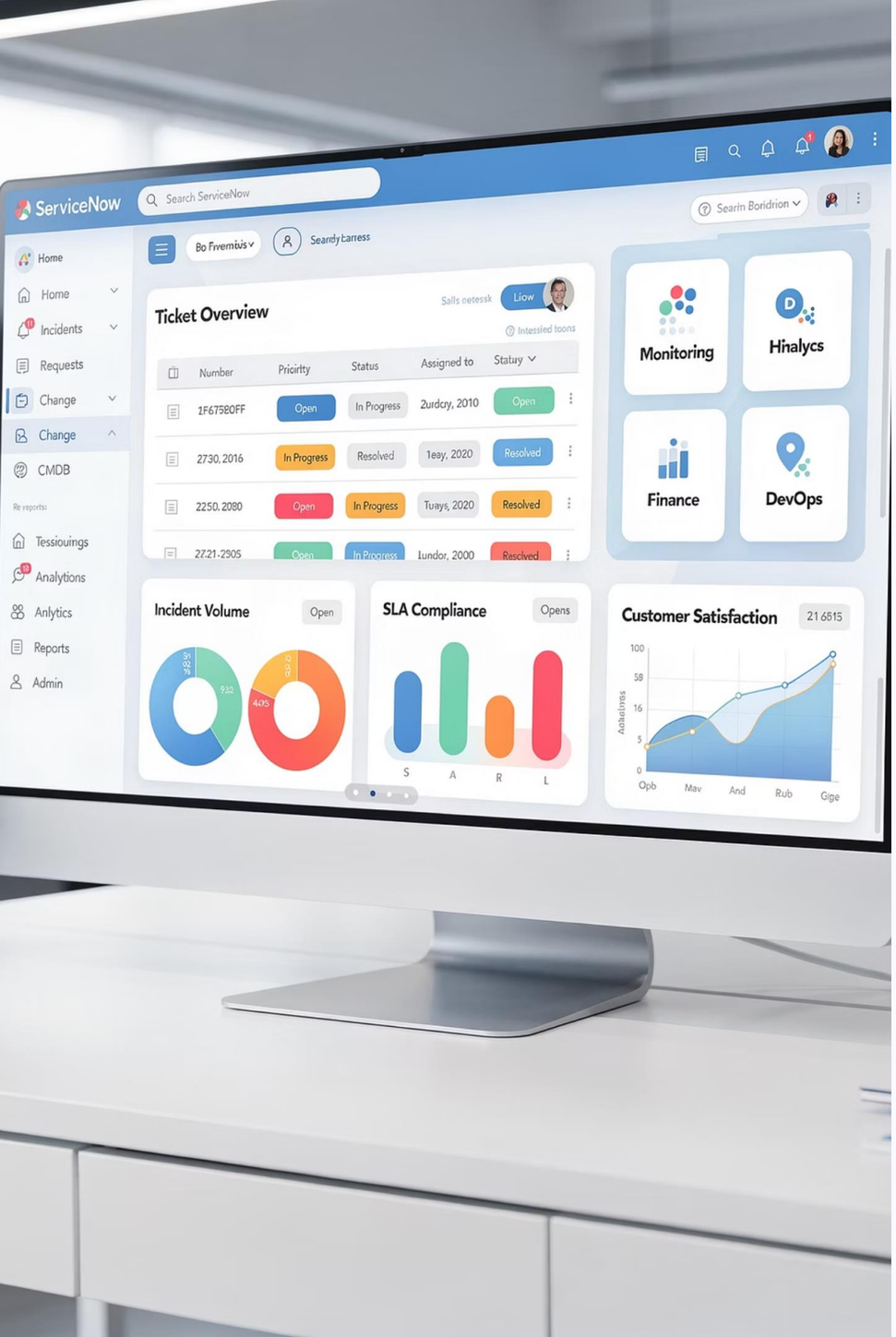
DevOps at Scale for ServiceNow and Integrations

One Backlog, One Team, Real Results

by Kimberly Wiethoff, MBA, PMP, PMI-ACP

Managing Projects The Agile Way

#ServiceNow #DevOps #ITSM #CSM #FSM #PlatformEngineering #SystemsIntegration #WorkflowAutomation #EnterpriseArchitecture #DigitalTransformation #AgileLeadership #DeliveryExcellence #CustomerExperience #ManagingProjectsTheAgileWay



The Platform Reality

DevOps is often discussed in the context of application development, but some of the most meaningful transformations happen in **platform and integration ecosystems**–particularly around ServiceNow.

At scale, ServiceNow is not just a workflow engine. It becomes a mission-critical platform where reliability, speed, and coordination directly impact customers, revenue, and operational cost.

Applying DevOps principles here is not optional–it's essential. The stakes are higher, the interdependencies more complex, and the need for disciplined execution more acute than in traditional application development.

This is where platform thinking meets operational excellence.

Why ServiceNow DevOps Is Different

ServiceNow DevOps is not about pushing code faster for its own sake. It's about reducing friction across teams, improving reliability of integrated workflows, and accelerating time-to-value without sacrificing control.

Multiple Modules

ITSM, CSM, FSM working together

Deep Integrations

Enterprise systems tightly coupled

Shared Configuration

Many teams, one platform

High Blast Radius

Mistakes impact multiple systems

This complexity makes **discipline, automation, and clarity of ownership** even more important than in traditional development environments.

One Backlog: The Foundation of Scaled DevOps

The most effective ServiceNow organizations operate with a single backlog spanning ServiceNow configuration, integrations, and automation. This creates clear prioritization based on business outcomes and shared visibility across onshore and offshore teams.

1

Multiple Backlogs Create

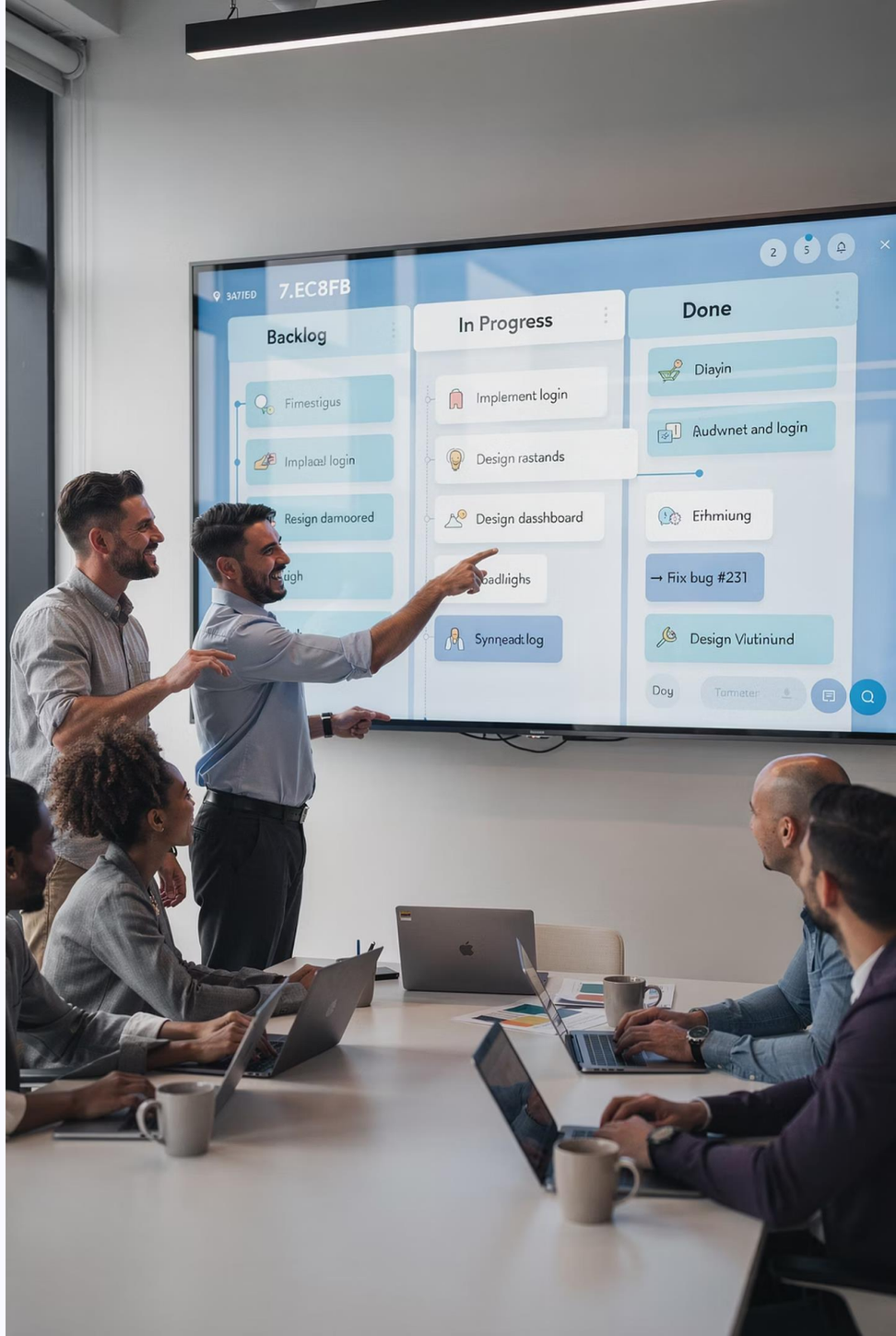
Artificial handoffs, misaligned priorities, and delays

2

One Backlog Creates

Focus, alignment, and shared outcomes

This doesn't mean everyone works on everything–it means everyone works toward the **same outcomes.**



One Team: Breaking Down Silos

DevOps breaks down the false separation between platform teams and integration teams, build and run, onshore and offshore. High-performing teams transcend these artificial boundaries to deliver better outcomes faster.

Design Together

Collaborative architecture decisions

Build Together

Shared ownership of delivery

Test Together

Quality built in from start

Release Together

Controlled, confident deployment

Support Together

Collective accountability for results

Faster Learning

Knowledge flows freely across boundaries

Fewer Defects

Early collaboration prevents issues

Higher Trust

Shared success builds team cohesion

Automating the Path to Production

Manual deployments and testing introduce risk and slow teams down. DevOps at scale requires automation across the entire lifecycle to ensure repeatability and confidence.

Version-Controlled Configuration

ServiceNow configuration tracked in source control with full change history

Automated Testing

Workflows and integrations validated automatically before promotion

CI/CD Pipelines

Integrations and scripts deployed through repeatable, auditable pipelines

Controlled Promotion

Changes move systematically across dev, test, and production environments

> Automation is not about speed alone–it's about repeatability and confidence.

Observability: Seeing the System as a Whole

In integrated ServiceNow ecosystems, failures rarely happen in isolation. A single incident can cascade across multiple systems, making end-to-end visibility essential for rapid diagnosis and resolution.

End-to-End Visibility

Track workflows and integrations across the entire platform

Flow Metrics

Measure flow, reliability, and quality continuously

Centralized Logging

Unified logging and alerting for faster troubleshooting

Clear Ownership

Defined accountability for incidents and remediation

Observability transforms troubleshooting from guesswork into insight, enabling teams to understand not just what failed, but why—and how to prevent it from happening again.

Governing Without Slowing Down

Governance and DevOps are not opposites. The key is lightweight, outcome-focused guardrails that enable speed while maintaining control.

When governance is aligned to outcomes rather than process for its own sake, teams move faster—not slower. The goal is to build safety and compliance into the flow of work, not layer it on top as an afterthought.

Architectural Standards

Reusable patterns that accelerate development and reduce risk

Change Governance

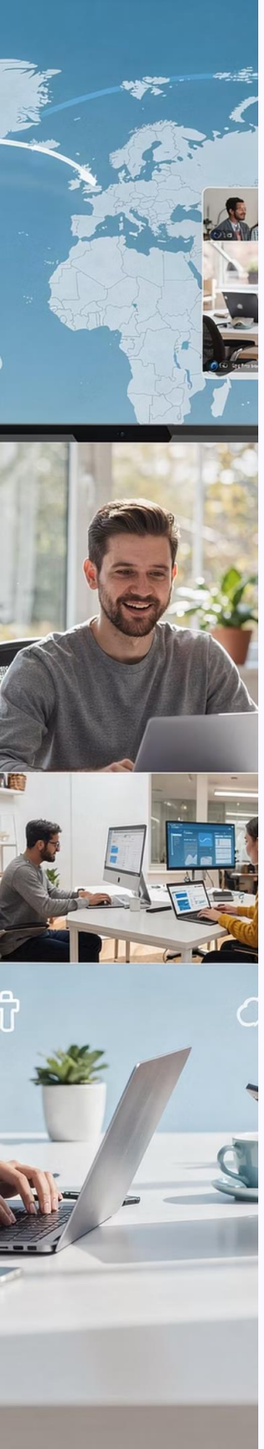
Risk assessment tied to actual service impact, not arbitrary approvals

Built-In Security

Security and compliance checks automated within deployment pipelines

Clear Escalation

Transparent paths for managing exceptions and high-risk changes



Leading Distributed Teams as One System

Scaling DevOps across onshore and offshore teams requires intentional leadership. Geography should not become a barrier to effectiveness–it should be an asset when managed deliberately.

Shared Goals and Metrics

Everyone aligned to the same outcomes and success measures

Clear Communication

Explicit expectations and transparent information sharing

Knowledge Management

Investment in documentation and accessible knowledge bases

Psychological Safety

Environment where accountability and trust coexist

> The best leaders focus less on control and more on **clarity, trust, and enablement.**

Measuring What Matters

DevOps success should be measured in outcomes, not activity. These metrics tell a clear story to executives and teams alike about the real impact of your transformation efforts.

85%

Lead Time Reduction

Time from commit to production deployment

3x

Deployment Frequency

More releases with greater stability

60%

Incident Reduction

Fewer incidents, faster recovery

40%

Cost Improvement

Lower cost-to-serve through automation

Track these metrics consistently over time to demonstrate continuous improvement and build confidence with stakeholders. Good metrics drive the right behaviors and create accountability for outcomes.

The Transformation Journey

Moving from traditional IT operations to scaled DevOps doesn't happen overnight. It requires deliberate steps, sustained commitment, and willingness to learn and adapt.

1. **Assess Current State**
   Understand pain points, bottlenecks, and opportunities across the platform
2. **Build Foundation**
   Establish version control, CI/CD infrastructure, and core automation
3. **Unite Teams**
   Consolidate backlogs and break down organizational silos
4. **Scale Practices**
   Expand automation, observability, and governance frameworks
5. **Optimize & Evolve**
   Continuously improve based on metrics and feedback

Each organization's journey is unique, but the principles remain constant: **automation, collaboration, and continuous improvement.**

Real Results in Action

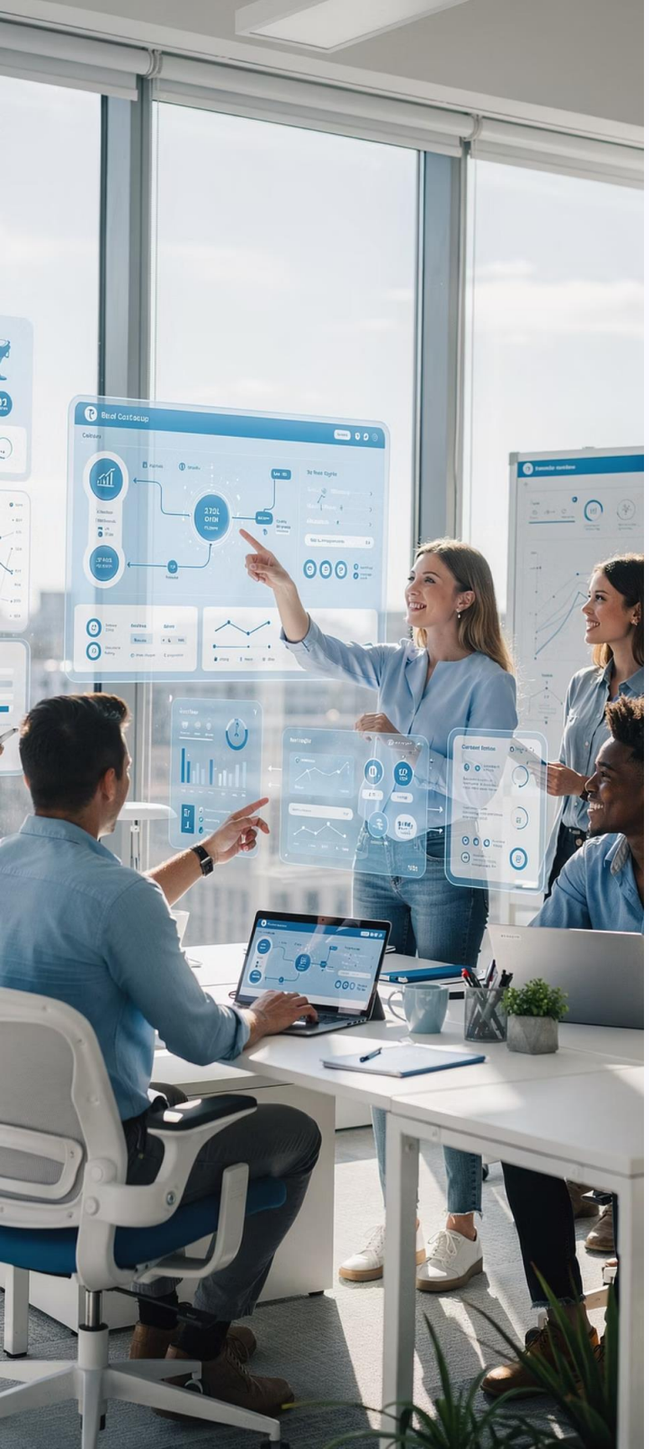
Before DevOps

- Siloed teams with separate backlogs
- Manual deployments taking days
- Frequent production incidents
- Limited visibility across integrations
- Friction between onshore and offshore
- Governance as a bottleneck

After DevOps

- Unified team with single backlog
- Automated deployments in hours
- Proactive issue detection and resolution
- End-to-end observability
- Seamless global collaboration
- Governance enabling velocity

These aren't theoretical benefits–they're real outcomes achieved by organizations that commit to the DevOps transformation journey. The difference is dramatic, measurable, and sustainable.



One Backlog. One Team. Real Results.

DevOps at Scale

DevOps at scale for ServiceNow is not about tools–it's about how teams work together to deliver reliable, integrated outcomes.

One Backlog

Unified priorities drive focused execution and eliminate wasteful handoffs

One Team

Collaborative ownership across the entire platform lifecycle

Clear Ownership

Accountability for outcomes, not just activities or outputs

Measured Impact

Metrics that matter to the business and drive continuous improvement

> That's how real results are achieved.