

AI for Teams - Smart Adoption Not Blind FOMO



John Crosby

Principal Engineer, Client Engineering @ MLB

✉️ john@hellocrosby.com

bsky: [@jccrosby.com](https://bsky.app/@jccrosby)



Navigating AI for Teams

Smart Adoption vs. Blind
FOMO

The Current Landscape



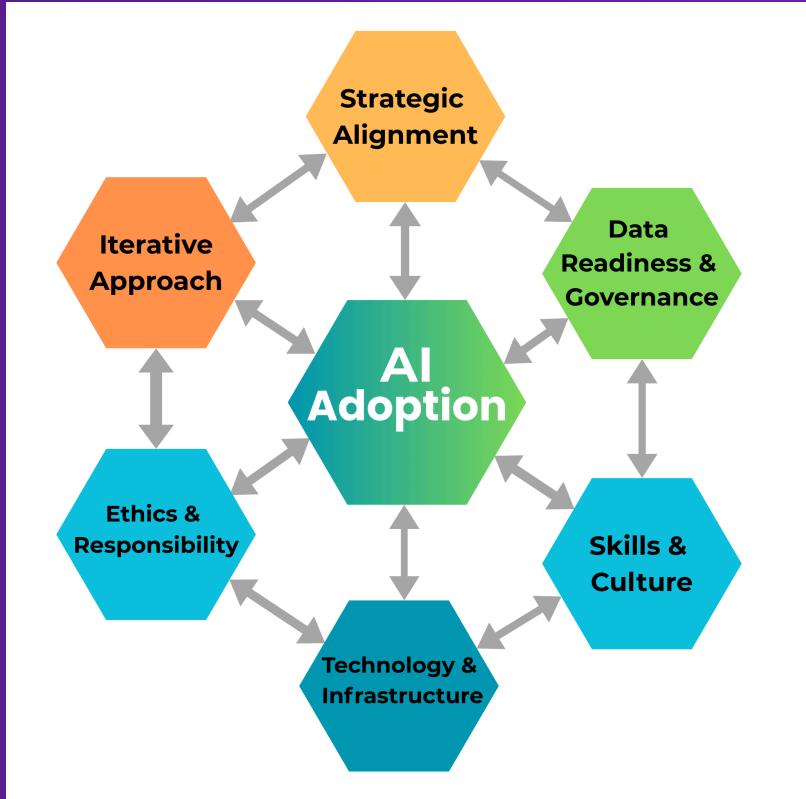
The Dual Edge: Hype vs. Opportunity

The pressure to "do something with AI" can lead to rushed decisions & poor outcomes

Smart Adoption vs. Blind FOMO

The Core Difference

The implementation of
AI matters more than its
mere adoption.

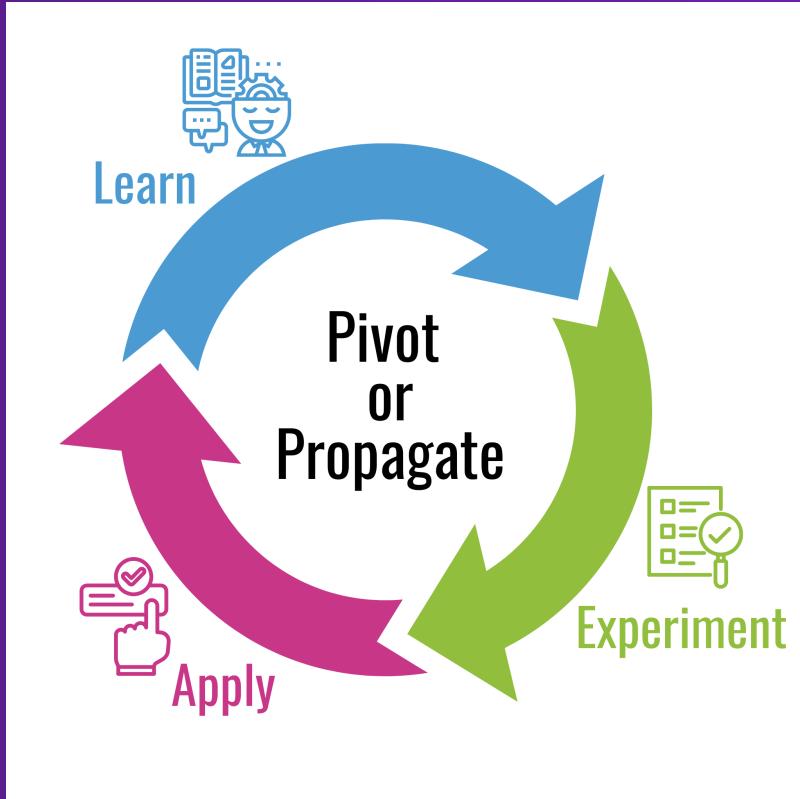


Core Principles of Effective AI Adoption

Foundational Principles to Smart Adoption

Six Core Principles

1. **Strategic Alignment:** Define *why* before *what*.
2. **Data Readiness & Governance:** Poor data leads to failure.
3. **Skills & Culture:** Learning, experimentation, critical assessment, and sharing are key.
4. **Technology & Infrastructure:** The right tools in the right place.
5. **Ethics & Responsibility:** Fairness, transparency, privacy, security
6. **Iterative Approach:** Short feedback loops and continuous learning.



The L.E.A.P. Framework

An Iterative Process for Smart Adoption

1. LEARN

Understand AI capabilities and limitations

Activities

- Research AI tools and advancements.
- Identify pain points in your process.
- Assess team skills.
- Evaluate data availability and quality.
- Consider ethical and security implications.
- Define **SMART** goals for experiments.

Output

- Prioritized use cases.
- Readiness assessment.
- Defined success metrics.

2. EXPERIMENT

Focus on small-scale, controlled pilots.

Activities

- Select a low-risk project experiments.
- Choose specific AI tool/model.
- Develop and execute a prototype or limited integration.
- Gather quantitative and qualitative feedback.

Output

- Working prototype.
- Performance data.
- User feedback.
- Lessons learned.

3. ASSESS

Critically evaluate outcomes to make better decisions.

Activities

- Evaluate results against your goals.
- Analyze the benefits vs. costs
- Assess impact on workflow, quality, productivity, satisfaction.
- Identify challenges and risks.
- Review ethical and security concerns.
- Make a clear **Go/No-Go** decision.

Output

- A more refined understanding of requirements and the results.
- An assessment report.
- A clear decision.

PIVOT

PROPAGATE

4. PIVOT

If the experiment fails or needs adjustment

Activities

- Analyze failure points and gather feedback.
- Adjust goals, tools, or methods based on insights.
- Reassess readiness and risks.
- Iterate on the experiment with a new focus or move to another experiment.

Output

- Revised experiment plan.
- Updated goals.
- A new approach to testing.

PROPAGATE

Strategically scale successful AI applications.

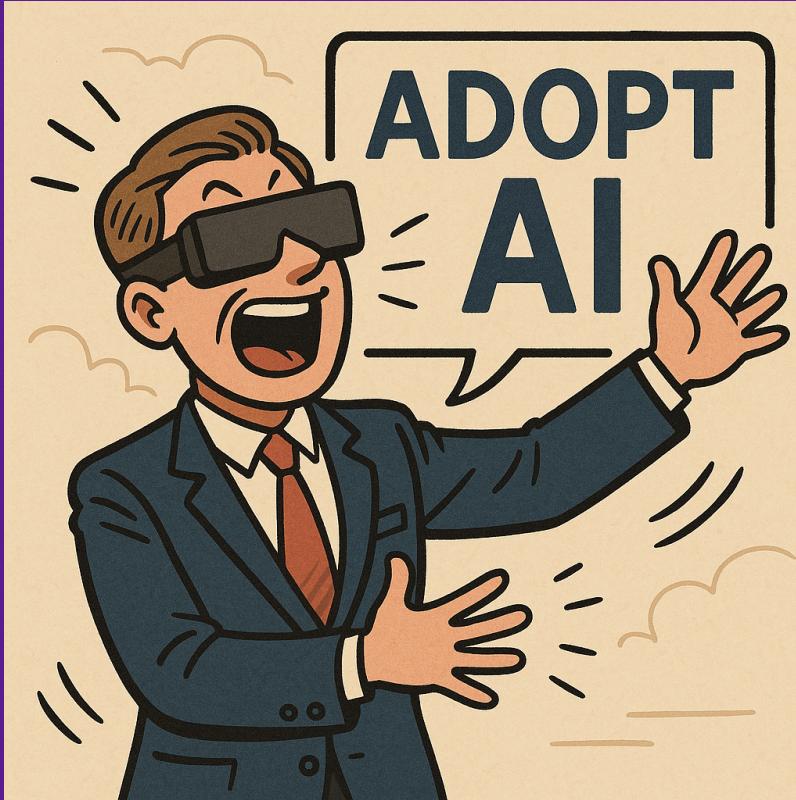
Activities

- Develop a detailed rollout roadmap.
- Identify infrastructure and tooling changes.
- Create and execute training programs.
- Establish governance policies and monitoring.

Output

- A useable AI solution.
- Maintenance & governance structures.
- Ongoing monitoring plan.
- Up-skilled team.

Share what you've learned from
the successes and failures.

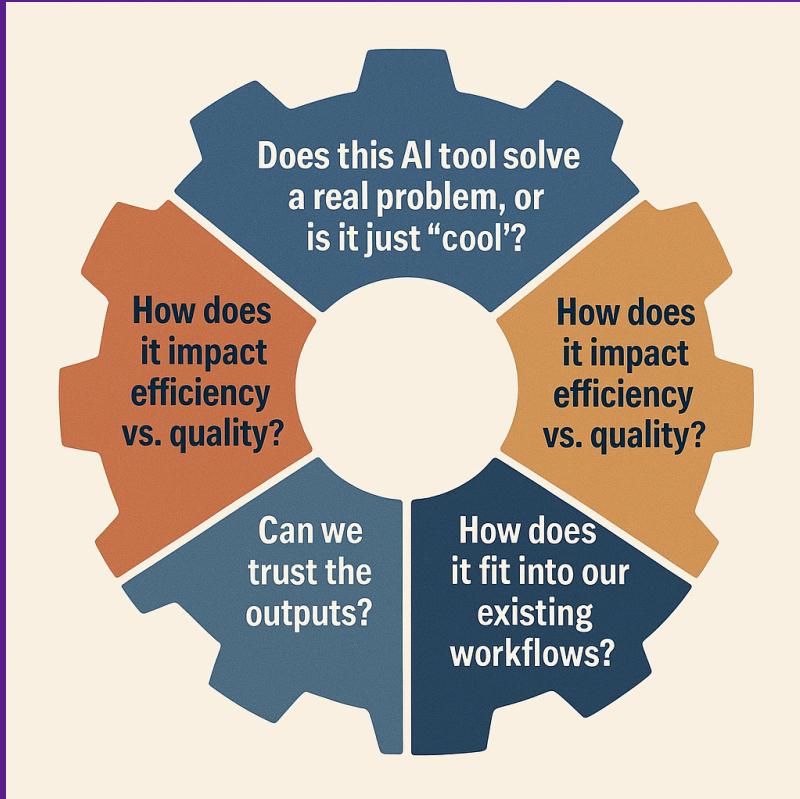


Story time: Blind FOMO

Cursor: AI Tooling For
Coding

The Cursor "Experiment"

- **Skipped Learn:** Vague goal, focused on tech over specific problems, and ignored readiness
- **Skipped Experiment/Assess:** We had no clear goals, no way to measure success, and no understanding of the risks
- **Failed Pivot/Propagate:** Resulted in chaos, inconsistent usage, wasted resources and time, potential security and IP issues, no way to measure impact

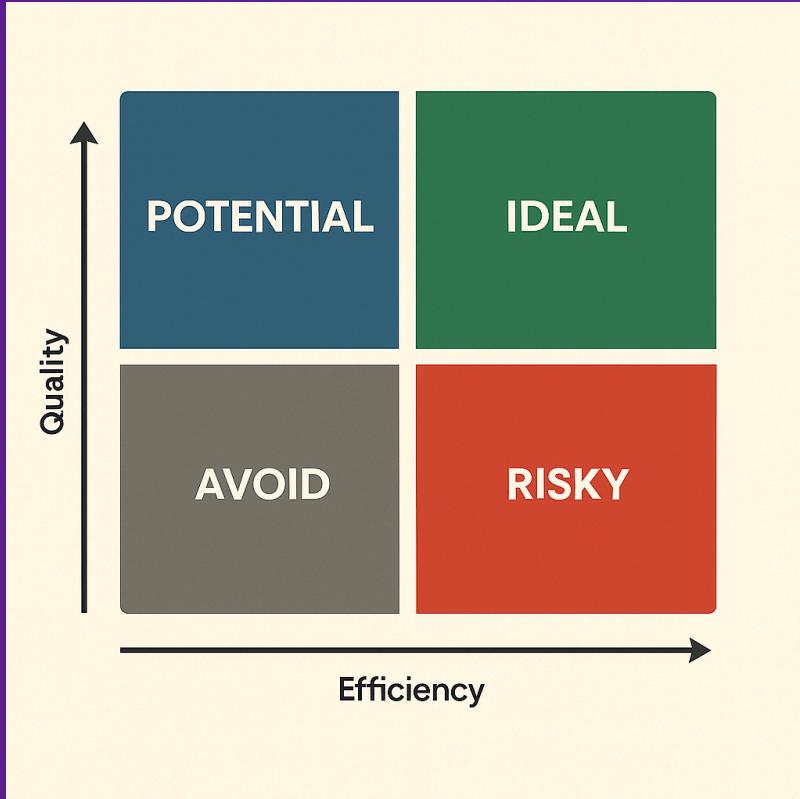


Smart AI Adoption: How to Evaluate AI for Your Team

Five Questions to Ask

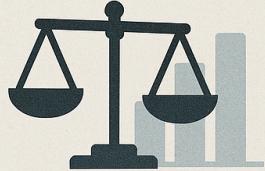


Question 1:
**Does it solve a
real problem, or
is it just "cool"?**



Question 2:
How does it
impact efficiency
vs. quality?

RISKS?



Bias

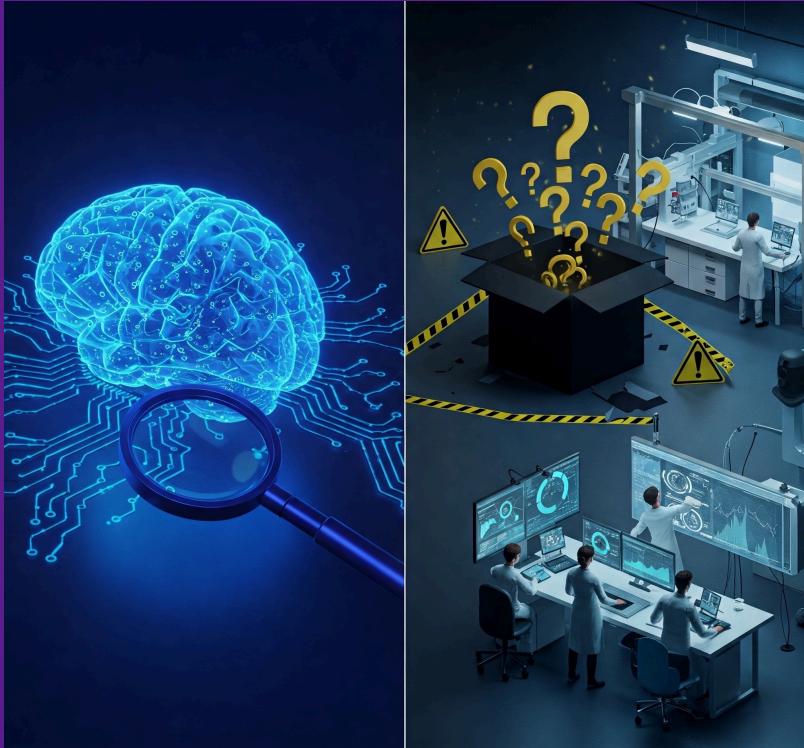


Security

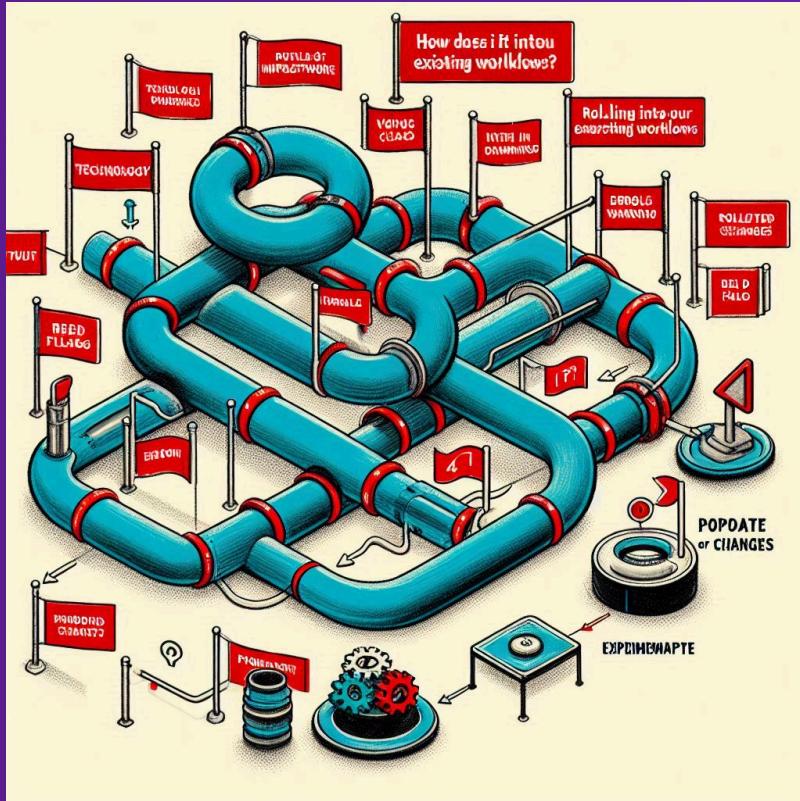


Ethics

Question 3:
What are the
risks?



Question 4:
Can we trust the
outputs?

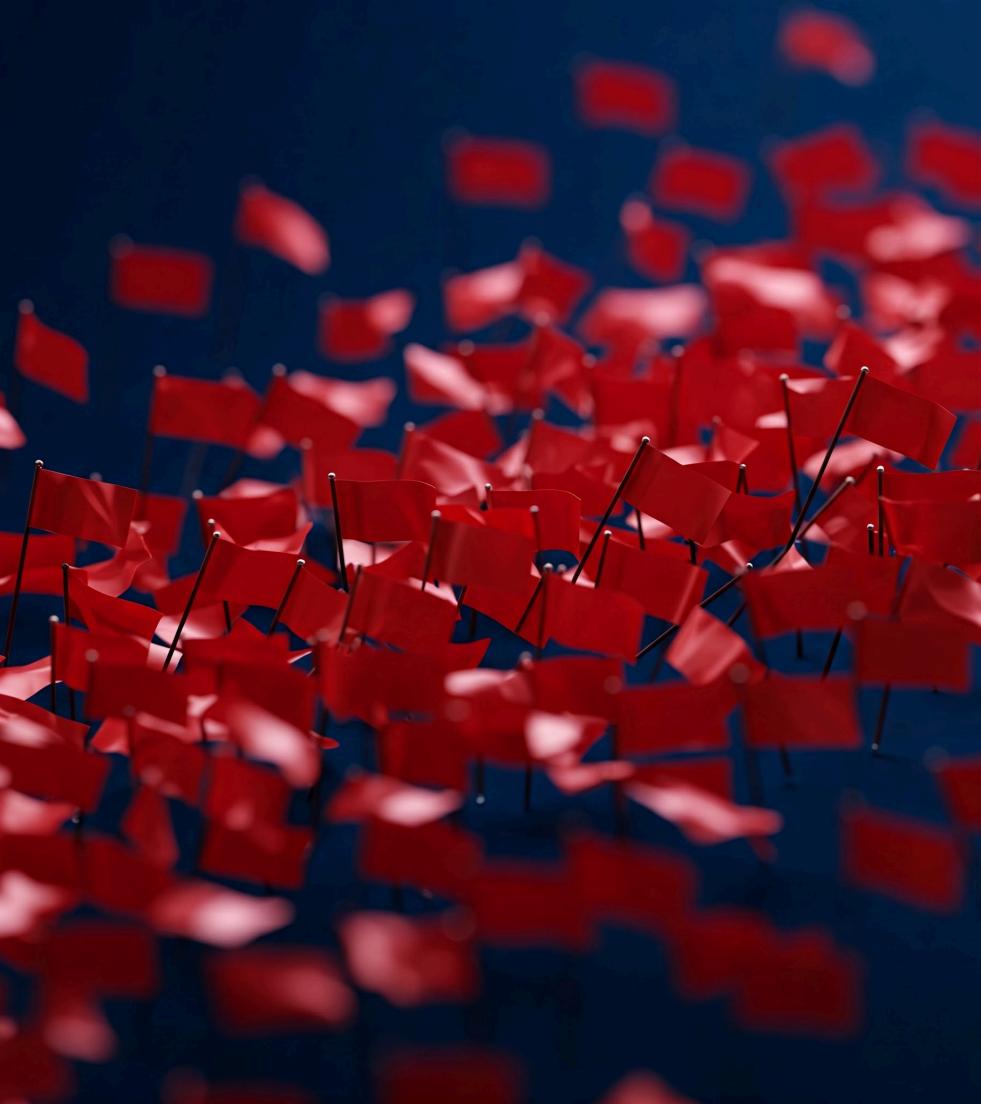


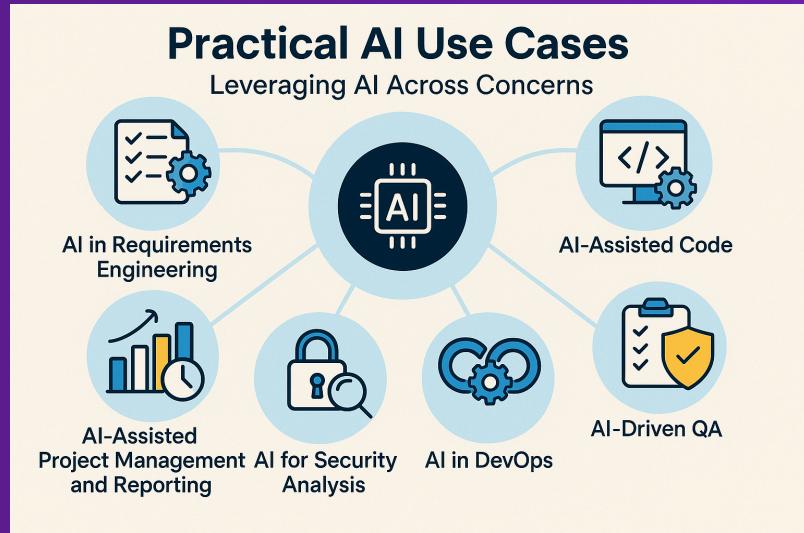
Question 5:

How does it fit with our existing workflows?

Red Flags to Watch For

- Hype-driven adoption
- Vague use cases
- Overemphasis on speed alone
- No clear KPIs
- Hidden quality costs
- Unverified data sources
- Weak security and privacy protocols
- No ethical guidelines
- Black-box models with no explanation
- Inadequate user training
- Lack of operational readiness
- No mechanism for verification
- Limited testing





Practical AI Use Cases

Leveraging AI Across Concerns

AI in Requirements Engineering

- NLP/LLMs for eliciting, analyzing, and validating requirements.
- Generating user stories and initial system models.
- **LEAP:**
 - **Learn:** Analyze requirements for ambiguity using NLP.
 - **Experiment:** Analyze a few existing user stories.
 - **Assess:** Gather feedback on improved clarity.
 - **Propagate:** Scale use the NLP tools.

AI-Assisted Code

- Tools like GitHub Copilot, Cursor, Gemini Code Assist, Amazon Q, and Claude Code.
- Capabilities: code generation, explanation, refactoring suggestions, bug fixes and identification.
- **LEAP:**
 - **Learn:** Research tools, their security implication, and best practices.
 - **Experiment:** Pilot with a small team on specific contained task.
 - **Assess:** Evaluate code quality and developer productivity.
 - **Propagate:** Roll out with guidelines, training, and best practices.

AI-Driven QA

- LLMs and specialized models for generating unit, integration, and end-to-end tests.
- AI for GUI testing, fuzz testing, visual regression, bug detection, chaos testing.
- **LEAP:**
 - **Learn:** Identify repetitive testing tasks.
 - **Experiment:** Use AI to generate tests for non-critical components.
 - **Assess:** Measure code coverage and effort required.
 - **Propagate:** Integrate successful techniques into QA workflows.

AI in DevOps

- Automation of CI/CD, intelligent monitoring, anomaly detection, predictive failure analysis.
- **LEAP:**
 - **Learn:** Identify manual DevOps bottlenecks.
 - **Experiment:** Automate a deployment or set of deployments.
 - **Assess:** Measure impact on deployment time and reliability.
 - **Propagate:** Expand automation across more services.

AI-Assisted Project Management and Reporting

- AI for task prioritization, resource allocation, and risk management.
- Automated reporting and status updates.
- **LEAP:**
 - **Learn:** Identify repetitive reporting tasks.
 - **Experiment:** Use AI to generate reports for a small project.
 - **Assess:** Measure time saved and accuracy.
 - **Propagate:** Integrate successful techniques into PM workflows.

AI for Security Analysis

- Enhanced threat detection, log analysis, vulnerability assessment, automated Security Operations Center (SOC) tasks.
- AI tools for code vulnerability scanning.
- **LEAP:**
 - **Learn:** Research AI-powered security tools.
 - **Experiment:** Run Static Application Security Testing (SAST) tools on a feature branch.
 - **Assess:** Analyze false positive/negative rates.
 - **Propagate:** Integrate validated tools into the DevSecOps pipeline.

Communicating AI Strategy to Leadership

Emphasize Value and the Risk Management

Conclusion & Takeaways

Embrace AI Thoughtfully and Strategically

Call to Action

Move beyond passive observation of AI.

Thank You!

✉️ john@hellocrosby.com

bsky: [@jccrosby.com](#)



<https://github.com/jcrosby/smart-ai-adoption>