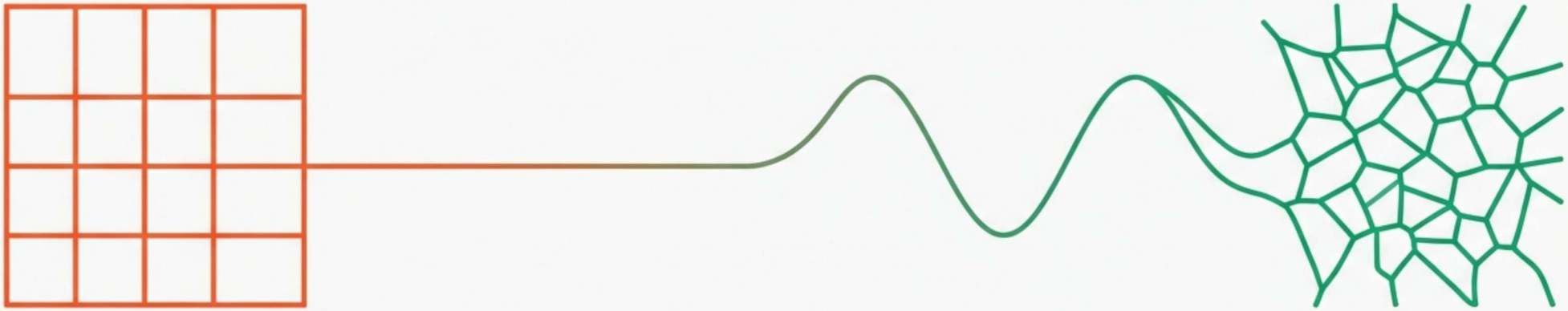
Architectures of Intelligence

From Rigid Rules to Adaptive Learning

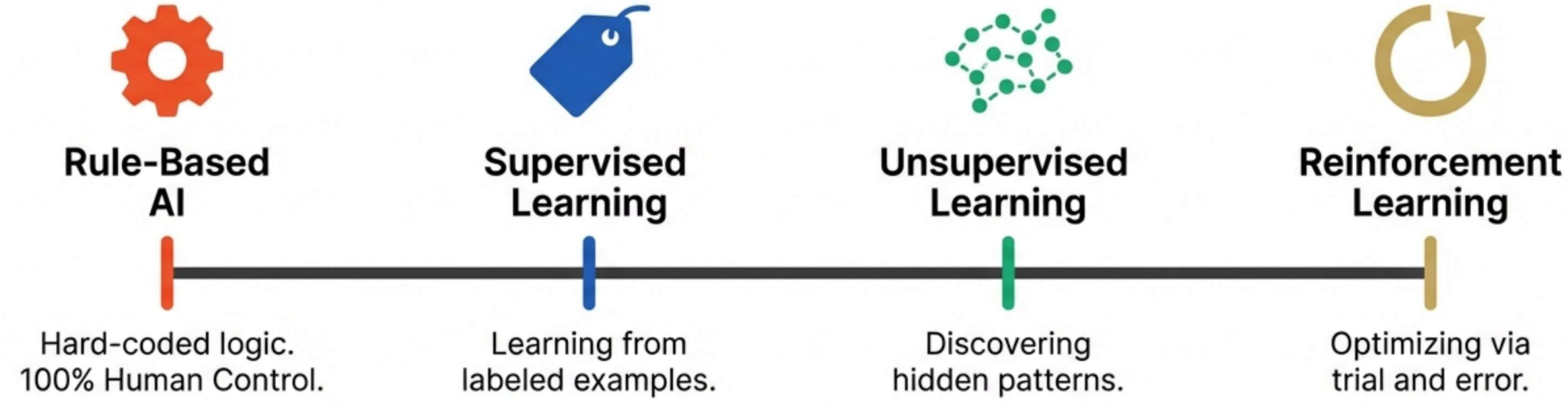


A structural breakdown of **Rule-Based Systems**, **Supervised Learning**, **Unsupervised Learning**, and **Reinforcement Learning**.

Prepared for Strategic Review

Executive Summary

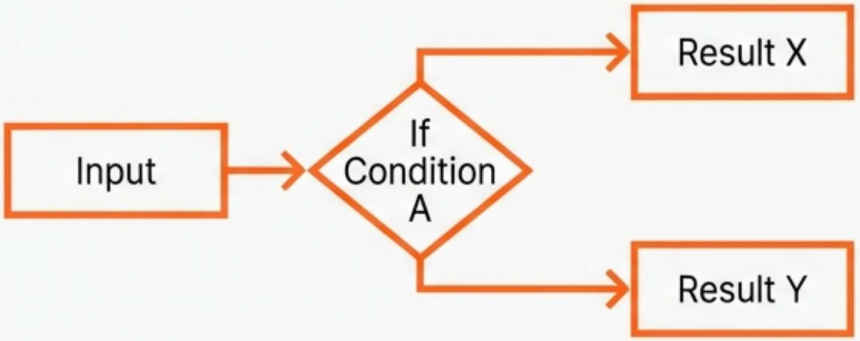
Artificial Intelligence spans a spectrum from explicit human instruction to implicit data inference.



KEY INSIGHT: The choice of architecture is a trade-off between the safety of pre-written rules and the flexibility of adaptive learning.

The fundamental divide is between **hard-coded** rules and **learned patterns.**

Rule-Based AI

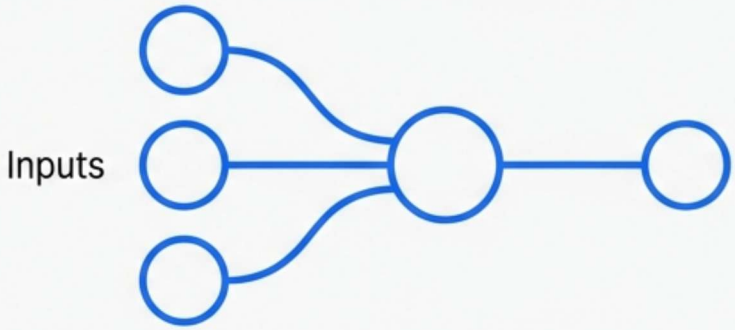


Definition: Tools that follow specific, hard-coded instructions created by developers.

Mechanism: Does not learn from new data. Follows a strict logic path.

Role: Precise execution of instructions.

Machine Learning



Definition: Tools that develop their own logic by analyzing datasets.

Mechanism: Identifies patterns, features, and characteristics to make predictions.

Role: Adaptation to complex, real-world variability.

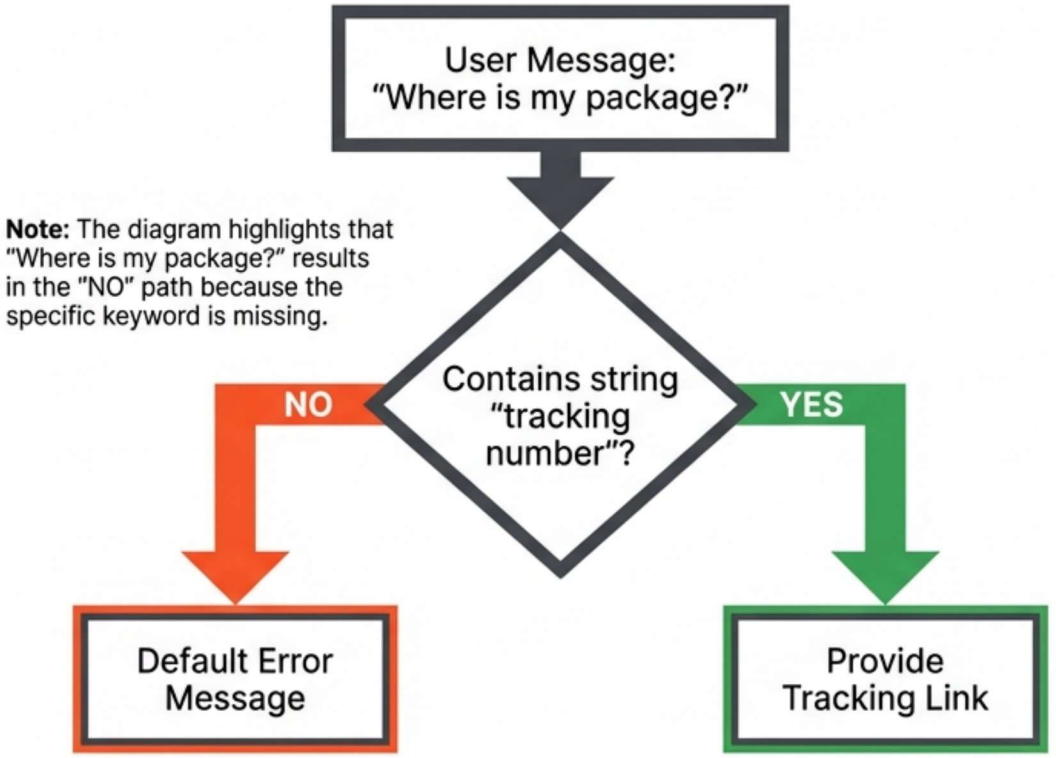
Rule-based systems excel at predictability but fail at adaptation.

Often found in workplace tools designed for repetitive tasks.

The Mechanics:

- Logic is strictly “If/Then”.
- System cannot understand requests outside pre-written code.

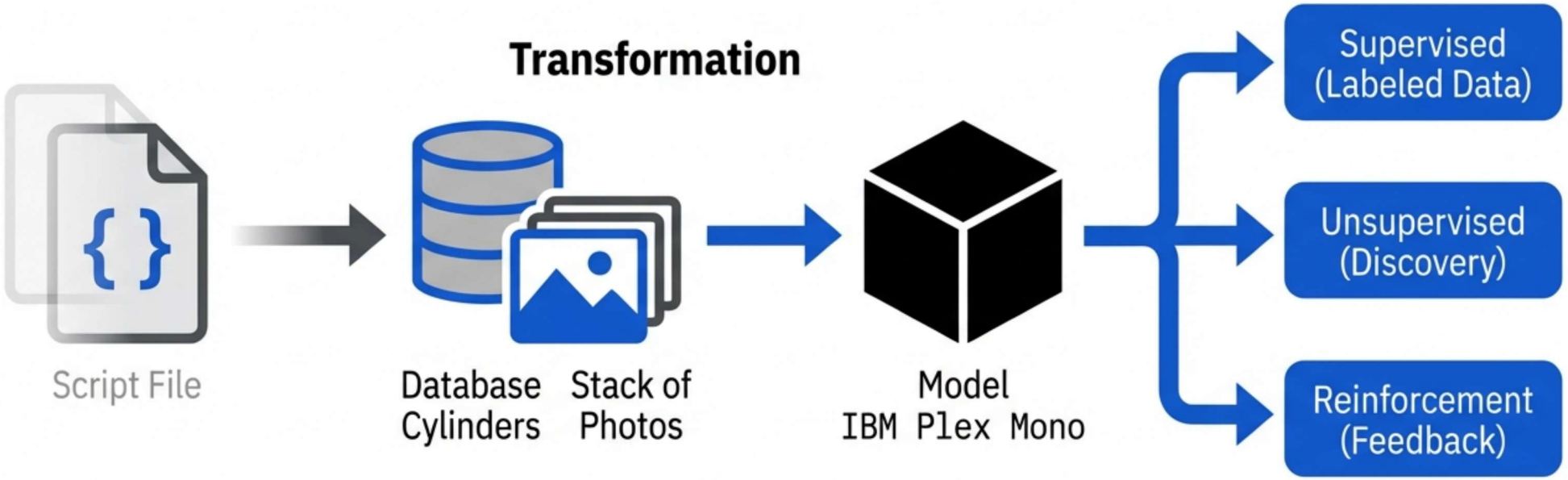
Concrete Example: A Customer Service Chatbot.
The Rule: If user message contains “tracking number” → Respond with link.
The Result: The bot follows this precisely every time.
The Limitation: It cannot adapt to phrasing that doesn’t trigger the keyword.



Note: The diagram highlights that “Where is my package?” results in the “NO” path because the specific keyword is missing.

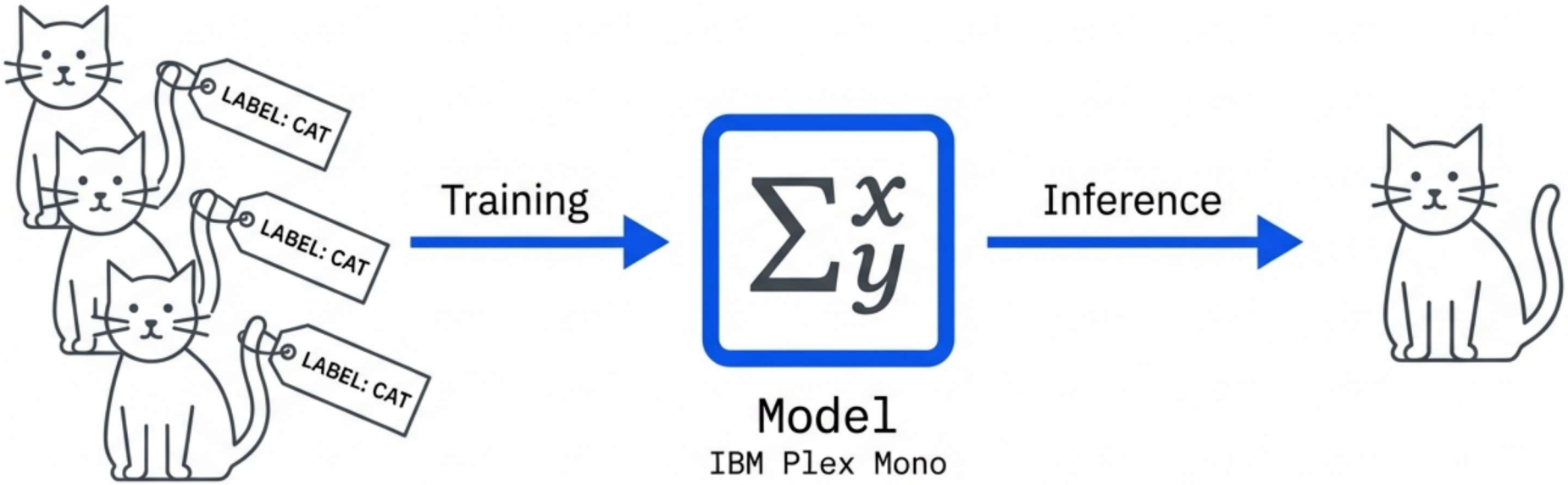
Machine Learning shifts the burden from coding rules to curating data.

Developers provide the data and architecture; the machine finds the solution.



ML Type 1: Supervised Learning

Mapping inputs to known outputs using human labels.

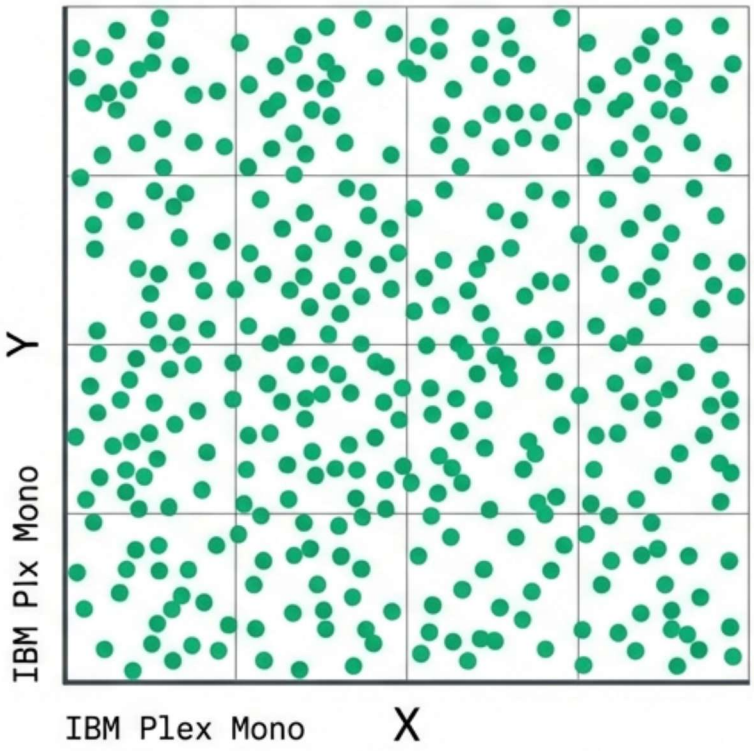


When to use: When there is a specific, known output in mind.
Key Insight: The machine doesn't know what a cat is; it knows the statistical patterns associated with the label 'cat'.

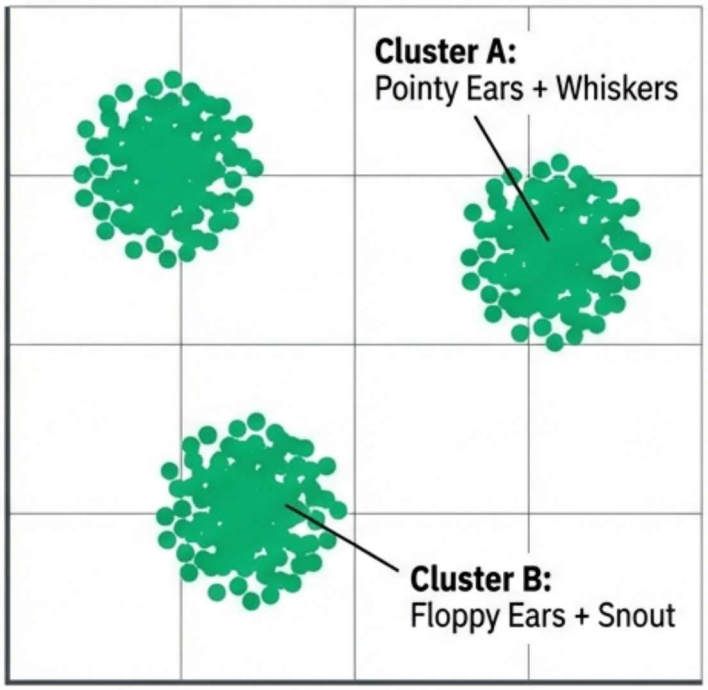
ML Type 2: Unsupervised Learning

Discovering hidden structures in data without human guidance.

A. Input: Unlabeled Data



B. Output: Clustered Patterns



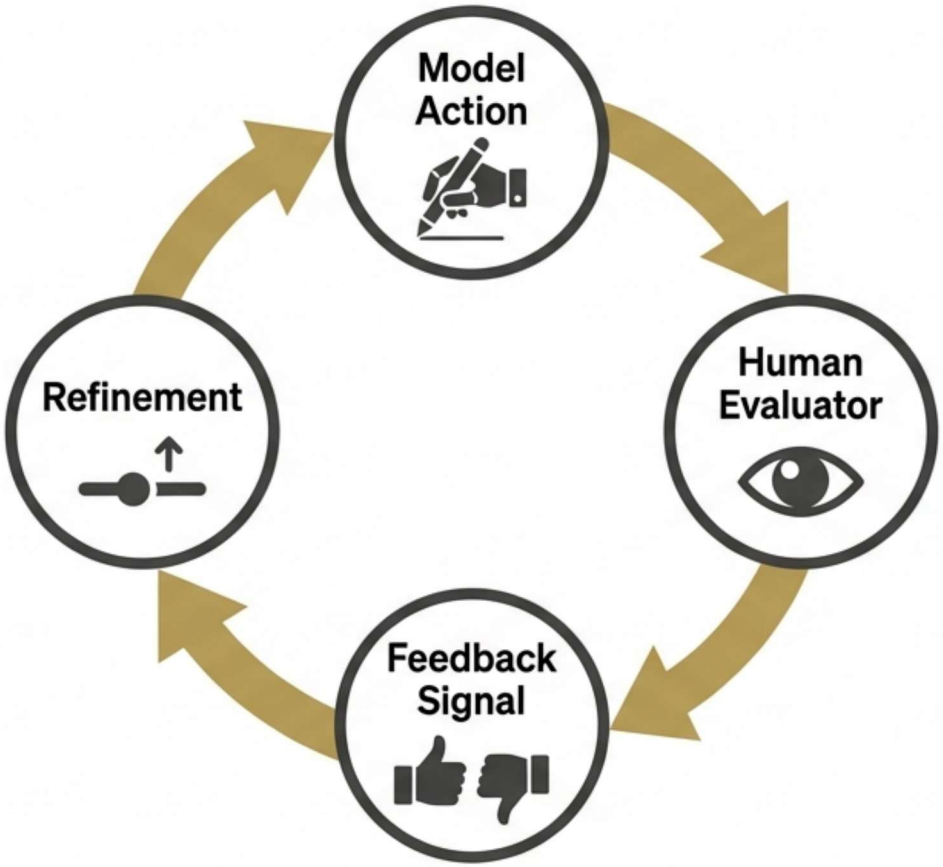
When to use: When there is no specific known output, but a need to identify groups.

The Process:

1. **Input:** Dataset of animal photos without labels.
2. **Training:** Tool analyzes data for similarities.
3. **Clustering:** Groups images by features (e.g., whiskers).
4. **Result:** Learns the concept of 'cat' implicitly.

ML Type 3: Reinforcement Learning

Optimizing performance through trial, error, and feedback.



When to use: To continuously refine performance on a specific task.

The Process:

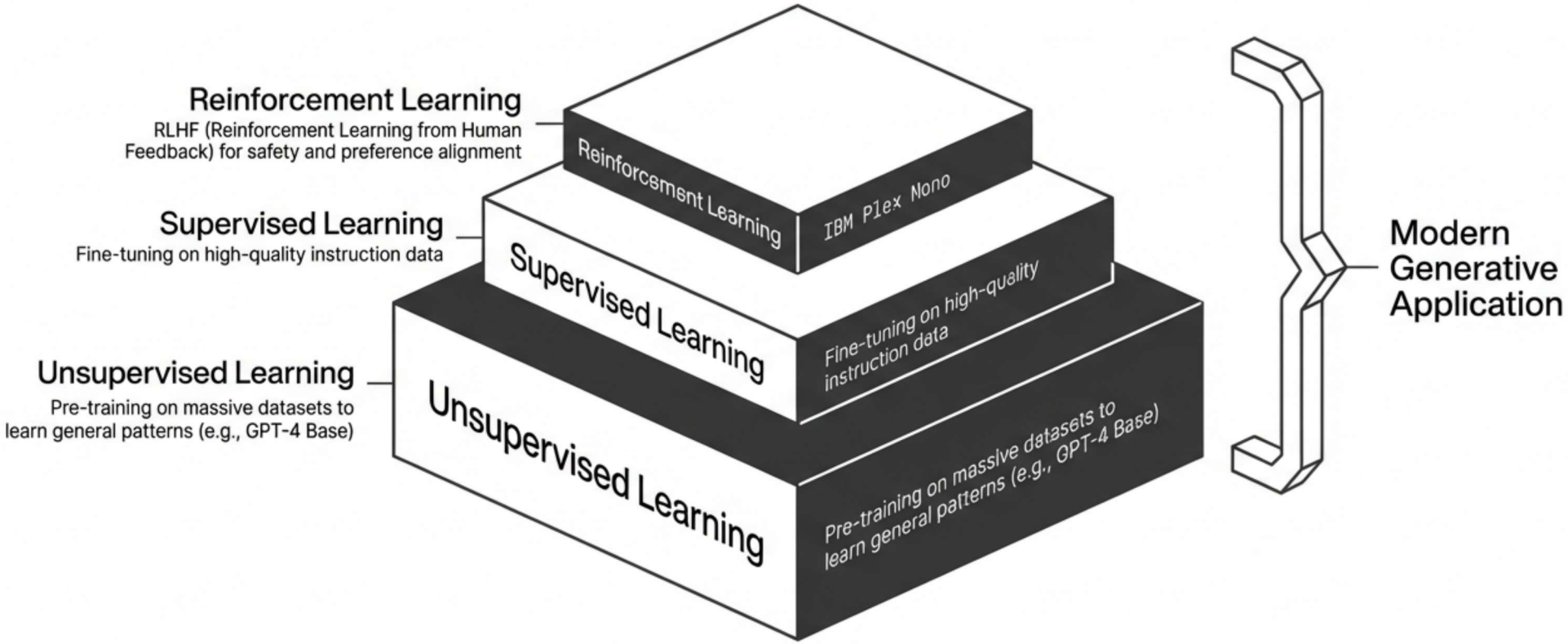
1. **Action:** The model attempts a task (e.g., generates an image).
2. **Feedback:** Human reviews the output.
3. **Signal:** Positive or negative critique is logged.
4. **Refinement:** Developers use feedback to adjust the model.

Comparing the three architectures of learning.

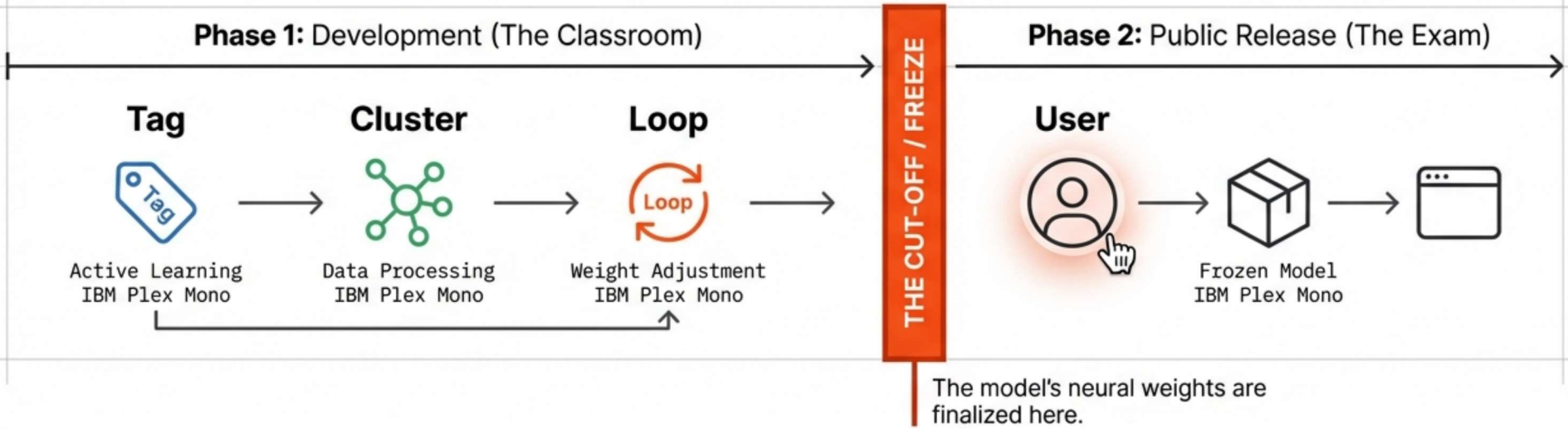
Type	Goal	Data Source	The “Cat” Analogy
Supervised	Specific Known Output	Labeled by Humans	Taught via explicit tags (“This is a cat”).
Unsupervised	Pattern Identification	Unlabeled / Raw	Discovers “whiskers/ears” cluster on its own.
Reinforcement	Continuous Refinement	Feedback Loops	Improves generation based on critique of past attempts.

Modern Generative AI is often a hybrid of these architectures

Today's most powerful tools rarely rely on a single method. They utilize a combination of approaches to achieve high-fidelity, safe, and useful results.



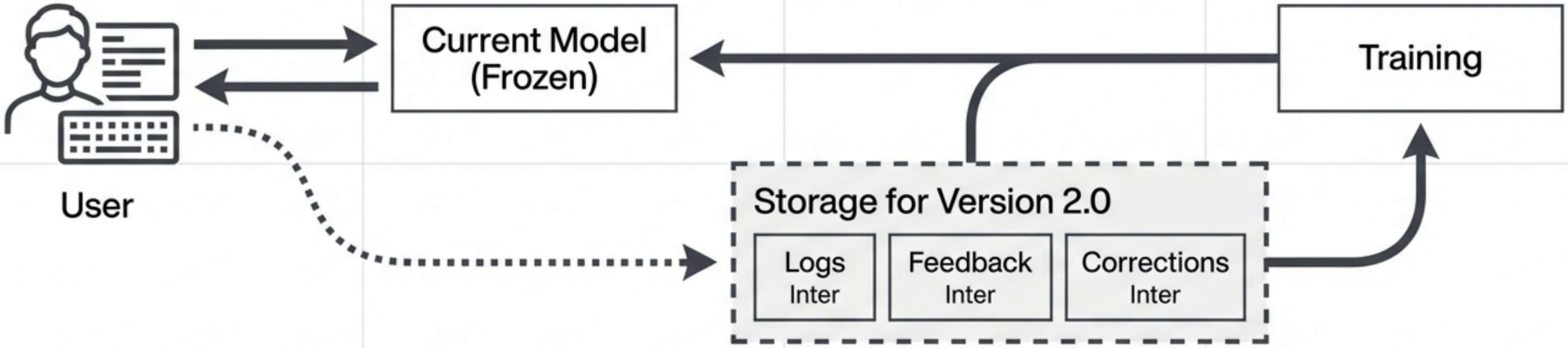
A critical distinction: Learning happens during development, not deployment.



- The **Myth**: The AI learns from me in real-time. 🤖
- The **Reality:** The tool applies what it learned previously. It is not actively updating its neural pathways while you use it. ✓

Swiss Editorial Technical

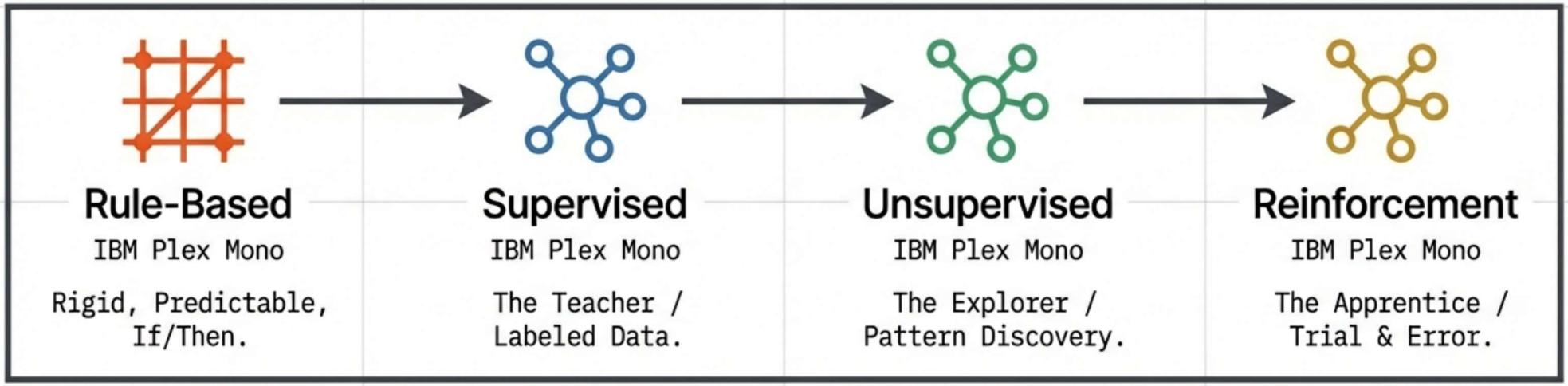
User interactions inform future versions, but the AI does not evolve in real-time.



The ‘brain’ of the AI you are using today is static. Your feedback is being filed away to train the brin of the AI you will use next year.

IBM Plex Mono

Summary: The Spectrum of Autonomy



Understanding these architectures allows us to move past 'AI magic' and evaluate tools based on how they process information and what level of autonomy they require.