**—Learn and master fundamental lean-agile principles, theories, and arithmetic, downsize immense IT acquisition scope, don't spend so much money just because you can, stop chartering so many disparate contracts, don't mix and match traditional and lean-agile practices like black magic, and create multiple competing complex IMSs called roadmaps hoping one of them will have the right answer*!
- **Multiple Competing Traditional and Lean-Agile Enterprise Models**. *This government agency, like most agencies, key suppliers, individual IT professionals, and even Fortune 500 firms have adopted immensely complex pantheons of traditional, lean, and agile IT operating principles, practices, models, tools, cultures, and job descriptions. That is, they seem to want it all, PfMP, FAC-P/PM, DAWIA, Systems Engineering, CMMI, ISO 9001, Lean, Agile, Scrum, Kanban, SAFe, LeSS, ITIL, etc. You name it, and they want it. Well it's a zero-sum game! There simply isn't enough money in the world to do everything, and certainly not all at once! Furthermore, many of these paradigms CONTRADICT one another (i.e., can't have a mile long IMS with CMMI activities in simple Scrum backlog with three two-week Sprints). It simply doesn't make sense. The organizational psychologist Edgar Schein noted in the 1960s that the CEO (leader) defined the culture of the enterprise based upon his or her vision, personality, or behavior. What this means is that if leaders are traditional, then employees, suppliers, and individuals will be traditional (and if they are mixed bag of pantheistic practices then so will everyone else)! This has led suppliers to spend billions of dollars on traditional paradigms (and they continue to do so). Now that government agencies are pivoting to Lean and Agile paradigms, suppliers cannot keep up, or simply add a new Lean and Agile god to their existing pantheon! Since traditional paradigms require millions of dollars and years to master, while lean and agile paradigms do not require high upfront investments, many suppliers continue to pour millions of dollars in operating capital (profits) into traditional paradigms, while expecting existing employees to implement lean and agile principles and practices with no additional resources (i.e., start using Scrum today while I spend $5 million on ISO 9001 or CMMI)!* ***Lesson-learned**—Traditional paradigms are antithetical to Lean-Agile models, so mixing and matching IT acquisition gods into a single pantheon is counterproductive, destructive, increases lead times rather than decreasing them, undermines IT success, and starves investments in Lean-Agile thinking (no wonder traditional IT acquisitions don't have the resources to learn and master Lean-Agile thinking principles and practices and jam millions of oversized business requirements into Scrum backlogs)*!

Rico, D. F. (2020). ROI of Lean Thinking, Retrieved January 29, 2020, from http://youtu.be/wkMfaPAxO6E

Appendix IV—Post SAFe Program Increment (PI) #4 Planning Retrospective, Observations, & Recommendations

SAFe Program Increment (PI) #3 came to a close with a little bit of uncertainty about midway through, but started picking up steam. Multiple key delivery contracts were still in limbo and remained so well into PI #4. The government agency rebooted and radically rebooted the overall program team structure in preparation and anticipation for the interaction and collaboration of multiple new delivery services contracts. However, there was still some uncertainty going into PI #4 planning. However, PI#4 planning ensued and was executed in spite of all of the contractual uncertainty and trepidation and was rather routine. The government leadership team tried to maintain as much positive energy as possible in spite of the obvious contractual issues. The program manager seemed cautiously optimistic about the overall program status and SAFe implementation, although the SAFe-based enterprise transformation program was going remarkably well. The PI#4 planning facilities continued to be somewhat of a challenge, with too many tables, not enough chairs, and very poor sound and visual quality. SAFe agile teams seemed somewhat disorganized and weren't as sharp as they had been in the first three PI planning events. The use of the agile application lifecycle management (ALM) tool continued to be a major challenge. The government agency seemed to be using the SAFe transformation to promulgate the use of the ALM tool across the entire enterprise. As such, there was much resistance to change from all levels—Outside of the program, across management teams, and within development teams. A key SAFe agile team was given impromptu on-the-job training on how to use the ALM tool for agile documentation purposes like Confluence, which should have been routine, but came across poorly.

This civilian public sector agency's SAFe transformation continues to be an exemplar in any sector, public or private, and the government leadership's commitment to SAFe is admirable! This is quite rare as few government leadership teams are the source of SAFe commitment, enthusiasm, and support. Another notable characteristic is the immense talent and motivation of both the government and contractor teams, which are making this SAFe implementation seem easy in spite of its many challenges, some of which are within the control of the agency and some of which are outside the control of the agency. Obviously, the blocked contracts seem to be outside of the control of the agency, although it was a mistake to scope the SAFe program so large as to require multiple large-scoped disparate contracts. Another element that's within the control of the agency is the obvious inexperience with lean thinking, lean portfolio management, and viciously limiting WIP to improve end-to-end delivery performance. Another shining element is the government's constant improvement, realignment, and reorganization of the overall SAFe program to keep progress moving forward. Some government leaders continue to use negative pressure to motivate people, while fewer government leaders recognize and apply positive pressure to keep the program moving forward. Obviously, there are still many barriers to communication, collaboration, and teamwork between the government and its suppliers, as well as supplier to supplier, which don't seem to be resolving themselves fast enough. There are still far too many cross cutting visible and invisible silos, cliques, and clans among both government and contractor teams that are hindering full cooperation, collaboration, teamwork, trust, quality, delivery speed, and overall SAFe effectiveness.

On a positive note, this government agency seems to be transforming its basic mindset, culture, psychology, and operating mode from traditional to SAFe thinking remarkably quickly, although the overall enterprise has a soft stance on its commitment to SAFe. Only time will tell about the future of SAFe on this program and within the agency.

<table>
<tr><th colspan="2">Post PI#4 Planning Observations</th></tr>
<tr><th>Strengths</th><th>Recommendations</th></tr>
<tr><td>• <i>Top down leadership-driven SAFe transformation.</i>
• <i>Highly talented and motivated buyer-supplier teams.</i>
• <i>Rigorously followed SAFe implementation roadmap.</i>
• <i>Good SAFe program increment planning ceremony.</i>
• <i>Routine apply SAFe program increment ceremonies.</i>
• <i>Top down application lifecycle management tool use.</i>
• <i>Top down use of basic SAFe dashboards and metrics.</i>
• <i>Continuous overall SAFe and program improvement.</i></td><td>• <i>Stay the course—Continuously improve SAFe use.</i>
• <i>Master and apply lean and agile thinking principles.</i>
• <i>Viciously limit WIP across all program functions.</i>
• <i>Apply highly visual lean, agile, and SAFe practices.</i>
• <i>Create vertical and horizontal "one team" culture.</i>
• <i>Create positive and enjoyable program environment.</i>
• <i>Simplify application lifecycle management tool use.</i>
• <i>Combine and simply multiple disparate contracts.</i></td></tr>
</table>

Rico, D. F. (2020). 32 Principles and Practices of Highly-Successful Scaled Agile Framework (SAFe) Implementations. Retrieved February 16, 2020, from http://davidfrico.com/safe-principles.pdf

- **Interfacing SAFe to the Real World**. *Interfacing lean and agile frameworks such as SAFe to enterprises is a major challenge. Oftentimes, large well-funded public and private sector transformation initiatives are run by powerful, intelligent, highly motivated, and deeply pocketed executives. They tend to be natural born multi-taskers, their brains operate in overdrive, their leaders are even more powerful and demand constant change in direction, and of course, peer-level enterprise stakeholders constantly criticize, harass, and challenge them to accept new out of scope requirements every day. With this constant churn, program managers often demand dozens of new out of scope requirements every day and insist that all cumulative requirements be accounted for—Which are usually tracked in daily action item logs! In other words, they are operating in anything but a lean and agile mindset with extremely limited WIP. It takes years to commit to a single Epic MVP and when they do, the paper it is written on—or ALM entry—is instantly obsolete before a SAFe solution or agile release train can conduct a solution or program planning event and codify or execute on a 90-day plan! It's unrealistic for solution or agile release trains to expect program managers to carefully think or plan their lives in 90-day SAFe solution/program increments! It's also unrealistic for program managers to add dozens of daily new requirements to solution and agile release trains in a cumulative fashion! There has to be a middle ground in SAFe!* ***Lesson learned****—Teach program managers lean, agile, and SAFe thinking fundamentals—As well as the dangers of daily churn and setting unrealistic expectations that dozens of out-of-scope epics, capabilities, features, user stories, and tasks can be added to portfolio, solution, and agile release trains. At the same time, portfolio, solution, and ARTs shouldn't expect executives to give clear up-front direction, setup 90-day solution and agile release trains, and expect executives to be happy with team performance when large new enterprise epics, capabilities, features, user stories, and tasks emerge in daily executive standup meetings. SAFe coaches are especially challenged to establish, manage, and ensure a dynamic cadence vs. out-of-the-box 90-day SAFe PIs based on clear up-front static direction!*
- **Setting Up Small Cross-Functional Teams**. *Setting up small cross functional SAFe teams in traditional enterprises is also a major challenge. Traditional enterprises are chock full of functional silos. Executives exist in far off corporate offices, teleworking middle managers are wed to long term traditional project plans and harrowing 18 month releases, and enterprise architecture, infrastructure, and tools teams operate in a vacuum and don't want to be bothered with human interactions and collaborative DevOps practices. Furthermore, part-time jet setting agile coaches and scrum masters are allocated to multiple part-time jobs, aspire to be project micro-managers, and don't have time to collaborate, and, of course, developers may be in another country all-together! With all of these functional silos operating in a vacuum, it's just too easy to codify or treat them as their own lean, agile, or SAFe team, which simply fragments solution and agile release trains when one or more of these groups can't be bothered, blocked, or synchronized on the same cadence. The easiest temptation is to form a lean, agile, or SAFe development team that is instantly blocked by an executive, middle manager, enterprise architect, tools team, or non-participating traditional or agile team.* ***Lesson-learned****—Form feature teams composed of at least one member from each silo, be it executive, middle manager, enterprise architect, tools team, tester, business analyst, UX designer, coder, Scrum master, Product Owner, etc. That is, make at least one member responsible for the success of a single set of features, user stories, and tasks. That means, each cross functional member must attend solution or agile release train planning, iteration planning, daily standups, demonstrations, retrospectives, etc. If any member is blocked by the necessity to have a cross-functional dependency resolved, that person should be present on the team to immediately unblock the impediment. Cross functional teams should be capable of delivering features, user stories, and tasks without direct interaction with another silo—With bothersome email. The next SAFe challenge is to synchronize multiple cooperating cross functional teams to work on epics and capabilities.*
- **Keep SAFe Simple, Visual, Face-to-Face, & FAST**. *Keeping SAFe simple, visual, and face-to-face in large traditional enterprises is a relentless and never-ending challenge. Traditional enterprises grew up on the manufacturing process era of the late 1900s and are chock full of enterprise architects, integrated master schedules, automated lifecycle management tools, and CMMI and ISO 9001 process experts. Furthermore, large traditional enterprises are often staffed with wickedly talented left-brained analysts that want to create infinitely complex flow charts for every conceivable daily routine, dozens of metrics dashboards, and business rules for lean, agile, traditional, and SAFe practices. First of all, if you have a shared services team dedicated to tools, metrics, processes, business rules, and flow charting complex multi-level SAFe ceremonies, you've defeated the purpose of having small cross-functional teams focused on delivering business value fast. Remember, your portfolio, program, and project managers are swimming around in shark tanks dreaming up dozens of new epics, capabilities, features, user stories, and tasks every day! The last thing you wanna do is create a barrier of complex business rules, flowcharts, processes, ceremonies, dashboards, tools, and other roadblocks between power hungry executives and ambitious delivery teams who need to hear a portfolio, large solution, and program vision directly from the lips of powerful executives early and often. The number one complaint of solution and agile release trains is absentee executives, visions, and a clear understanding of high-priority business needs. Likewise, the number one complaint is that delivery teams are languishing in last quarter's obsolete business requirements.* ***Lesson-learned****—Don't form autonomous teams of SAFe middle managers to set up complex business rules, flowcharts, dashboards, and agile lifecycle management systems as a barrier to fast, dynamic, and market responsive delivery of business value to enterprise, portfolio, and large solution themes, epics, capabilities, and features. Keep SAFe implementation lean, mean, live, face-to-face, visual, dynamic, informal, and fast moving. There's no reason why a delivery team cannot fire off multiple new business experiments quickly emerging from daily executive standup meetings. Don't fall in love with silver bullets from the 1970s in the form of sleek new agile lifecycle management systems that are already obsolete, resulting in millions of dollars of wasteful licensing and setup costs and work in process (WIP) routines and deliverables, because SAFe is evolving too fast and so are the market, technological revolution, competitors, and executives.*