Mcroabrs

Streamline your Workflow

Request Demo

Collaborate, automate, and
deliver with confidence

Planning

Execution

Reporting

Integrating Microsoft Dynamics 365 with Azure DevOps

Bridge the gap between business operations and software development with this powerful integration that offers a unified view of strategy, execution, and delivery.

by Kimberly Wiethoff

Today's Agenda

Why Integrate?

The business case for connecting these platforms

Key Use Cases

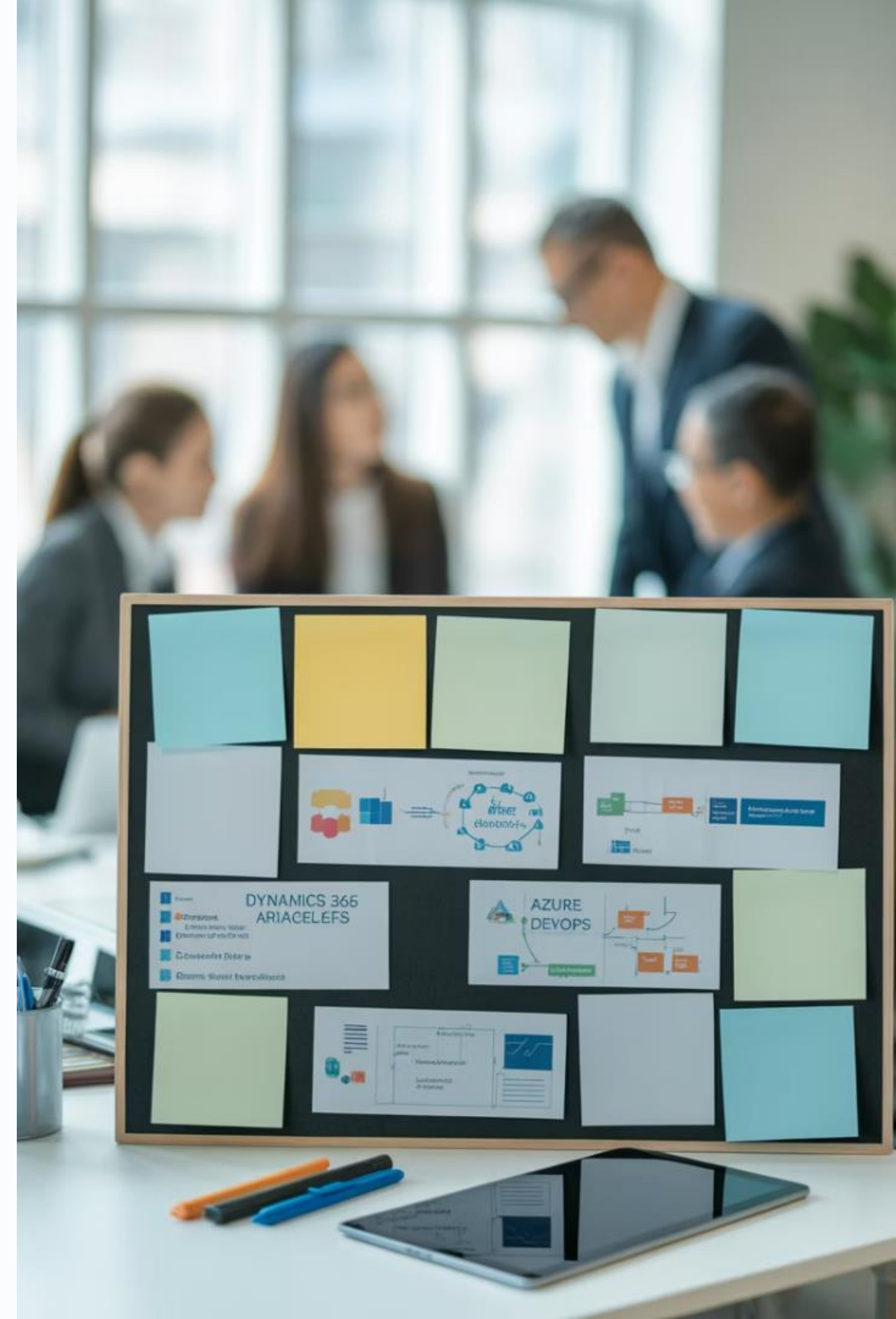Real-world integration scenarios

Integration Methods

Technical approaches to connect systems

Best Practices

Tips for successful implementation

The Integration Challenge

Disconnected Systems

CRM, project tasks, and development tickets exist in separate tools. This creates information silos.

Visibility Gaps

Business requirements and technical implementation become misaligned. Teams lose sight of priorities.

Process Delays

Manual updates across systems cause delays. Reporting becomes time-consuming and error-prone.

Why Integrate These Platforms?

👁 **Full Transparency**

Track progress from customer requirements to deployment in one view.

👤 **Tighter Collaboration**

Connect sales, product, development, and QA teams effectively.
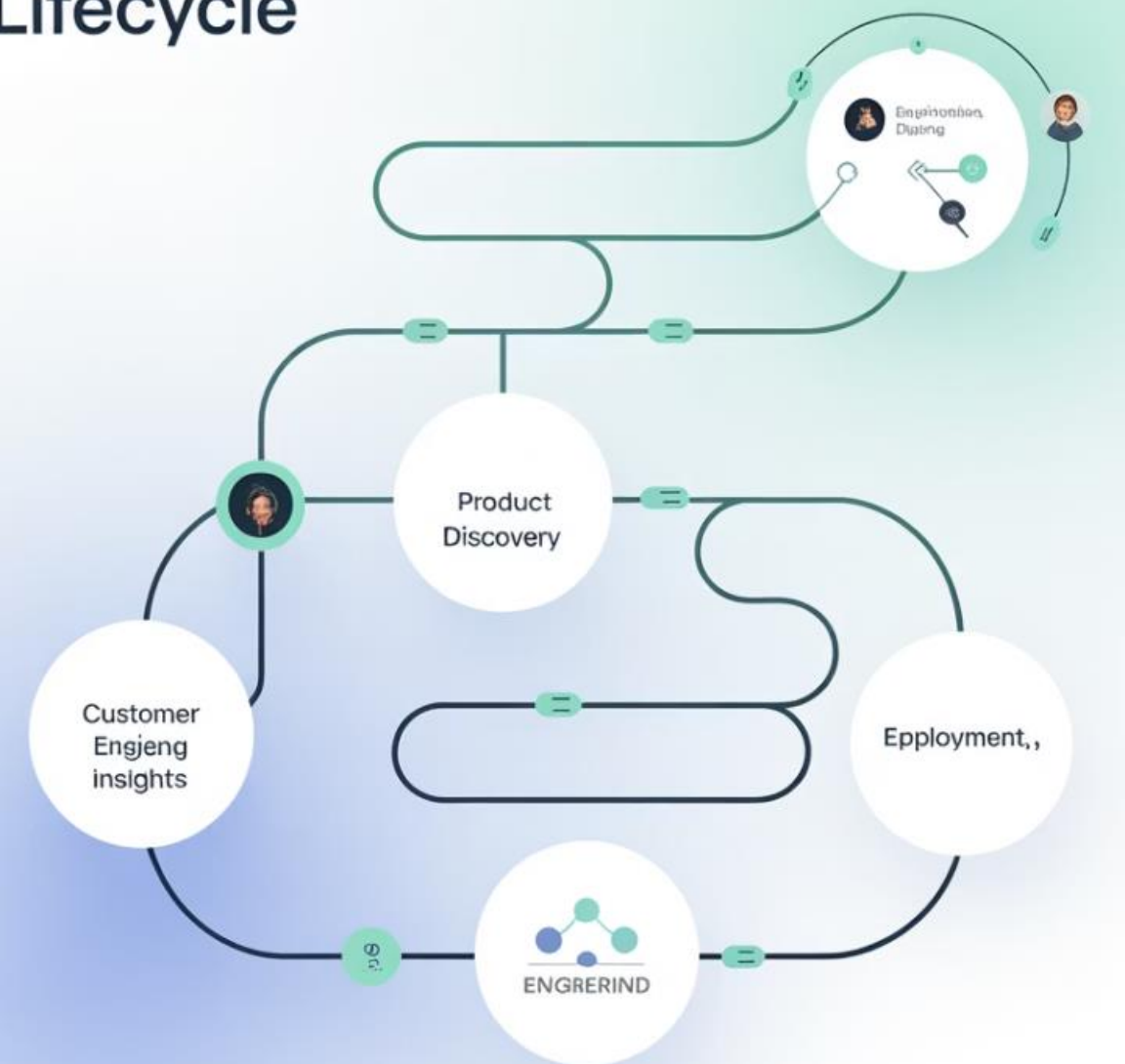
🔄 **Real-time Syncing**

Automatically update project tasks, bugs, and features with CRM data.

Business-Engineering Alignment

Ensure what business wants is what engineering builds.

Traceability from Opportunity to Deployment

1 — **Deal Closed in Dynamics**

Customer opportunity is won in CRM system.

2 — **Feature Creation in ADO**

User stories auto-generated in development backlog.

3 — **Development Progress**

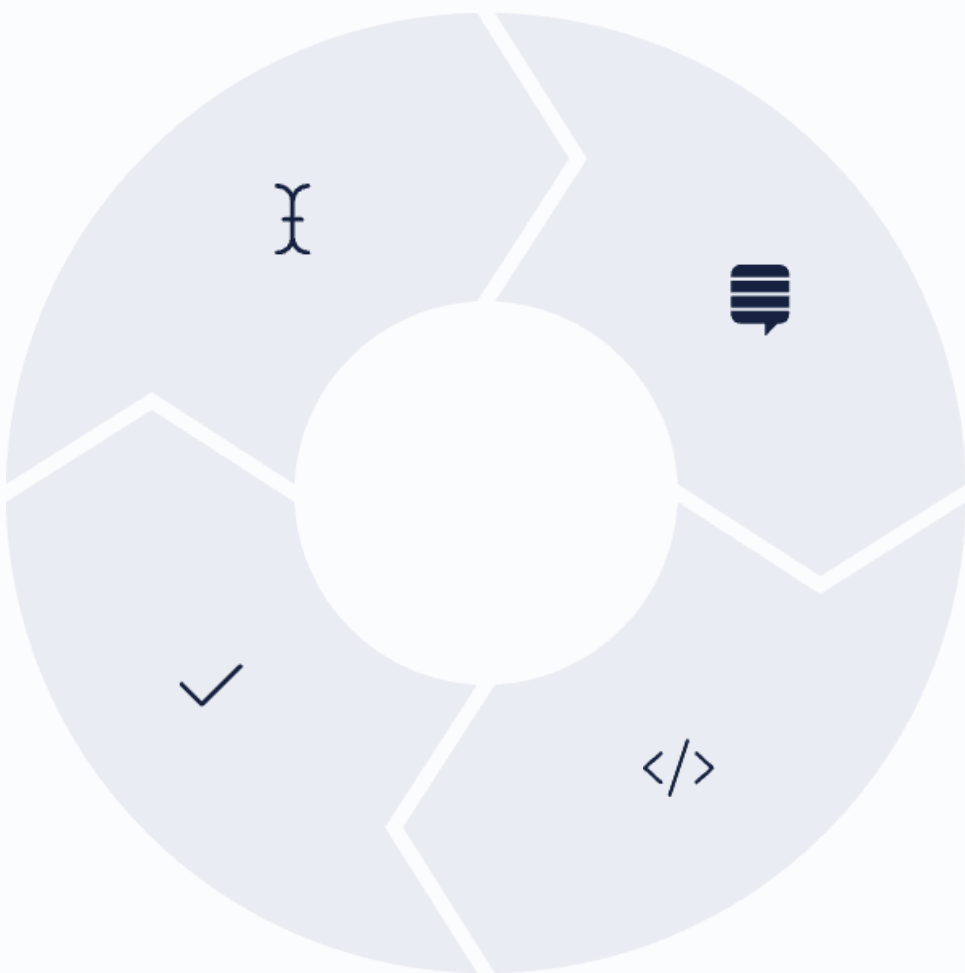Status updates sync back to CRM as work progresses.

4 — **Deployment Notification**

CRM updated when feature goes live.

Syncing Work Items and Project Tasks

Create Project Task

Task created in Dynamics 365 Project Operations

Auto-Sync

Bidirectional sync moves data between systems

Status Updates

Changes in either system reflect in both

Dev Work Item

Corresponding work item appears in ADO

Sprint and Resource Planning Benefits

Before Integration

- Manual capacity checks

- Disconnected planning

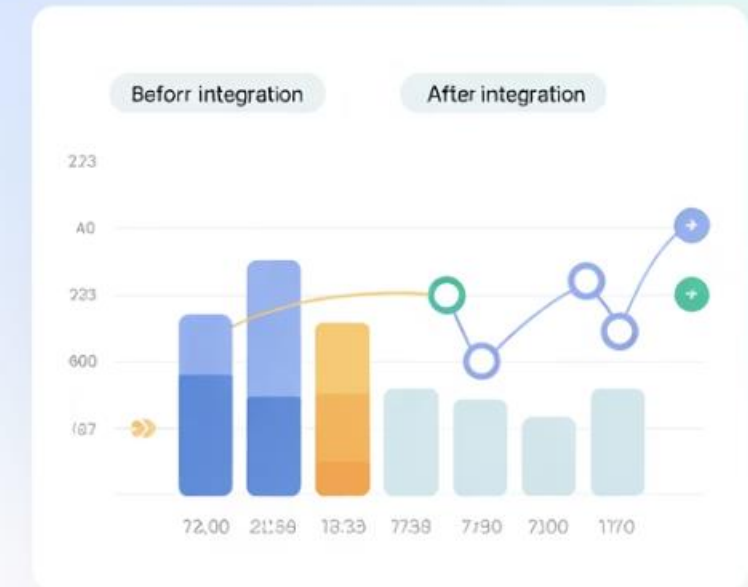- Priority conflicts

- Delayed adjustments

After Integration

- Real-time capacity views

- Business-aligned planning

- Clear priority signals

- Agile timeline adjustments

Enhanced Reporting with Power BI
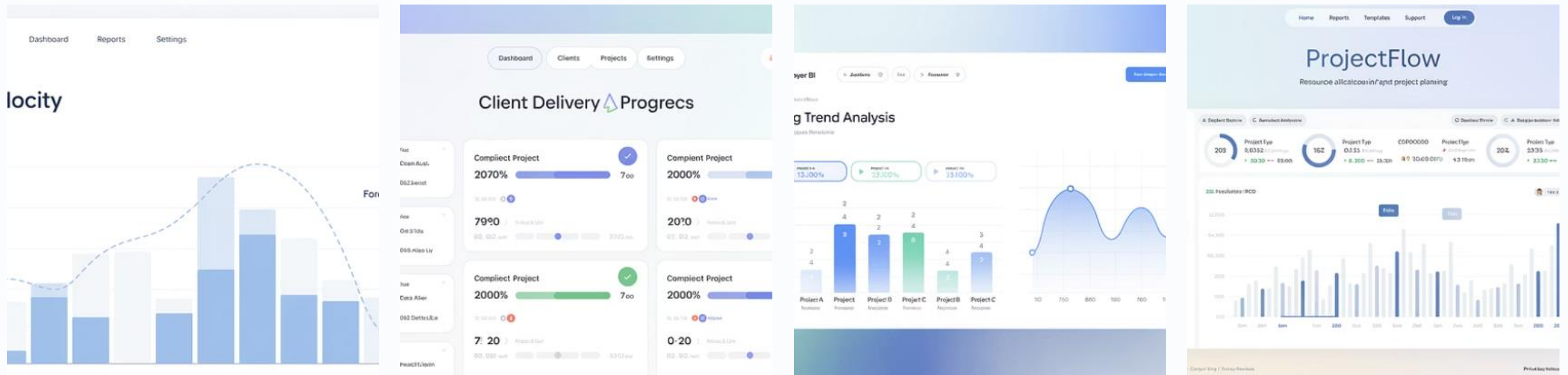
Combine data from both systems to create powerful dashboards. Track sprint velocity, client delivery progress, and bug trends.

Integration Methods

Custom APIs

Tailored integrations for complex requirements

Azure Logic Apps

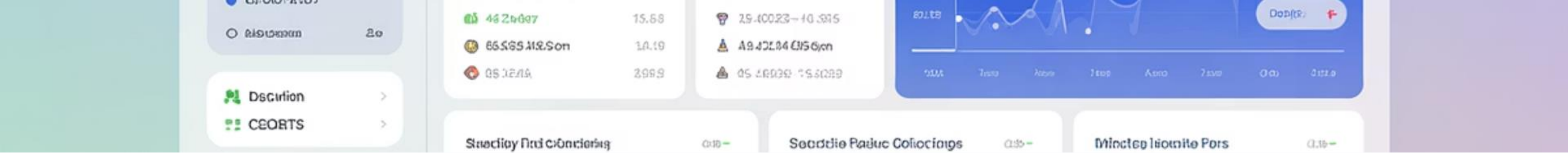Enterprise-grade integration for scalability

Power Automate

No-code/low-code workflows for quick setup
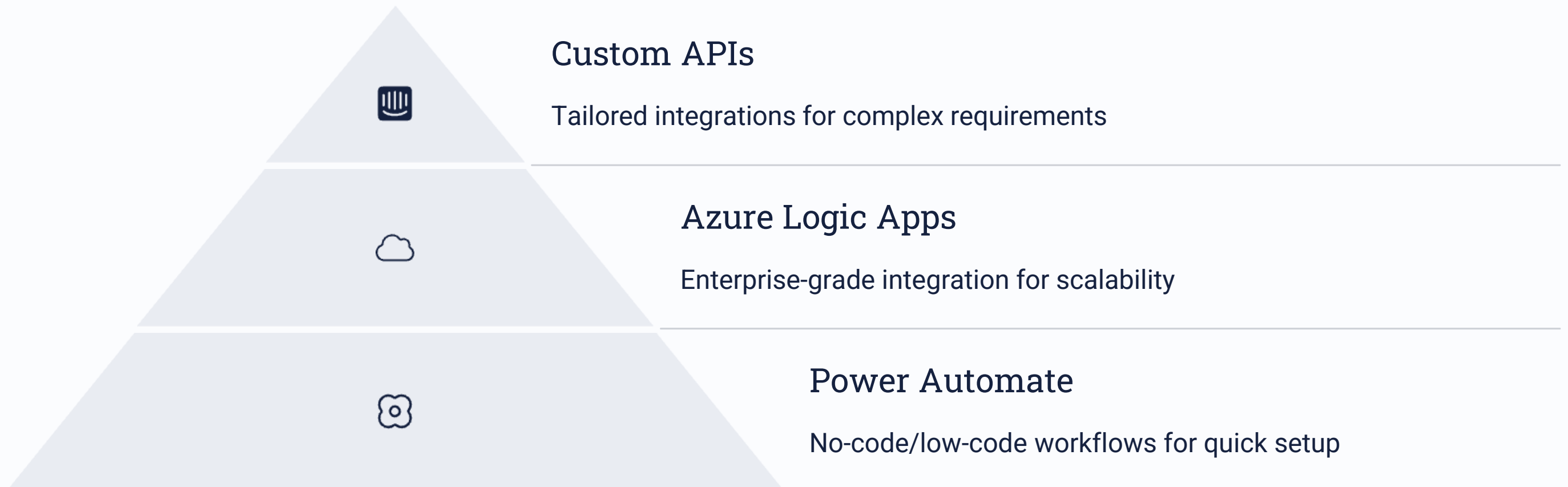
Integration Requirements

⚡ **Real-time Sync**

Ensure data updates happen immediately between systems to prevent delays.

🛡️ **Security Compliance**

Maintain proper role-based access and data protection across platforms.

🔧 **Error Handling**

Implement robust conflict resolution and notification systems for sync issues.

Project Manager Considerations

Define Key Use Cases

Start with high-impact workflows, not everything

Align Stakeholders

Get buy-in from sales, IT, product and QA teams

Plan Change Management

Prepare training and support for new workflows

Implementation Roadmap

Discovery & Planning

Map processes and define integration points. Identify key stakeholders.

Build & Configure

Implement full integration. Set up monitoring and alerts.

Proof of Concept

Test integration with limited scope. Validate sync functionality.

Training & Rollout

Train teams and deploy to production. Monitor adoption.

Measuring Success

30%
Time Savings
Reduced manual updates and reporting efforts

40%
Visibility Improvement
Better cross-team awareness of status

25%
Faster Delivery
Reduced cycle time from requirement to deployment

90%
Alignment
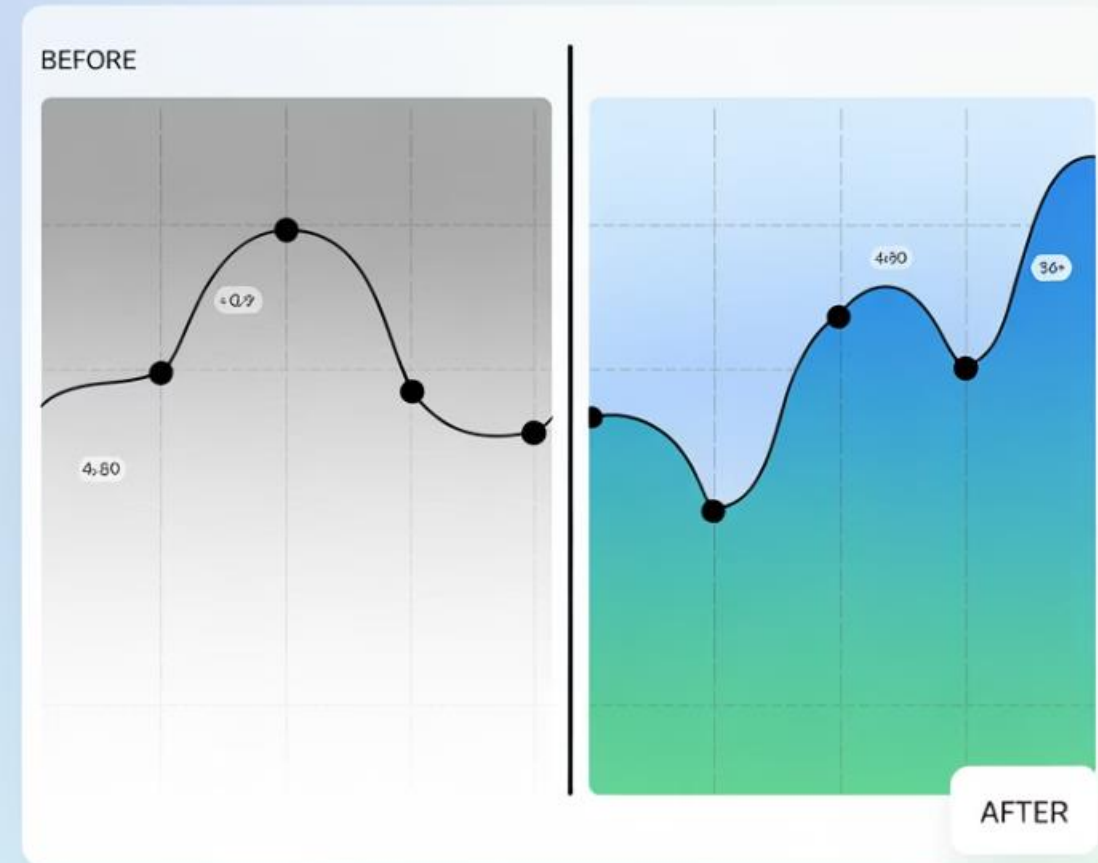Business-development requirement match rate

Key Takeaways

Break Down Silos

Integration connects business operations with technical delivery. This enables true end-to-end visibility.

Start Small, Scale Up

Begin with critical workflows that deliver immediate value. Add complexity as teams adapt to the integration.

Measure the Impact

Track time savings, delivery speed, and alignment metrics. Use data to refine your integration approach.

Final Thoughts

The integration of Microsoft Dynamics 365 with Azure DevOps is a strategic move for IT project managers aiming to break down silos and increase alignment between business and development. By streamlining collaboration, improving traceability, and enhancing reporting, this integration transforms how projects are planned, executed, and delivered.

In a world where agile transformation is no longer optional, this kind of cross-platform connectivity is essential to staying competitive and responsive.