Reflecting DevOps in IT-CMF
Workshop Outline

Part-1 (10 mins)
- Introductions, Agenda, and Context
- DevOps Overview – Working Definition, Principles and Cycle
- Focus on key IT-CMF Capability: Solutions Delivery (SD)

Part-2 (20 mins)
- Exercise & Discussion on Solutions Delivery (SD)
- Continuing the Work
Development Process

We are here

- Multivocal literature review
- Member feedback
- Expert Interviews

Development Stage
Evaluation/Quality Stage
Workgroup

**Objective:** To update IVI product structures and content to ensure that it provides the support needed to accelerate your DevOps capability

To establish a Community of Practice

**Members:**

- Linda Beckett  |  AIB
- Graeme Clarke  |  SQS
- Eamon Kelleher |  Irish Revenue
- John McHale    |  NashTech
- Laura Mc Quillan |  IVI
- Shaun Percival |  Bearing Point
- Brian Skehan   |  ESB, Electric Ireland
- Louise Veling  |  IVI
At work!
## DevOps Working Definition & Principles

### Definition

DevOps refers to a set of technical, architectural and cultural practices aimed at increasing the efficiency and effectiveness of delivering business needs into production, through improved communication and collaboration between business, development and IT operations.

### Principles

- Design services to be **lightweight**, independent, and to **serve a business goal**
- Deliver **small batches** of services and features in **short cycles** of design, build, test, deploy
- **Orchestrate** the delivery cycle using **automation**, integrated tooling & configuration management
- Maintain a **commitment to quality**, security and compliance, with on-going validation, monitoring and feedback
- **Balance speed and agility with stability and continuity**
- Maintain a focus on **business/user needs** and an end-to-end view of services offerings
- Ensure an organizational commitment to **collaboration & shared ownership**, with defined roles and responsibility
- Use **Agile** and Lean practices in **cross-functional** teams
- Embed **learning**, continuous improvement and innovation practices
DevOps Cycle

- Collaboration & Ownership
- Feedback & Learning
- UX
- Data & InfoSec
- Architecture
- Infrastructure
- Instrumentation & Monitoring
- Orchestration & Automation
- Design
- Construct
- Test
- Support
- Deploy
- Release

Service Users
DevOps Team
Service Owner

Innovation Value Institute © 2017
DevOps and IT-CMF

Service Users

DevOps Team

Service Owner

Support -> Deploy -> Release

UX

Value Needs

BAR Benefits Assessment & Realisation
PM Portfolio Management

EAM Technical Infrastructure Mgmt.
EAM Enterprise Architecture Mgmt.

UED User Experience Design
BPM Business Process Mgmt.

SD Solution Delivery
SRP Service Provisioning

SAI Service Analytics & Intelligence
PPM Project & Programme Mgmt.

EIM Enterprise Information Mgmt.
ISM Information Security Mgmt.

Today’s Focus

Innovation Value Institute © 2017
## Solutions Delivery (SD) Capability Building Blocks (CBBs)

**Definition**
The Solutions Delivery (SD) capability is the ability to design, develop, validate, and deploy IT solutions that address the organization’s business requirements and opportunities.

<table>
<thead>
<tr>
<th>Category</th>
<th>Capability Building Blocks</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Design Phase</strong></td>
<td><strong>Requirements</strong></td>
<td>Manage requirements and their traceability throughout the IT solutions delivery life cycle to serve business needs.</td>
</tr>
<tr>
<td></td>
<td><strong>Design Conceptualization</strong></td>
<td>Apply architecture principles and guidelines to inform the design of IT solutions to meet requirements.</td>
</tr>
<tr>
<td><strong>Develop Phase</strong></td>
<td><strong>Fabricate</strong></td>
<td>Construct IT solutions based on design principles and standards – for example, multi-tier architecture, coding, and security.</td>
</tr>
<tr>
<td></td>
<td><strong>Test</strong></td>
<td>Conduct validation testing to ensure that IT solutions meet specified requirements. This can include unit, integration, system, user acceptance, and regression testing.</td>
</tr>
<tr>
<td><strong>Deploy Phase</strong></td>
<td><strong>Release Management</strong></td>
<td>Manage the deployment of IT solutions into the operational environment.</td>
</tr>
<tr>
<td></td>
<td><strong>Version Control</strong></td>
<td>Manage the control of versions that occur during the solution’s delivery life cycle using appropriate methods – for example, methods for initiating, defining, evaluating, and approving/disapproving proposed changes.</td>
</tr>
<tr>
<td><strong>Adoption of Solutions Delivery Methodologies</strong></td>
<td><strong>Methods and Practices</strong></td>
<td>Ensure the availability and use of IT solutions delivery methods and practices – for example, requirements management, configuration management, and release management.</td>
</tr>
<tr>
<td></td>
<td><strong>Practice Evolution</strong></td>
<td>Evolve IT solutions delivery approaches and methodologies in response to business needs and in line with industry practices.</td>
</tr>
</tbody>
</table>
Solutions Delivery (SD) – Member Feedback, Top issues

- “It needs to be clearly able to accommodate contemporary approaches – like agile, DevOps”
- “Phases may not be the best way to structure, how about principles?”
- “SD is huge, overloaded”
- “How does SD relate to SRP [service provisioning], what's the distinction between solution and service?”
DevOps Workshop Outputs
### Solutions Delivery (SD) Capability Building Blocks (CBBs)

<table>
<thead>
<tr>
<th>Category</th>
<th>Capability Building Blocks</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Needs</td>
<td>Understanding Needs</td>
<td>Identify and understand needs (e.g. user requirements, usage analytics, issues)</td>
</tr>
<tr>
<td></td>
<td>Business Alignment</td>
<td>Align solutions with business strategy and value</td>
</tr>
<tr>
<td></td>
<td>Feature Management</td>
<td>Manage requirements and priorities</td>
</tr>
<tr>
<td></td>
<td>Practices and Methodology</td>
<td>Identify and select delivery practices and principles to meet needs</td>
</tr>
<tr>
<td>Quality</td>
<td>QA/Test</td>
<td>Conduct testing (e.g. code review, regression testing, user acceptance)</td>
</tr>
<tr>
<td></td>
<td>Feedback and Analytics</td>
<td>Gather feedback and analytics (e.g. instrumentation, survey, user comments)</td>
</tr>
<tr>
<td>Solution</td>
<td>Architecture Principles</td>
<td>Apply architecture principles</td>
</tr>
<tr>
<td></td>
<td>Design</td>
<td>Design solutions with consideration of functional and non-functional needs</td>
</tr>
<tr>
<td></td>
<td>Construct Solutions</td>
<td>Develop/construct and evaluate solutions</td>
</tr>
<tr>
<td>Pipeline</td>
<td>Release Design</td>
<td>Design the orchestration of the integration, deployment and release pipeline</td>
</tr>
<tr>
<td></td>
<td>Automation and Orchestration</td>
<td>Automate and orchestrate the pipeline, using an integrated suite of tools</td>
</tr>
<tr>
<td></td>
<td>Release Management</td>
<td>Manage the integration, test and deployment of services, and their delivery into production</td>
</tr>
<tr>
<td></td>
<td>Environment and Configuration Management</td>
<td>Manage the environments, software, platform and service component configurations and relationships</td>
</tr>
</tbody>
</table>
Workshop

1. On your own (2 min)
   - Read through the proposed set of Solutions Delivery (SD) Building Blocks
   - Note down your reactions

2. In groups of 4-5 (10 min)
   - Decide who will give feedback
   - Explore these questions
     - Do any of these Building Blocks not belong?
     - Are there any Building Blocks missing?
     - Any other thoughts? (Time allowing!)

3. Feedback to the room (1-2 min per group)
CONTINUING THE WORK
Development Process

- Foundational Research
- Initial Scoping
- Workgroup Agreement
- Initial Review
- Design & Build
- Expert Review
- Evaluation in Use (Pilot)
- Final Review
- Feature Release

Development Stage
Evaluation/Quality Stage

We are here
Continuing the work

- Do you want to give feedback or pilot some of the DevOps assets?
- Sign up!
THANK YOU!