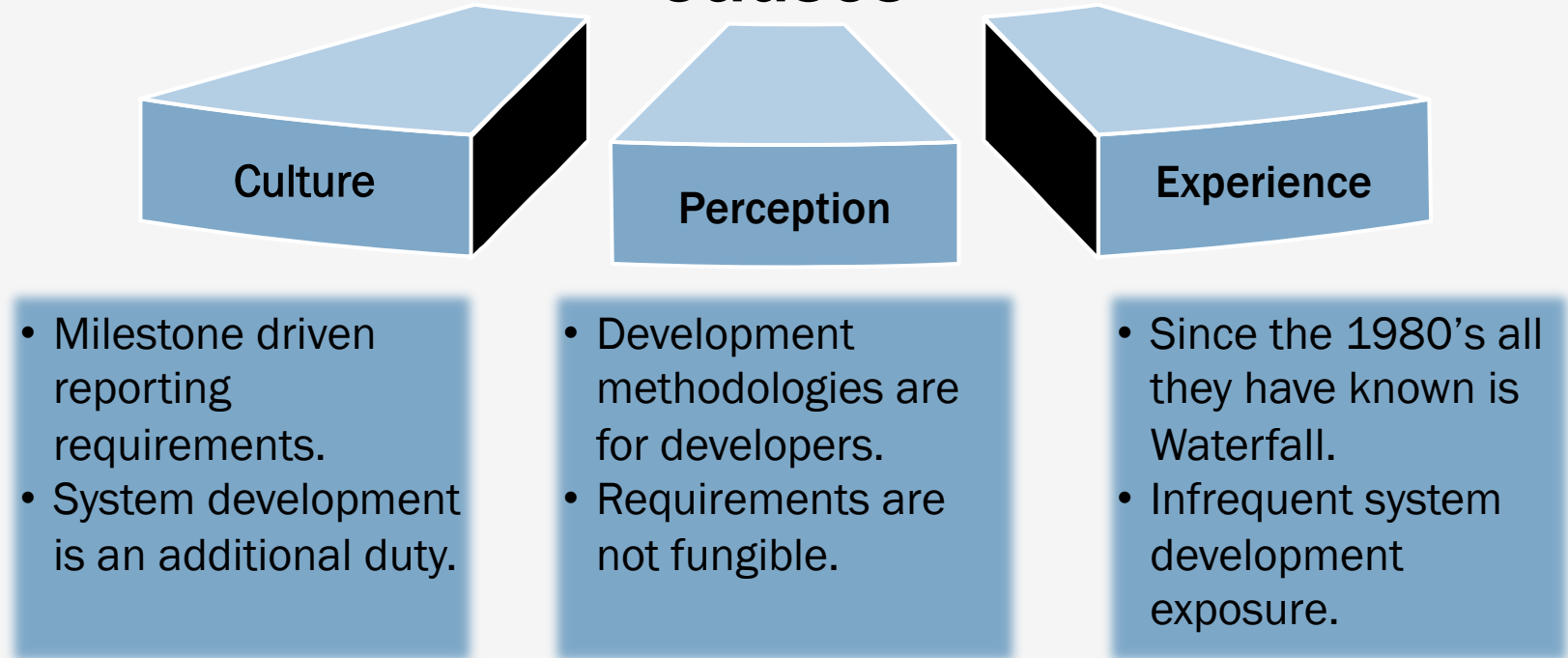Addressing the elephant in the room through shared experiences.

ENGAGING A PRODUCT OWNER ON A GOVERNMENT CONTRACT

Challenges and Solutions

PROBLEM: When a Product Owner is not capable or actively engaged, the development team looses its ability to evolve requirements, which suffocates *functional innovation*.

Causes

Culture

Perception

Experience

- Milestone driven reporting requirements.
- System development is an additional duty.

- Development methodologies are for developers.
- Requirements are not fungible.

- Since the 1980's all they have known is Waterfall.
- Infrequent system development exposure.

Agile Development teams often seek out tools and techniques to create great systems, however too frequently the elephant in the room holding them back is the Product Owner. This talk shares solutions I have used for challenges I see again and again on government contracts.
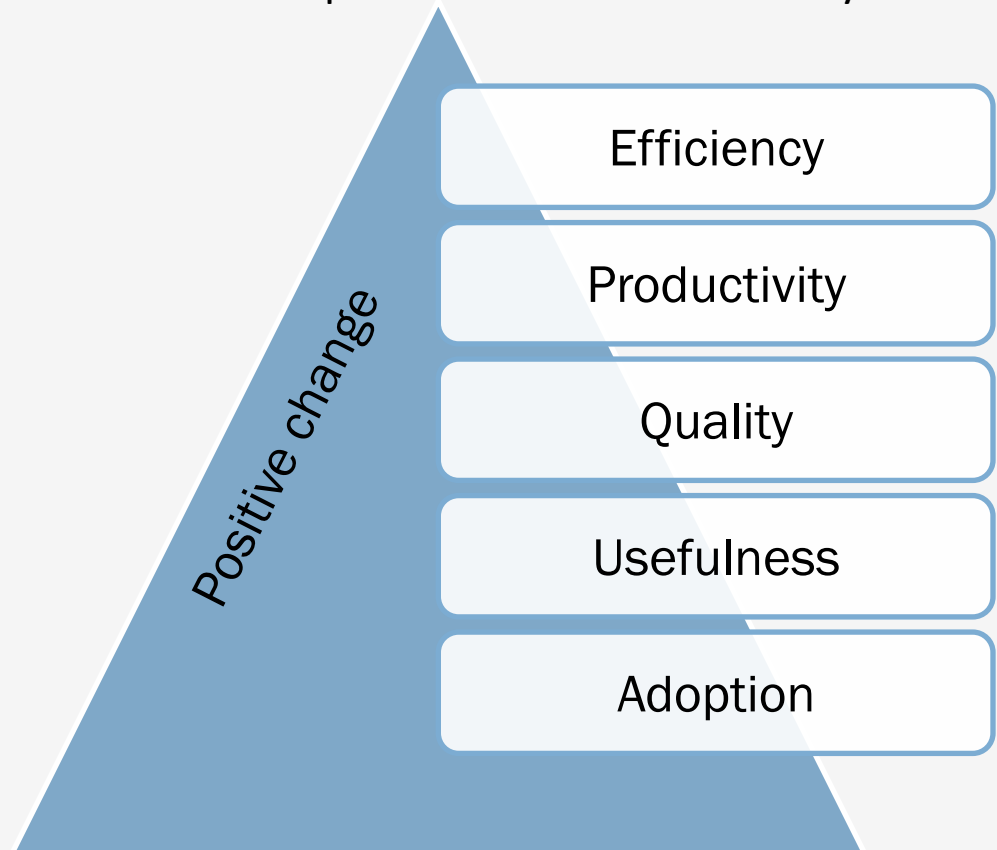
FUNCTIONAL INNOVATION: The user community is interested in the sausage taste, not in how the sausage is made. To create great software, the software must positively change what that user community is doing today.

Functional Innovation

- Can never be prescribed, must be discovered.
- Discovery originates from the feedback of real users.
- Is not an "enhancement" request to be built later, rather an on-going commitment to embrace change.

Functional innovation creates positive change in efficiency, productivity, quality, usefulness, and adoption. It is driven by decades of functional experience within an organization and diffused through the technical implementation of the new system.

Positive change

- Efficiency
- Productivity
- Quality
- Usefulness
- Adoption

HOW DID IT COME TO THIS? Three effects have had significant influence on engaging Product Owners in government Agile implementations.

Waterfall Effect

- Discrete events make coordination easier.
- Known levels of involvement from the functional community at pre-determined times.
- Decades of experience developing software using Waterfall.
- Waterfall applies to large non-IT procurements, often with great success.
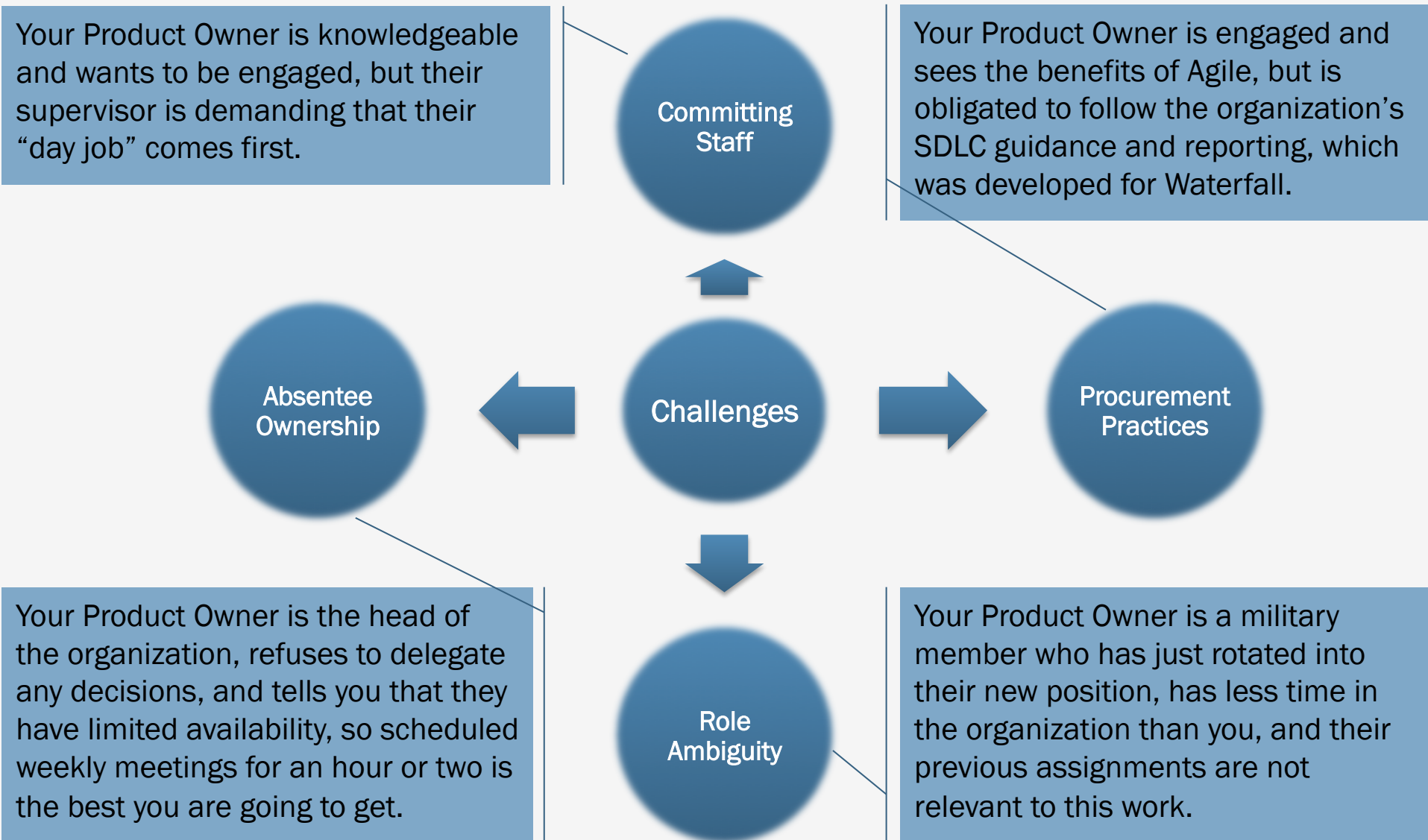
Contract Effect

- Procurement is easier when bids can be evaluated against a predetermined set of requirements.
- Evaluation of contractor performance is easier to gauge when requirements are predetermined.
- Traceability, knowing clearly you received 100% of what you bought.

Contractor Effect

- If we paid for a system, why should we have to lend a hand?
- The temporary nature of contractors drives a desire to create documentation that will last after the contractor leaves.
- Development is what contractors do, we manage and report.

These effects have driven a solution that benefits those that must manage the system to the detriment of those that must use the system.

CHALLENGES: While not an exhaustive list and not exclusive to government contracting, four recurring challenges with Product Owners that I have faced are shown below.

Your Product Owner is knowledgeable and wants to be engaged, but their supervisor is demanding that their "day job" comes first.

Your Product Owner is engaged and sees the benefits of Agile, but is obligated to follow the organization's SDLC guidance and reporting, which was developed for Waterfall.

Committing Staff

Absentee Ownership

Challenges

Procurement Practices

Role Ambiguity

Your Product Owner is the head of the organization, refuses to delegate any decisions, and tells you that they have limited availability, so scheduled weekly meetings for an hour or two is the best you are going to get.

Your Product Owner is a military member who has just rotated into their new position, has less time in the organization than you, and their previous assignments are not relevant to this work.

CHALLENGE: COMMITTING STAFF. Your Product Owner is knowledgeable and wants to be engaged, but their supervisor is demanding that their "day job" comes first.

- Include the Product Owner's leadership in a **Special Demonstration** that includes members from the broader user community. This demonstration is in addition to the normal Sprint Reviews and is designed to showcase the working software and generate excitement in the user community. By being "public" about the progress, the fear of failure starts to increase and leadership tends to shift resources to you, so that you succeed.

- Be **upfront** on the level of commitment required from the Product Owner during the project kickoff. Discuss expectations. Stress that you are reducing their time at the beginning of the project by not doing months long detailed requirements gathering effort.

- Show **success,** early and often. Nothing drives commitment better than deploying working software. Frequent deployments, even to a test environment show a return on investment of their worker's time and the importance of their participation.

- Include **leadership's priorities** early in your development. When they see that they are getting what is important to them, they will be more committed to your project. However if all they see is design documentation and "foundation" code, they will rapidly lose interest.

CHALLENGE: PROCUREMENT PRACTICES. Your Product Owner is engaged and sees the benefits of Agile, but is obligated to follow the organization's SDLC guidance and reporting, which was developed for Waterfall.

- It never hurts to ask for an **Exception** to the existing practices. If that does not work, attempt a **Pilot Program,** then build on the successes and find the common ground.

- **Compromise.** The organization wants to be successful in this effort; they are looking to you help them find a way to achieve success and stay compliant. Examples:

  - The existing methodology required detailed **architecture** diagrams before development. They accepted an approach where a high-level design was provided at the beginning and then the details were provided at the end of the project.

  - The **Critical Design Review** (CDR) was replaced by an Incremental Design Review (IDR) that occurred after each Sprint Review; avoiding a big design up front approach.

  - A work breakdown **schedule** was required before funding could be allocated. We "walked the skeleton", identified Sprint Themes, performed release planning, included User Stories that were "buffers" for refactoring, and provided a schedule baseline.

  - **Reporting metrics** and **traceability** were a required deliverable, so we created a Functional Traceability Matrix (FTM) that traced written requirement to wireframe to user story to automated test.

CHALLENGE: ROLE AMBIGUITY. Your Product Owner is a military member who has just rotated into their new position, has less time in the organization than you, and their previous assignments are not relevant to this work.

- We bring the experience in software development and Agile. We need to do the **Outreach** to Product Owners to help them be successful and come up to speed speed quickly in what we are doing and why we are doing it. We will have more books in our private library on Agile than they will, share a book with them.

- Point out **Certification Programs** that are available.

- Walk the Product Owner through several **Activities** that the development team does so that they understand the process better. Again, we have the experience using Agile and we need to share that. There are many techniques to drawn quality out of a team and many tools to increase a developer's efficiency, but there are few when it comes to Product Owner involvement. One of my personal proudest moments was when I had a Product Owner tell me that he had instituted scrum within his community volunteer organization because he was tired of ineffective meetings.

CHALLENGE: ABSENTEE OWNERSHIP. Your Product Owner is the organizational head, refuses to delegate, and has limited availability, so scheduled weekly meetings for an hour or two is the best you are going to get.

- In my experience, this challenge is the most difficult challenge to overcome. When you attempt to do Agile without a Product Owner, you are not doing Agile. You can use **some Agile techniques**, but you forfeit many of the benefits you get from actually doing Agile and you will not be able to build great software.

- You might be able to **Proxy** the Product Owner by using a Subject Matter Expert (SME). However the proxy will not have the authority to make the definitive decisions that you require on a daily basis. A proxy can be deadly to the team if the scrum master attempts to wear both hats. If you have to proxy the Product Owner, then it helps to have a dedicated person with some functional knowledge in that role separate from the development/testing staff. The risk with a proxy is that when you get to a Sprint Review, the true Product Owner does not accept the product because it isn't what they intended.

- The most effective solution that I have used in this situation is establishing a formal **Lead Product Owner / Product Owner** relationship. In this structure, the Product Owner makes all the decisions, but off-line vets those decisions with the Lead Product Owner. This situation keeps the Lead Product Owner "in-charge" but gives the team access to frequent feedback. Because they sync up more frequently, the impact of necessary corrections is less than if you have to wait until the Sprint Review to get input. Still, you will not achieve that functional innovation that is key to great software.

CONCLUSION.

Great systems require active, capable Product Owners. Functional innovation is not possible without their commitment and involvement in the project. Too often in government contracting, the Product Owner is an Absentee Owner. Development teams in this situation must face the elephant in the room if they desire to build a system that brings positive change in efficiency, productivity, quality, usefulness, and adoption. Below is a summary of the challenges and solutions that I have experienced and used.

Challenge	Solutions
Committing Staff	• Introduce a fear of failure through Special Demonstrations. • Be upfront on the commitment. • Show success early and often. • Include leadership's priorities.
Procurement Practices	• Try a Pilot Program approach. • Split architectural design reviews leaving the detail until the end. • Replace CDRs with Incremental Design Reviews. • Generate a WBS from a roadmap. • Develop a Functional Traceability Matrix.
Role Ambiguity	• Share a book from your personal library. • Recommend certification programs. • Immerse them in development activities for first hand experience.
Absentee Ownership	• Proxy using an SME. • Form a Lead Product Owner / Product Owner team.