Most Agile Principles Directly Reflect a Service Mindset
SW Development & Test as a Service: Why? How?
This work was created in the performance of Federal Government Contract Number FA8721-05-C-0003 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center. The Government of the United States has a royalty-free government-purpose license to use, duplicate, or disclose the work, in whole or in part and in any manner, and to have or permit others to do so, for government purposes pursuant to the copyright license under the clause at 252.227-7013.

This Presentation may be reproduced in its entirety, without modification, and freely distributed in written or electronic form without requesting formal permission. Permission is required for any other use. Requests for permission should be directed to the Software Engineering Institute at permission@sei.cmu.edu.

NO WARRANTY

THIS MATERIAL OF CARNEGIE MELLON UNIVERSITY AND ITS SOFTWARE ENGINEERING INSTITUTE IS FURNISHED ON AN “AS-IS” BASIS. CARNEGIE MELLON UNIVERSITY MAKES NO WARRANTIES OF ANY KIND, EITHER EXPRESSED OR IMPLIED, AS TO ANY MATTER INCLUDING, BUT NOT LIMITED TO, WARRANTY OF FITNESS FOR PURPOSE OR MERCHANTABILITY, EXCLUSIVITY, OR RESULTS OBTAINED FROM USE OF THE MATERIAL. CARNEGIE MELLON UNIVERSITY DOES NOT MAKE ANY WARRANTY OF ANY KIND WITH RESPECT TO FREEDOM FROM PATENT, TRADEMARK, OR COPYRIGHT INFRINGEMENT.
Purpose & Objectives

My purpose: to help you take advantage of service science concepts that provide high leverage for improving the performance of product development-focused organizations.

copyright 2012, Carnegie Mellon University. All rights reserved.
- SW Product developers who want to have productive relationships with ongoing customers
- SW Product developers who combine product development with consultancy, user training, or other services to their customers
• Different ways of looking at product development
• Non-development things a "customer intimate" product developer can do to improve performance
• Developers using agile methods are already on this path
• Summary

Agenda
What is a service?

Services are delivered through the operation of a service system.

Services imply ongoing relationships governed by agreements.

Services are often simultaneously produced and consumed.

Services have a different business rhythm than goods.

<table>
<thead>
<tr>
<th>Goods</th>
<th>Develop</th>
<th>Deliver</th>
</tr>
</thead>
<tbody>
<tr>
<td>Service</td>
<td>Develop</td>
<td>Deliver</td>
</tr>
</tbody>
</table>
service: an intangible, non-storable product (e.g., operations, maintenance, logistics, and IT).
SuZ' favorite description of the key characteristic of a good service:
Co-Creation of Value
Services have a different business rhythm than goods.
How does service delivery differ from product development?

- Relationship often trumps content
- Attentiveness
- Immediacy and simultaneity
- Attention to details
- Consumers notice details immediately
- Visibility into how problems are resolved
- Real time communication
- Value strongly influenced by point of contact
- Quality can be objective or amorphous

- Iterating toward final correct product frequently expected
- Quality frequently defined in objective terms
- Value easily connected to specific performance results, independent of point of contact
- Communication infrequently real time
- Circuitous path from failure to producer

copyright 2012, Carnegie Mellon University. All rights reserved.
• Relationship often trumps content
• Attentiveness
• Immediacy and simultaneity
• Attention to details
• Consumers notice details immediately
• Visibility into how problems are resolved
• Real time communication
• Value strongly influenced by point of contact
• Quality can be objective or amorphous
• Delivered solution is tangible, physical
• Fixed-term relationship based on a delivery contract
• Delivery of product generally takes place after development (and maybe after manufacturing)
• More time spent on development

• Iterating toward final correct product frequently expected
• Quality frequently defined in objective terms
• Value easily connected to specific performance results, independent of point of contact
• Communication infrequently real time
• Circuitous path from failure to producer
How is product development becoming more similar to service delivery?
Similarities are most visible in an Agile environment....

- Relationship often trumps (initial) content
- Attentiveness, esp with customer collaboration
- Immediacy and simultaneity
- Attention to details
- Customers notice (software) details immediately
- Visibility into how problems are resolved
- Real time communication
- Value strongly influenced by point of contact
- Quality can be objective or amorphous
Why think about adopting a service mindset if you're a product developer?

- Do we provide training services to others?
- Do we provide software maintenance services to others?
- Do we provide analysis or engineering services to others?
- Do we provide configuration or other logistics services to others?
- Do we do provide test services to others?
We and our customers spend much of our time delivering services to others
But are we providing them mindfully?
Are we taking advantage of SERVICE concepts best practices to get and keep the customer relationships we need to have ongoing success?
Setting the Context of Superior Service

The Discipline of Market Leaders

3 "disciplines" all successful businesses must be competent in:

- Operational excellence
- Product innovation
- Customer intimacy

BUT...

World class organizations excel at only one of these
If, as an organization, you are trying to excel at (or even just be competent in) Customer Intimacy, you are engaging explicitly in building ongoing customer relationships -- you are adopting a service mindset toward your customers.
Most Agile Principles Directly Reflect a Service Mindset
Principles behind the Agile Manifesto

We follow these principles:

Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.

Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.

Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.

Business people and developers must work together daily throughout the project.

Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.

The most efficient and effective method of conveying information to and within a development team is through the daily Scrum.

One of the ways we demonstrate the value of the relationship vs the convenience to the developer.

A key way we build the relationship for an ongoing service.

Real time communication...

A key organization behavior tenet regardless of service or product development.

The service by which we co-create value.
Self-organizing teams is one of the ways we recognize the value of relationships in achieving goals.

Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.

Continuous attention to technical excellence and good design enhances agility.

Simplicity— the art of maximizing the amount of work not done—is essential.

Working software is the primary measure of progress.

The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.

Give them the environment and support they need, and trust them to get the job done.

Build projects around motivated individuals.

Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.

Business people and developers must work together daily throughout the project.

Agile processes harness change for the customer's competitive advantage.

One of the ways we demonstrate the value of the relationship is the convenience to the developer.

A key way we build the relationship for an ongoing service.

Service providers value a sustainable, predictable rhythm.

Real-time communication is an important characteristic of a service relationship.

The service is the vehicle by which we create value, through early and continuous delivery of valuable software.

Moving toward simultaneous production and consumption as much as feasible.
Give them the environment and support they need, and trust them to get the job done.

The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.

Working software is the primary measure of progress.

Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.

Continuous attention to technical excellence and good design enhances agility.

Simplicity—the art of maximizing the amount of work not done—is essential.

The best architectures, requirements, and designs emerge from self-organizing teams.

At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.

real time communication—an important characteristic of a service relationship

service providers value a sustainable, predictable rhythm

one of the quality attributes the software development service tries to achieve

one of the best practices in service delivery is to consistently look for ways to improve the service

tenet regardless of service or product development

how to allow the customer to see the details and have enough visibility to correct course quickly

these are particular attributes of the "software" service we provide

Self-organizing teams is one of the ways we recognize the value of relationships in achieving goals
Summary: What do we do differently if we mindfully adopt a service mindset?

- Think about the "service agreement" we want with our customers (internal or external)
- Look at the things that can go wrong with the relationship as much as looking at what can go wrong with the product
- Look at and explicitly manage capacity and availability issues related to providing our software or test service
- Be mindful and explicit about the boundaries of our "service catalog": do we do or not do installation and user training, for example
- Be mindful of what it takes to transition our products into use and to evolve them (both from an architecture and a whole product viewpoint)
- Think about service continuity--how do we continue to provide our software or test service in the face of a significant disruption?
- Explicitly develop our service offerings, not just our product components
Contact information

SuZ Miller
smg@sei.cmu.edu

General
info@sei.cmu.edu