

AI for Teams - Smart Adoption Not Blind FOMO



John Crosby

Principal Engineer, Client Engineering
at MLB

@jcc.bsky.social



Navigating the AI Wave in Software Development

Smart Adoption vs. Blind FOMO

The Current Landscape

1. The software development world is experiencing rapid change due to AI advancements - especially in with LLMs and generative AI.
2. Many developers and organizations are experimenting with or using AI tools in their workflows.
3. Applications span the Software Development Life Cycle, from coding to deployment.
4. Teams may feel overwhelmed by the hype and the sheer volume of available tools.
5. The pressure to "do something with AI" can lead to rushed decisions and poor outcomes.
6. The challenge is to navigate this landscape wisely, avoiding the pitfalls of blind FOMO while harnessing the potential of AI.

The Dual Edge: Opportunity vs. Hype

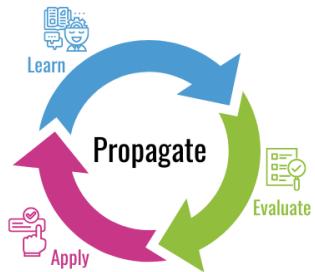
- **Opportunity:** AI offers benefits like enhanced productivity, improved code quality, faster development cycles, and potential for innovation.
- **Risk:** AI project failure rates remain high due to flawed adoption strategies.
- **Common Pitfalls:**
 - Lack of clear goals, no coherent strategy, neglecting data readiness, and "Blind FOMO".
 - Pressure to "do something with AI" can exacerbate these issues.

Smart Adoption vs. Blind FOMO: The Core Difference

- **Smart Adoption:** Deliberate, strategic approach focused on solving specific problems and achieving measurable improvements. Involves learning and assessment before significant investment.
- **Blind FOMO:** Reactive, technology-centric adoption driven by hype rather than a specific problem. Often skips crucial learning and assessment.
- **How AI is adopted is more critical than whether it is adopted.**

LEAP

- Learn: understand & educate
- Evaluate: examine & consider
- Apply: Integrate & monitor
- Propagate: Refine & iterate

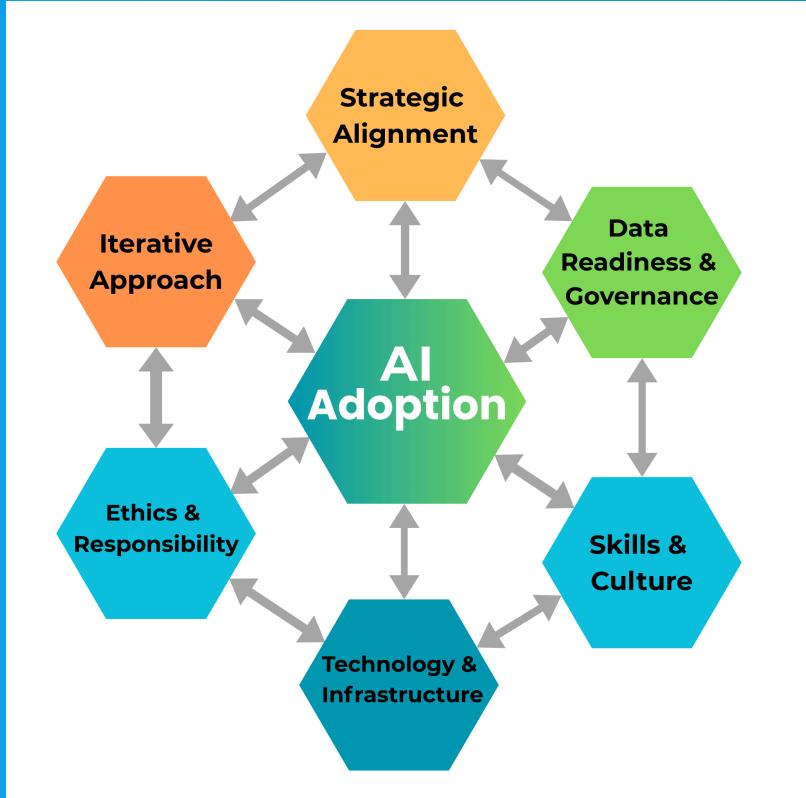


Introducing the LEAP Framework: A Practical Compass

Learn, Experiment, Assess,
Propagate/Plan

LEAP: A Structured Approach

- LEAP is designed to guide teams through AI adoption wisely.
- Focuses on maximizing value while mitigating risks.
- Encourages a shift from reactive FOMO to proactive, strategic adoption.
- Emphasizes iterative progress and critical evaluation.
- Provides a sustainable and effective structure for AI efforts.



Core Principles of Effective AI Adoption

Foundational to Smart Adoption and LEAP

Six Core Principles

- **Strategic Alignment:** AI initiatives must solve specific problems aligned with business goals. Define *why* before *what*.
- **Data Readiness & Governance:** Strong data quality, accessibility, security, and governance are essential. Poor data leads to failure.
- **Skills & Culture:** Requires technical skills and a supportive culture encouraging experimentation and learning.
- **Technology & Infrastructure:** Select the right tools compatible with existing workflows.
- **Ethics & Responsibility:** Integrate ethical considerations (fairness, transparency, privacy, security) from the start.
- **Iterative Approach:** Start with small pilots, learn, and scale successful initiatives.

The LEAP Framework for Software Teams

Actionable Steps for Smart
Adoption

LEARN

- **Understand AI capabilities and limitations.**
- **Activities:**
 - Identify pain points in the SDLC.
 - Research relevant AI tools and advancements.
 - Assess team skills and identify gaps.
 - Evaluate data availability and quality.
 - Consider ethical and security implications.
 - Define **SMART goals** for experiments.
- **Output:** Prioritized use cases, readiness assessment, defined success metrics.

EXPERIMENT

- Focus on small-scale, controlled pilots to test hypotheses.
- **Activities:**
 - Select a low-risk pilot project.
 - Choose specific AI tool/model.
 - Develop and execute a prototype or limited integration.
 - Gather quantitative and qualitative feedback.
- **Output:** Working prototype, performance data, developer feedback, lessons learned.
- **Share:** Share your prototype, performance data, feedback, and those lessons learned.

Experimenting and sharing is large part of expanding the usage and benefits of AI

ASSESS

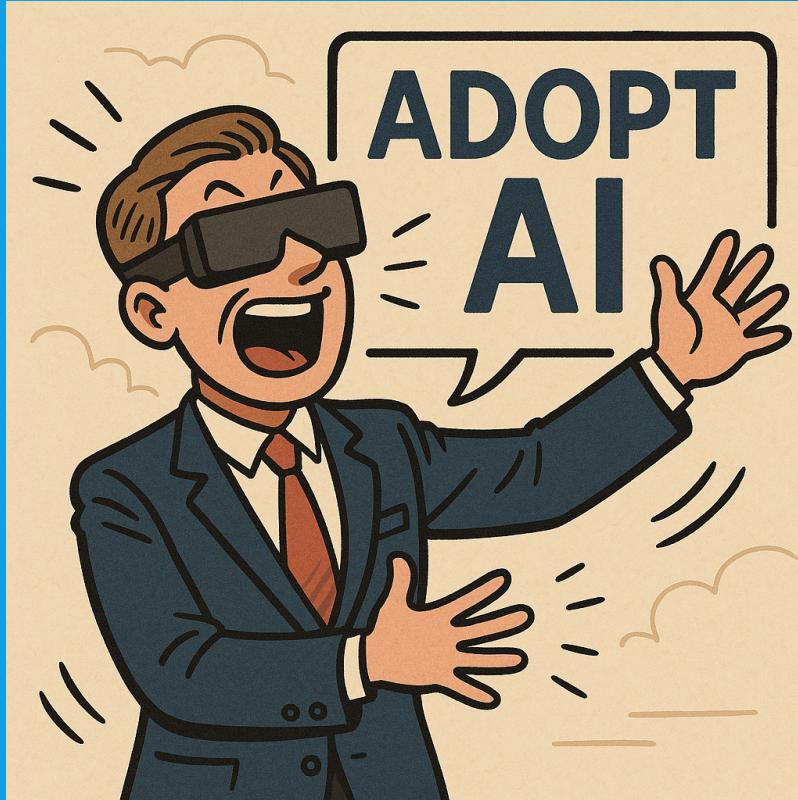
- Evaluation experiment outcomes rigorously to make informed decisions.
- **Activities:**
 - Evaluate results against SMART goals.
 - Analyze ROI (benefits vs. costs).
 - Assess impact on workflow, quality, productivity, satisfaction.
 - Identify technical challenges and risks.
 - Review ethical and security posture.
 - Make a clear **GO/NO-GO/PIVOT** decision.
- **Output:** Assessment report, clear decision, refined understanding of requirements and the results.

PIVOT

- If the experiment fails or needs adjustment, pivot to a new approach.
- 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, and 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.
 - Implement change management practices.
 - Continuously monitor performance and ethical implications.
- **Output:** Scaled AI solution, governance structures, ongoing monitoring plan, up-skilled team.

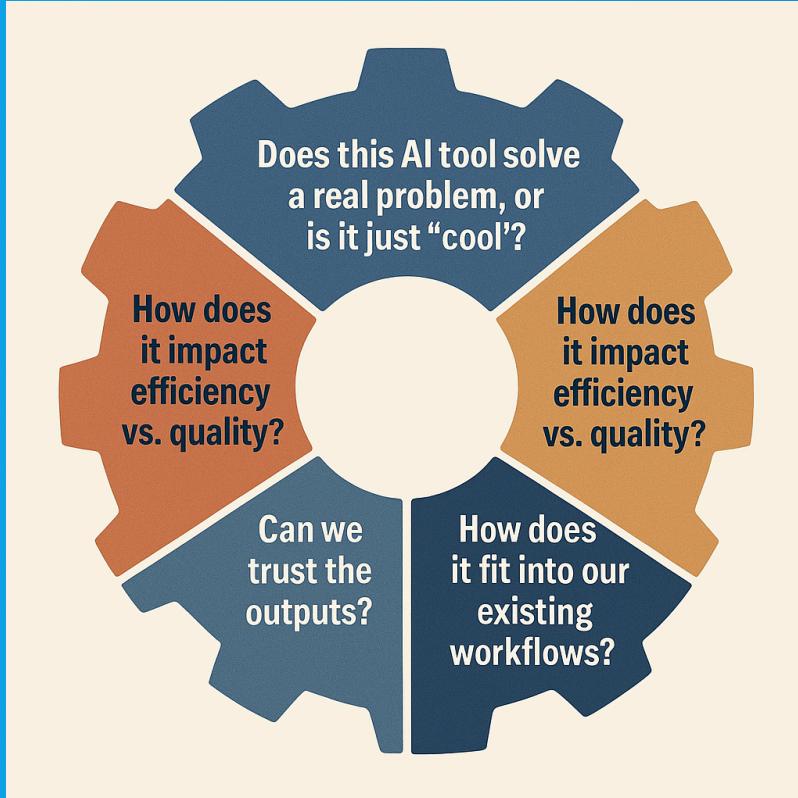


Analogy - Blind FOMO

Mandated Generative AI for Code

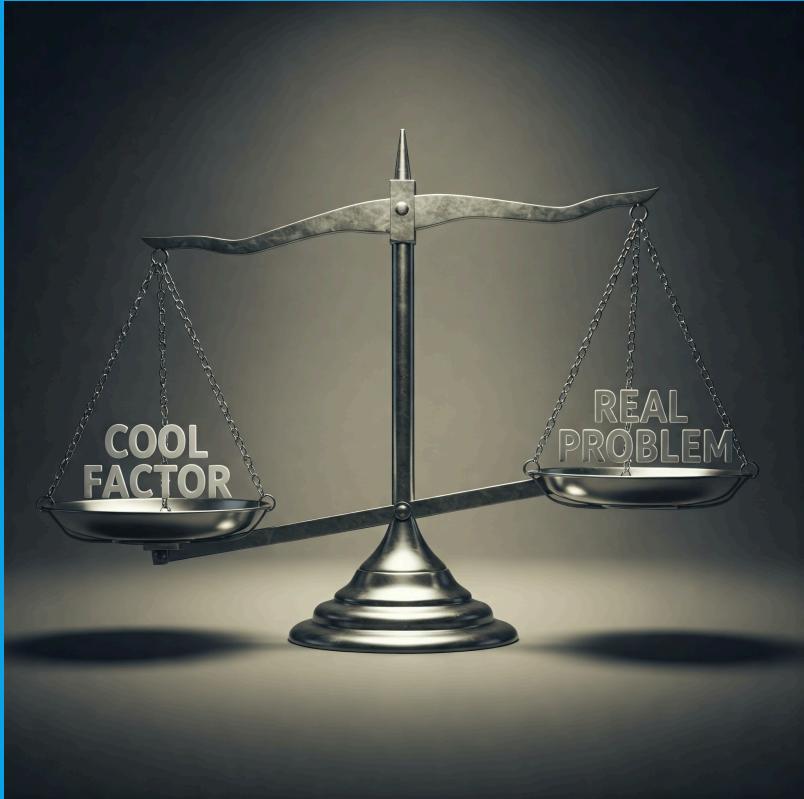
Blind FOMO - Skipped LEAP Steps

- **Skipped Learn:** Vague goal ("increase velocity"), ignored readiness (skills, codebase, IP), focused on tech (GenAI) over specific problems.
- **Skipped Experiment/Assess:** Underestimated verification needs, overlooked accuracy and security risks, ignored domain/context limitations.
- **Failed Propagate/Plan:** Resulted in chaos, inconsistent adoption, decreased productivity, potential security issues, no way to measure impact.



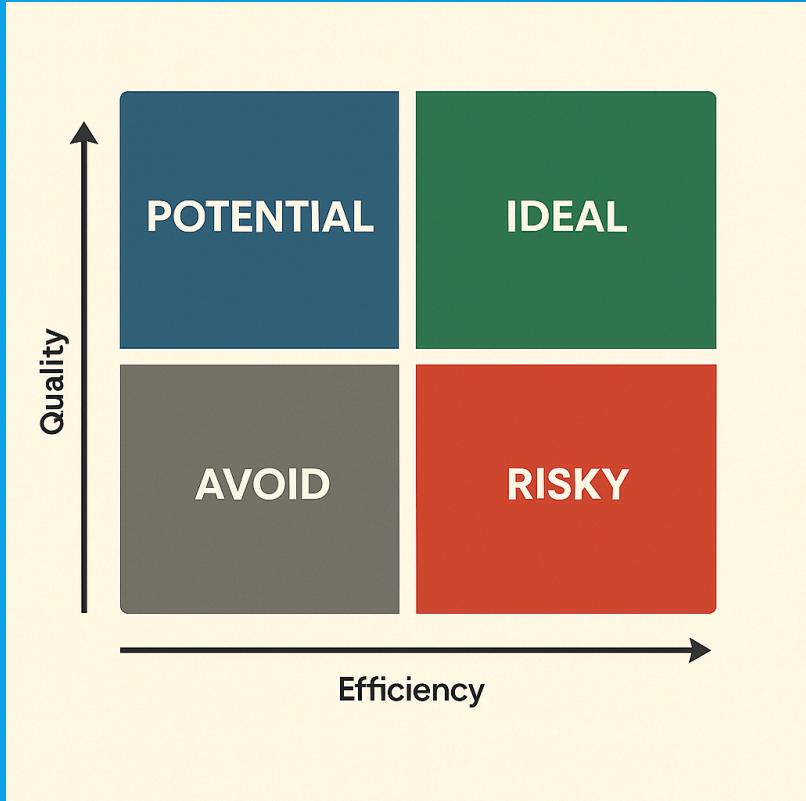
Smart AI Adoption: How to Evaluate AI for Your Team

Five Questions to Ask



Question 1: Does this AI tool solve a real problem, or is it just ‘cool’?

- **Aligns with Strategic Alignment:** Ensure the AI addresses a specific bottleneck or pain point.
- **Red Flag:** Vague use cases with no clear, tangible problem.
- **LEAP - Learn:** Identify team bottlenecks and potential AI solutions.



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

- **Weigh the trade-offs:** Focus solely on speed can hide quality costs.
- **Red Flag:** Overemphasis on speed alone without considering accuracy.
- **LEAP - Assess:** Analyze the impact on workflow and code quality.

RISKS?



Bias



Security



Ethics

Question 3: What are the risks (bias, security, ethics)?

- **Integrate Ethics & Responsibility:** Consider potential biases, security vulnerabilities, and ethical implications.

- **Red Flags:** Unverified data, weak security, no ethical guidelines.

- **LEAP - Learn & Assess:** Proactively consider risks and evaluate the ethical posture.

Question 4: Can we trust the outputs?

- **Emphasize Verification:** Understand AI's decision-making process and have mechanisms to verify its results.
- **Red Flags:** Black-box models, no verification process, limited testing.
- **LEAP - Experiment & Assess:** Gather data and feedback on the reliability of AI outputs.

Question 5: How does it fit into our existing workflows?

- **Technology & Infrastructure:** Ensure compatibility and minimize disruption.
- **Red Flags:** Major workflow disruption, inadequate training, lack of IT/operations readiness.
- **LEAP - Propagate/Plan:** Plan for integration and necessary changes.

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 protocols
- No ethical guidelines
- Black-box models with no explanation
- Lack of operational readiness
- No mechanism for verification
- Limited testing
- Major workflow disruption
- Inadequate user training



Practical AI Use Cases for Teams

Leveraging AI Across Concerns

AI in Requirements Engineering (RE)

- NLP/LLMs for eliciting, analyzing, and validating requirements.
- Generating user stories and initial system models.
- **LEAP:**
 - **Learn:** Analyze requirements for ambiguity using NLP.
 - **Experiment:** Use NLP to analyze a small set of user stories.
 - **Assess:** Gather feedback on improved clarity.
 - **Propagate/Plan:** Scale use of validated NLP tools.

AI-Assisted Code

- Tools like GitHub Copilot, Gemini Code Assist, Amazon Q, & Claude Code.
- Capabilities: code generation, explanation, refactoring suggestions, bug identification.
- **LEAP:**
 - **Learn:** Research tools and their security implications.
 - **Experiment:** Pilot with a small team on specific tasks.
 - **Assess:** Evaluate code quality and developer productivity.
 - **Propagate/Plan:** Roll out with 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.
- **LEAP:**
 - **Learn:** Identify repetitive testing tasks.
 - **Experiment:** Use AI to generate tests for non-critical components.
 - **Assess:** Measure code coverage and effort required.
 - **Propagate/Plan:** Integrate successful techniques into QA workflows.

AI in DevOps

- Automation of CI/CD, intelligent monitoring, anomaly detection, predictive failure analysis.
- MLOps platforms for managing ML pipelines.
- **LEAP:**
 - **Learn:** Identify manual DevOps bottlenecks.
 - **Experiment:** Automate deployment of a microservice.
 - **Assess:** Measure impact on deployment time and reliability.
 - **Propagate/Plan:** 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/Plan:** 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/Plan:** Integrate validated tools into the DevSecOps pipeline.

Communicating AI Strategy to Leadership

Emphasize Value, Risk
Management, and the LEAP
Framework

Key Principles for Smart Integration

- **Strategic Alignment:** Solve real problems with clear goals.
- **Iterative Experimentation:** Start small and gather evidence.
- **Rigorous Assessment:** Evaluate benefits and risks objectively.
- **Focus on Value:** Prioritize tangible improvements.
- **Continuous Learning:** Adapt to the evolving AI landscape.

The Enduring Value of LEAP

- A practical, repeatable process for wise AI adoption.
- **Learn** thoroughly before acting.
- **Experiment** in a controlled manner.
- **Assess** results critically.
- **Propagate/Plan** successful initiatives sustainably.
- Integrates data, skills, ethics, and governance.
- Fits naturally with agile development.

Call to Action

- Move beyond passive observation of AI.
- **Initiate a 'Learn' cycle:** Identify a pressing team challenge.
- Research potential AI solutions.
- Define a clear goal for a small experiment.
- Start the LEAP cycle and embrace the learning process.
- Chart a course for wise AI adoption that delivers real value.

Share your successes and failures!

Thank You!



john@hellocrosby.com

[@jcc.bsky.social](https://jcc.bsky.social)