

Pro Dev Practices in a Citizen Dev World: Leveling Up Your Power Platform Game

TONY AMBROSE

M365 TWIN CITIES CONFERENCE

SEPTEMBER 19, 2025

A solid green horizontal bar at the bottom of the slide.



Please thank our sponsors





Tony Ambrose

[linkedin.com/in/ambrosetony](https://www.linkedin.com/in/ambrosetony)

Personal

- Live in Fargo, ND
- Two children, both about to be full-time adults. You've been warned.

Education

- Truman State University
- Computer Science, Mathematics

Career

- Programmer / Analyst → IT Director → CTO → Consultant
- Small Business Owner
- Adjunct Trainer, NDSCS

Hobbies

- Avid Indoorsman
- Sports Nut
- Foodie / Amateur Chef
- Enjoy Hiking and Biking (in reasonably small doses)

Community

- President, Technology Council of North Dakota (TechND)
- Knights of Columbus

10: Why Stuff Breaks and
What We Can Do About It

20: The Application Lifecycle

30: Storing Our Stuff

40: Managing Our Work

50: Calls To Action / Q&A

Agenda

Why Stuff Breaks And What We Can Do About It

0

10

20

30

40

50

You (Yes, You)
Are A Developer



Personal
Productivity
Solutions



Group /
Team
Solutions



Enterprise
Solutions

Citizen Developers

Pro Developers

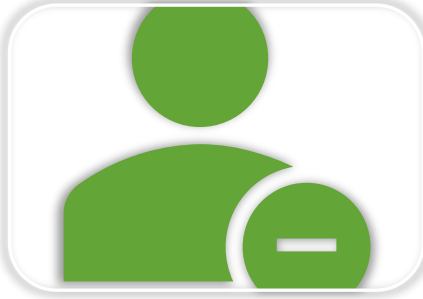


The Citizen Developer

A **citizen developer** is typically a business user who creates applications, workflows, or data solutions using **low-code or no-code platforms** like Microsoft Power Apps, Power Automate, or Power BI.

Citizen developers help organizations **innovate** faster, **reduce** IT backlog, and **empower** teams to **build** solutions that meet their **immediate needs**.

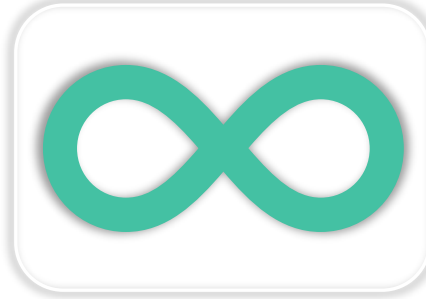
Why Great Solutions Break



The **Lottery** Factor



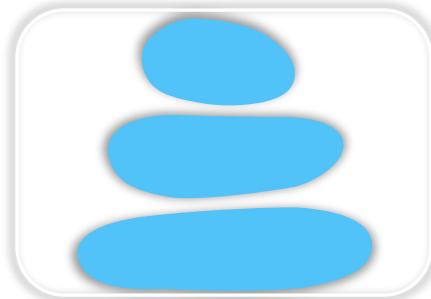
The **Gray Matter**
Requirements Storage
Factor



The **Long-Term Short-**
Term Solution



The **It Just Went Viral**
Solution



The **Can It Also Do This**
Solution



The **Busted Shadow IT**
Solution



The **Not-So-Secure**
Secure Solution

Pro-Dev Principles Starter Kit

Solutions are **Products**, not One-Off Projects

Name Everything Something

Version Everything

Document Everything

- Business Requirements
- Configurations
- That One Cool Thing You Did That Was Super Clever

Use Purpose-Driven **Environments**

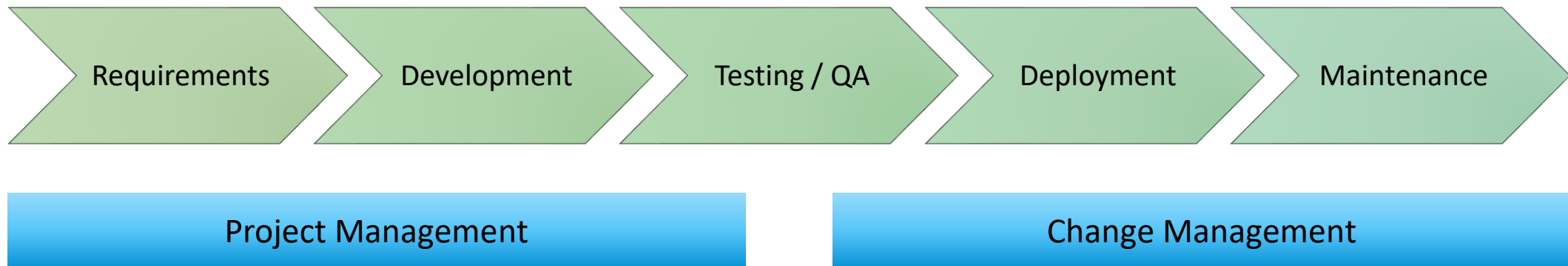
- Development -> Test -> Production

Embrace **Standardization**, It Enables **Automation**

The Application Lifecycle

0 10 20 30 40 50

Application Lifecycle Management



Application Lifecycle Considerations



Evolving Requirements



Enhancements



Operations vs Maintenance



Solution Version Control



Asset / Artifact Management



Environment Management

Application Lifecycle Management

POWER PLATFORM

POWER BI / FABRIC



- Create Separate Environments for Sandbox and Production
- Use Solutions
- Develop Ownership Strategy
- Create Naming Convention

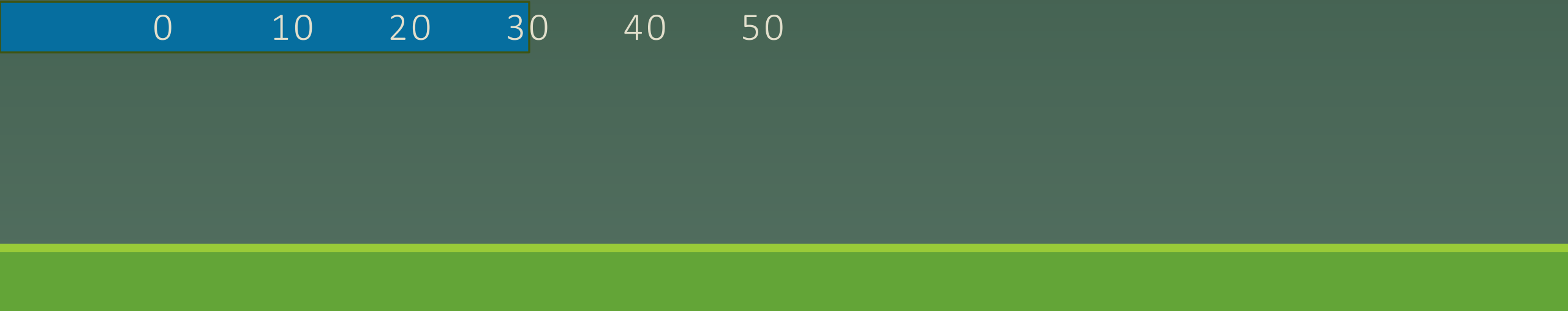
- Create Separate Workspaces for Development and Production
- Develop Ownership Strategy
- Create Naming Convention
- Parameterize Data Sources in Power Query

- Create and Use Development, Test, Production Environments
- Use Pipelines
- Leverage the [Microsoft Power Platform Center of Excellence Starter Kit](#)

- Create and Use Development, Test, Production Workspaces
- Separate Semantic Models from Reports
- Use Deployment Pipelines
- Endorse / Certify “Official” Reports and Datasets

Storing Our Stuff

0 10 20 30 40 50

A horizontal bar at the bottom of the slide is divided into two segments. The segment from 0 to 30 is blue, and the segment from 30 to 50 is green. The numbers 0, 10, 20, 30, 40, and 50 are positioned above the bar, corresponding to the scale.



Imagine This Scenario...

Wait A Minute...



M365 List / Library Templates



Power App / Power Automate
Templates



Power Platform Solution Files



Power BI Semantic Model / Report /
Dataflow Files



Documentation

Managing Development Artifacts

How do we manage our code?

What about Passwords / Accounts / Secrets?

Configuration Documentation?

Deployment Documentation?

Well-Organized File Server with Daily Backup



Well-Organized SharePoint Library with Version Control Enabled



Managed Source Code Repository (e.g., Git, Mercurial, etc....)



Managed Source Code Repository With Platform Integration



Managing Our Work

0 10 20 30 40 50

A horizontal bar chart is displayed below the title. It features a scale from 0 to 50 with major tick marks at intervals of 10. A solid blue bar starts at the 0 mark and extends to the 40 mark. The bar is outlined in a thin dark blue line. The numbers 0, 10, 20, 30, 40, and 50 are printed in white below the scale line.

Work Management Considerations



Business Requirement Documentation

What does the solution need to do?



Work Prioritization / Planning

What gets done first and who should do it?



Work Execution Tracking

What's done and what's left to do?



Technical Documentation

How does this solution work?



Stakeholder Communication

Who needs to know what about the solution?



- Capture Business Requirements in Email / Chat
- Track Tasks in Personal ToDo or Excel
- Ad-Hoc Prioritization and Scheduling
- Informal Communication To Stakeholders Using Teams



- Business Requirements in OneNote / SharePoint
- Task Tracking in Planner / Shared ToDo
- Assigned Task Owners / Due Dates
- Regular Check-Ins with Stakeholders and Structured Communication



- Azure DevOps Boards / Repositories / User Stories for Work Definition, Tracking, and Documentation
- Basic Prioritization
- Link Work Items to Code for Traceability



- Agile / Scrum (Sprints, Stand-Ups, Retrospectives) for Planning
- Automate Deployment
- Technical Documentation in the Repository

Managing Work

In Closing...

0 10 20 30 40 50

Calls To Action

*Do what you can,
with what you've got,
where you are.*

--Theodore Roosevelt

**Help
Future You**

**Document
Stuff**

Name Stuff

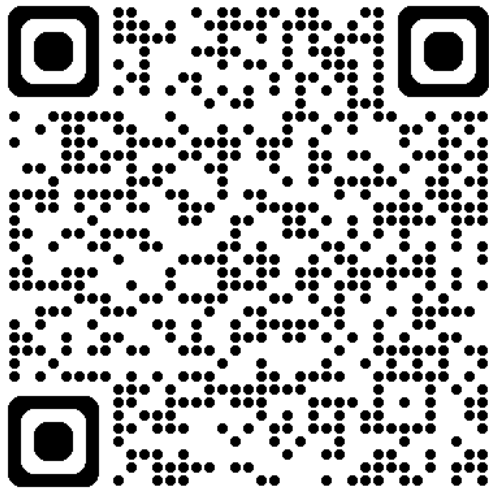
Track Stuff

**Think Long
Term**

**Think Big
Picture**



Questions / Answers



Thank You!

Tony Ambrose

tony@ambrose.solutions

<https://www.linkedin.com/in/ambrosetony>



Please Submit
Session Feedback
