Slide 1 — Welcome and Intent

Hello everyone, and thank you for being here today. My name is Robert Crane. This session is called “Effectively Managing Microsoft 365 Environments with Microsoft Tools.” It’s a technical, not a sales, session for SMB MSPs and IT professionals who need practical, repeatable ways to run Microsoft 365 with confidence.

By the end of this talk, you’ll have a playbook and working patterns you can take home and apply this week. We’ll lean on the tools most of us already have: Microsoft 365 Admin Center, PowerShell, Microsoft Graph, Microsoft Defender, Intune, and Copilot. I’ll show you live demonstrations, share scripts and prompts, and keep everything lean, reliable, and repeatable—because in SMB, time and margins matter.

[PAUSE]

[CLICK]

Slide 2 — Why Effective Management Matters

Let’s start with the why. In an SMB, the person who’s fixing the printer might also be the person who’s defending identities and data. If management is ad-hoc, little issues become outages, engineers burn time on repeatable manual work, and customers lose trust.

Effective management reduces noise, hardens security by default, and gives your engineers time back. It standardises how your team operates so that every engineer can deliver the same quality under pressure. The goal is simple: make routine work boring and make hard work doable.

[CLICK]

Slide 3 — The Playbook: People, Process, Platform

Here’s the playbook that anchors everything: People, Process, Platform.

People: Set expectations. Every engineer follows the same baseline steps for onboarding, incident triage, and maintenance.

Process: Define what “good” looks like—what we check daily, weekly, and monthly; when we pilot a change; and how we capture lessons learned.

Platform: Use the tools built into Microsoft 365. The Admin Center for visibility. PowerShell and Microsoft Graph for automation. Defender for detection and response. Intune for policy and control. And Copilot to summarise, correlate, and accelerate the work.

[CLICK]

Slide 4 — Tool Map

A quick map of where we're going.

Microsoft 365 Admin Center: Your single pane for health, service health, message centre changes, and reports.

PowerShell: Your force multiplier for safe, repeatable operations.

Microsoft Graph: The unified API across Microsoft 365—identity, devices, SharePoint, Teams, Intune, Defender, and more.

Microsoft Defender XDR: From alert to incident to outcome, with advanced hunting via KQL.

Intune: Baseline configuration, compliance, updates, and app deployment.

Copilot: A fast drafting and analysis partner—great for summaries, KQL suggestions, and first-pass scripts you then review.

[CLICK]

Slide 5 — Admin Center: Visibility That Drives Action

The Admin Center is your heartbeat monitor. It answers two daily questions: "Is anything broken?" and "What needs my attention?"

We'll use Health and Service Health to catch outages quickly, Message Center to understand upcoming changes, and Reports to track adoption and posture. We'll convert insights into action items, and where possible, into automation so tomorrow is easier than today.

[CLICK]

Live Demo 1 — Admin Center Health and Reports

[DEMO — switch to your browser and open Microsoft 365 Admin Center]

Narration: I'm in the Microsoft 365 Admin Center. On the left, I open Health, then Service health. I scan for any active incidents. If there's a degradation affecting, say, Exchange Online, I capture the incident ID and link it to our customer ticket so everyone sees status without re-explaining it all day.

Next, I go to Reports → Usage. I'm looking for unusual drops in activity or adoption changes that might connect to an incident or a rollout. If Message Center shows an authentication change or something likely to impact devices, I tag it for review and add it to our weekly maintenance checklist.

If the portal is slow or unresponsive: If the portal stalls, the process is the same: check Service health, correlate to user impact, and record a quick customer-friendly note. The key is consistency and clear communication when things wobble.

[CLICK]

Slide 6 — PowerShell: Your Force Multiplier

PowerShell turns one engineer into many. The goal is consistency and safety: run once, run reliably, and document results.

For Microsoft 365, you'll routinely connect to Exchange Online and to Entra ID using Microsoft Graph PowerShell. Use least-privilege admin roles and consider workload-specific permissions. The pattern is always the same: connect, query small, validate, scale, and log. If a script isn't safe to run twice, it isn't ready.

[CLICK]

Live Demo 2 — Quick Wins with PowerShell

[DEMO — switch to your terminal]

Narration: I'll start by connecting to Exchange Online.

Say: Connect-ExchangeOnline

Say: Get-EXOMailbox -ResultSize 5 | Select DisplayName, PrimarySmtpAddress

Say: Get-EXOMailbox -ResultSize Unlimited | Where-Object { $_.LitigationHoldEnabled -eq $true } | Measure

Say: Connect-MgGraph -Scopes User.Read.All, Group.Read.All

Say: Get-MgUser -Top 5 | Select-Object DisplayName, UserPrincipalName, AccountEnabled

Explanation: Start small, verify output, then expand. Save the script, tag it with a version and a purpose, and store it in source control. This makes the fix repeatable by any engineer on the team.

[CLICK]

Slide 7 — Microsoft Graph: The Unified API

Microsoft Graph is your single, coherent API for Microsoft 365. It covers users, groups, devices, Teams, SharePoint, Intune, Defender signals, and more. Use Graph PowerShell for quick wins, and REST when you need precise control or automation pipelines.

Permissions matter. Request only what you need. Prefer app-based authentication for scheduled automation and use delegated auth when you're doing one-off admin tasks interactively.

[CLICK]

Live Demo 3 — Graph in Practice

[DEMO — Option A: Graph PowerShell (recommended)]

Say: Connect-MgGraph -Scopes Device.Read.All

Say: Get-MgDevice -Top 5 | Select DisplayName, OperatingSystem, TrustType

[DEMO — Option B: REST with an already acquired token]

Say: Invoke-RestMethod -Method GET -Uri "https://graph.microsoft.com/v1.0/users?$select=displayName,mail,accountEnabled&$top=5" -Headers @{ Authorization = "Bearer <token>" }

Explanation: The same approach works across workloads. Start small, validate, expand, and then automate. If you can describe it as a repeatable outcome, it belongs in a script or workflow.

[CLICK]

Slide 8 — Microsoft Defender XDR: From Alert to Outcome

Defender provides detection and response across identities, endpoints, email, and applications. In SMB, the objective is to triage quickly, reduce noise, and capture learning in playbooks so next time is faster.

We'll look at Incidents, then jump into Advanced Hunting with KQL—for example, spotting suspicious encoded PowerShell or investigating unusual sign-ins.

[CLICK]

Live Demo 4 — Triage and KQL

[DEMO — switch to Microsoft Defender portal]

Narration: I open Incidents and filter to Active. I choose one with medium severity. I scan the entities—user, device, mailbox—and read the alert story to understand sequence and scope. I check what automated investigation has already done and whether there are pending actions.

Now I'll move to Advanced Hunting for a quick hunt. Here's a small query to spot encoded PowerShell:

Say: DeviceProcessEvents | where ProcessCommandLine has "-enc" | top 10 by Timestamp desc

Explanation: With Copilot in Defender, I can also ask: "Summarise the risk on this incident and recommend next steps." I review the draft, tighten it, and then capture those steps in our runbook.

[CLICK]

Slide 9 — Intune: Policy as Code, Consistency by Default

Intune enforces your standards: compliance, configuration, updates, and apps. In SMB, keep it simple and opinionated: a secure baseline, device compliance with Conditional Access,

update rings, and a small set of configuration profiles. Use filters to target by device attributes, and remediation scripts to correct drift.

[CLICK]

Live Demo 5 — Build a Baseline

[DEMO — switch to Intune admin centre]

Narration: I'll create a device compliance policy for Windows. I set BitLocker required, Secure Boot required, and a minimum OS version. I assign it to All Windows devices with a filter to exclude lab machines.

Next, I create an update ring with deadlines and grace periods aligned to business hours, so updates land predictably without surprising users.

Then I add a configuration profile to enforce OneDrive Known Folder Move and Defender SmartScreen. I assign to a pilot group first, validate results, then expand.

Explanation: Every setting should map to a customer outcome: fewer incidents, smoother onboarding, or stronger protection. If you can't tie a setting to an outcome, reconsider it.

[CLICK]

Slide 10 — Copilot: Acceleration, Not Autopilot

Copilot is an accelerator, not an autopilot. It's excellent for summaries, first drafts, KQL suggestions, and starting points for scripts. Treat Copilot like a fast junior engineer: great instincts, but always review and verify before you adopt its output.

[CLICK]

Live Demo 6 — Copilot for Admins and Security

[DEMO — scenario, with your preferred Copilot surface]

Say: Copilot, summarise the last 24 hours of Defender incidents that involve email attachments and propose a customer update in plain English, around 120 words.

Say: Copilot, propose a KQL query to find PowerShell launched by Office processes where the command line includes encoded arguments. Please explain each clause.

Explanation: I take the draft, test it, and commit the final version into our runbook and scripts repository so the whole team benefits.

[CLICK]

Slide 11 — Best Practices You Can Adopt This Week

Here are practical best practices you can implement immediately.

One: Standardise connections in PowerShell. Create a small module or script that connects to Exchange Online and Graph with the right scopes every time.

Two: Version your scripts. Tag them with purpose, inputs, and outputs, and store them in source control.

Three: Enforce least privilege via RBAC and separate admin accounts. Elevate only when needed.

Four: Turn recurring checks into scheduled jobs—use Graph or PowerShell to email a weekly health report.

Five: Define a 30-minute incident triage routine: check Service health, scan Defender incidents, review Intune failures, and record actions in a single place your team can see.

[CLICK]

Slide 12 — Monitoring and Reporting

Build lightweight reporting your customers understand. Automate a weekly summary that captures: incidents handled, device compliance, risky sign-ins resolved, and upcoming changes from Message Center. Keep it simple, make it visual where it helps, and always include recommended next steps.

Internally, track Mean Time To Detect and Mean Time To Resolve. Over time you should see the benefits of standardisation—less variance, fewer surprises, and clearer accountability.

[CLICK]

Slide 13 — Common Pitfalls and How to Avoid Them

Here are frequent pitfalls and fixes.

Pitfall one: Running as Global Admin for everything. Fix: Create workload-specific roles and elevate only when needed.

Pitfall two: Scripts without error handling. Fix: Add try-catch, timeouts, and logging to a central location.

Pitfall three: Portal-only changes. Fix: Document and then automate the change so it’s repeatable and reviewable.

Pitfall four: Too many policies at once. Fix: Pilot first, measure impact, then roll out.

[CLICK]

Slide 14 — What’s Next

Here’s how to start this week.

Choose one repeatable task—for example, reporting on mailbox sizes, device compliance, or inactive accounts. Automate it with PowerShell or Graph.

Create a shared runbook page for your team. Paste in the demos and notes from today.

Then schedule a monthly improvement sprint: one automation, one policy improvement, and one report enhancement. Small, consistent gains compound quickly.

[CLICK]

Slide 15 — Call to Action

If you take one thing away, let it be this: standardise the routine, so you can focus on the exceptional. Your customers feel that reliability, and your team gets time back to do its best work.

I'll share a starter pack of scripts and queries that you can adapt. Use them, change them, make them yours.

[CLICK]

Slide 16 — Q&A

I'd love to hear your questions. If it's specific to your tenant or a customer, please give me the pattern rather than sensitive details, and we'll reason through the approach together. After the session, I'm happy to dive deeper.

[PAUSE]

Appendix — Speaking Cues, Commands, and Fallbacks

Pacing: Keep a clear, measured delivery. Use the [PAUSE] marks to let points land, especially before starting a demo or after a major conclusion.

Demo fallback if anything wobbles: "The exact clicks don't matter—the process does. I'll narrate the steps and share the script afterwards so you can reproduce it."

PowerShell call-outs to say naturally (not verbatim code-read): Connect to Exchange Online. List five mailboxes with display name and email address. Count how many mailboxes have Litigation Hold enabled. Connect to Microsoft Graph with user and group read scopes. List five users with display name, UPN, and whether the account is enabled.

KQL call-out to say naturally: "Show device process events where the command line includes the dash e n c switch, sorted by newest first."

Intune baseline talking points: Compliance requires BitLocker and Secure Boot. Update rings with clear deadlines and grace periods. Configuration profile to enforce OneDrive Known Folder Move and Defender SmartScreen. Pilot first, then expand.

Copilot prompts you can read verbatim: “Summarise the last 24 hours of Defender incidents involving email attachments and draft a 120-word customer update.” “Propose a KQL query to find encoded PowerShell launched by Office processes and explain each clause.”

Optional Slide Timing Guide

Slides 1–4 (Framing & Tool Map): ~8 minutes

Slides 5 & Demo 1 (Admin Center): ~6 minutes

Slides 6 & Demo 2 (PowerShell): ~6 minutes

Slides 7 & Demo 3 (Graph): ~6 minutes

Slides 8 & Demo 4 (Defender): ~6 minutes

Slides 9 & Demo 5 (Intune): ~6 minutes

Slide 10 & Demo 6 (Copilot): ~5 minutes

Slides 11–15 (Practices, Monitoring, Next): ~6 minutes

Slide 16 (Q&A): remainder