Module 2 · Session 1

Understanding AI, intelligence and problem solving

The AI Spectrum — From Classical AI to Agentic AI

Leadership with AI - IIT Bombay. Rajendra M Sonar, Ph.D.

What we are going to learn

Understanding the science, art and engineering of building intelligence

1. AI context
AI revolution, impact, and why AI changes how work is designed.

2. Intelligence basics & knowledge
Data sources, learning, reasoning, memory, creativity, adaptability and problem solving.

Building intelligence

Acquire → Structure → Understand → Apply → Adapt

Knowledge + Data + Tools + Orchestration

3. AI spectrum
Semantic AI, traditional ML, Gen AI and Agentic AI — when each is useful. Building semantic intelligence.

4. Metadata & its role
How meta-data connects the dots, helps in increasing accuracy and model operational and common-sense intelligence

5. Modelling Generic entities
How generic entities like party, product, resource, transaction, etc. can be modelled?

AI: When it started!

VOL. LIX. No. 236.] [October, 1950

MIND

A QUARTERLY REVIEW

OF

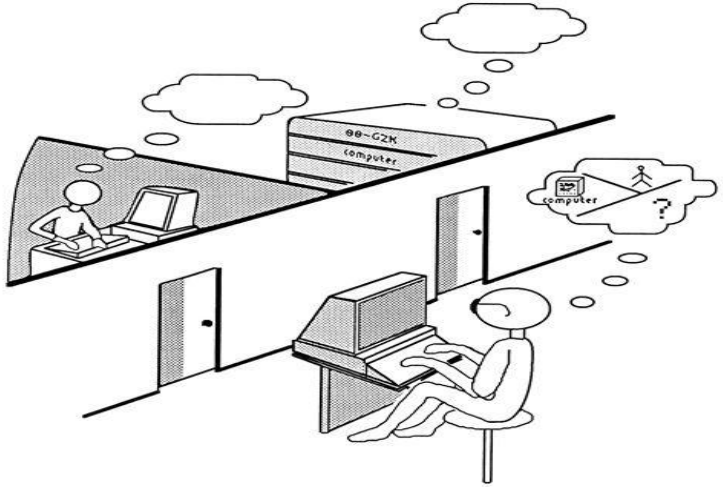
PSYCHOLOGY AND PHILOSOPHY

I.—COMPUTING MACHINERY AND INTELLIGENCE

BY A. M. TURING

1. *The Imitation Game.*

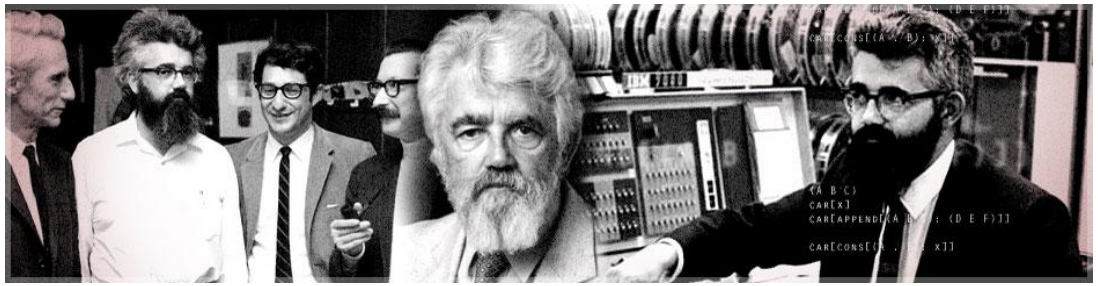
I PROPOSE to consider the question, 'Can machines think ?'



By Jack Copeland, © Copyright B.J. Copeland, July 2000

No machine has passed this test so far!
Will it ever be passed by machines???
(machines simply don't have natural and common sense intelligence like biological species)

Prof. John McCarthy



- "The science and engineering of making intelligent machines".
- He was the first to introduce the term Artificial Intelligence.
- A Proposal for the Dartmouth Summer Research Project on Artificial Intelligence, August 31, 1955 (1956?).
- Was Professor of Computer Science at Stanford University

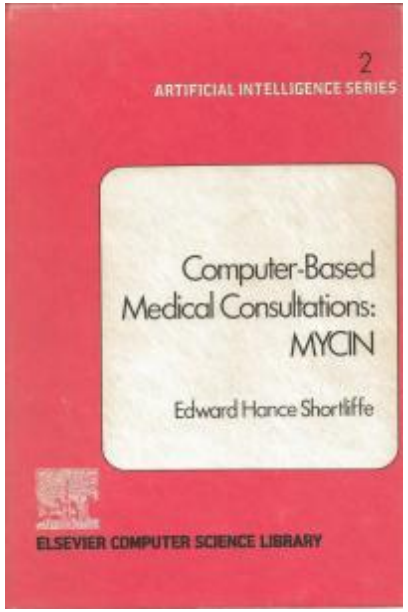
AAAI* (founded 1979) defines AI as:

The scientific understanding of the mechanisms underlying thought and intelligent behaviour and their embodiment in machines – www.aaai.org

AI in year 1984! Prof John McCarthy and Prof. Edward Feigenbaum speak

*Association for the Advancement of Artificial Intelligence

AI: When it started!



Edward H. Shortliffe
286 pp., references, index, illus.
ISBN-10: 0444569693
ISBN-13: 978-0444569691
Elsevier, New York, 1976

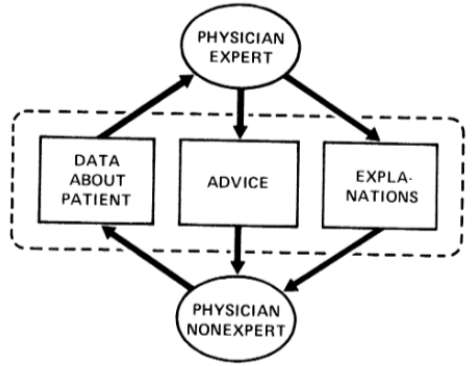
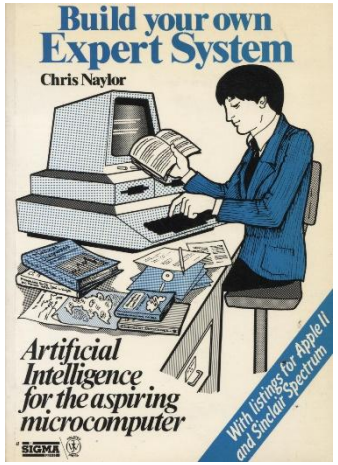


Figure 1 - Diagram summarizing the flow of information between physician and expert in the human consultation process. (Figure reproduced from reference 10).



Expert Systems: Principles and Practice*

Edward A. Feigenbaum
Professor of Computer Science
Stanford University

Most applications of the Artificial Intelligence (AI) science and technology are of a type called Expert Systems. An Expert System (ES) is a computer program that reasons using knowledge to solve complex problems. This overly brief caricature will be expanded upon below, but it serves to indicate an alignment of ES with AI's long term goals. Traditionally, computers solve complex problems by arithmetic calculation (not reasoning using logic); and the knowledge needed to solve the problem is known only by the human programmer and used to cast the solution method in terms of algebraic formulas.

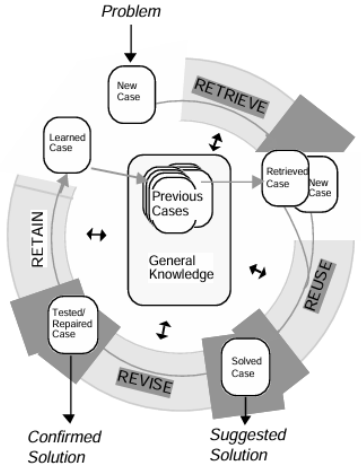
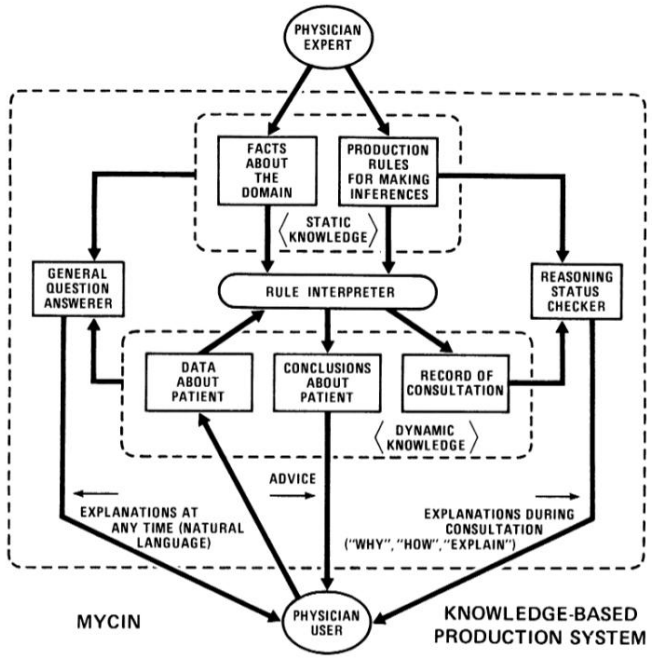
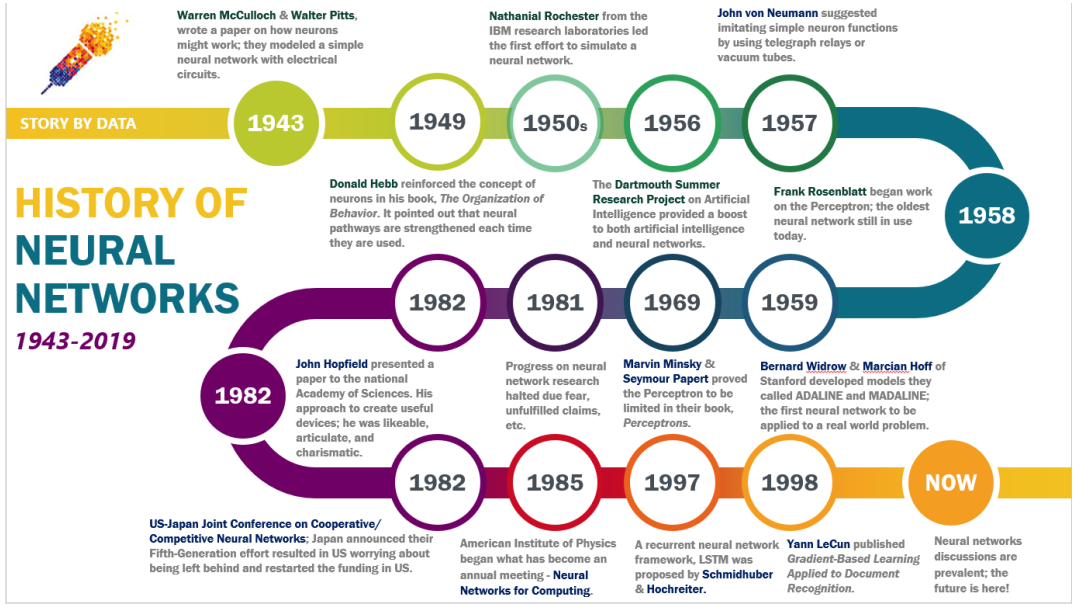


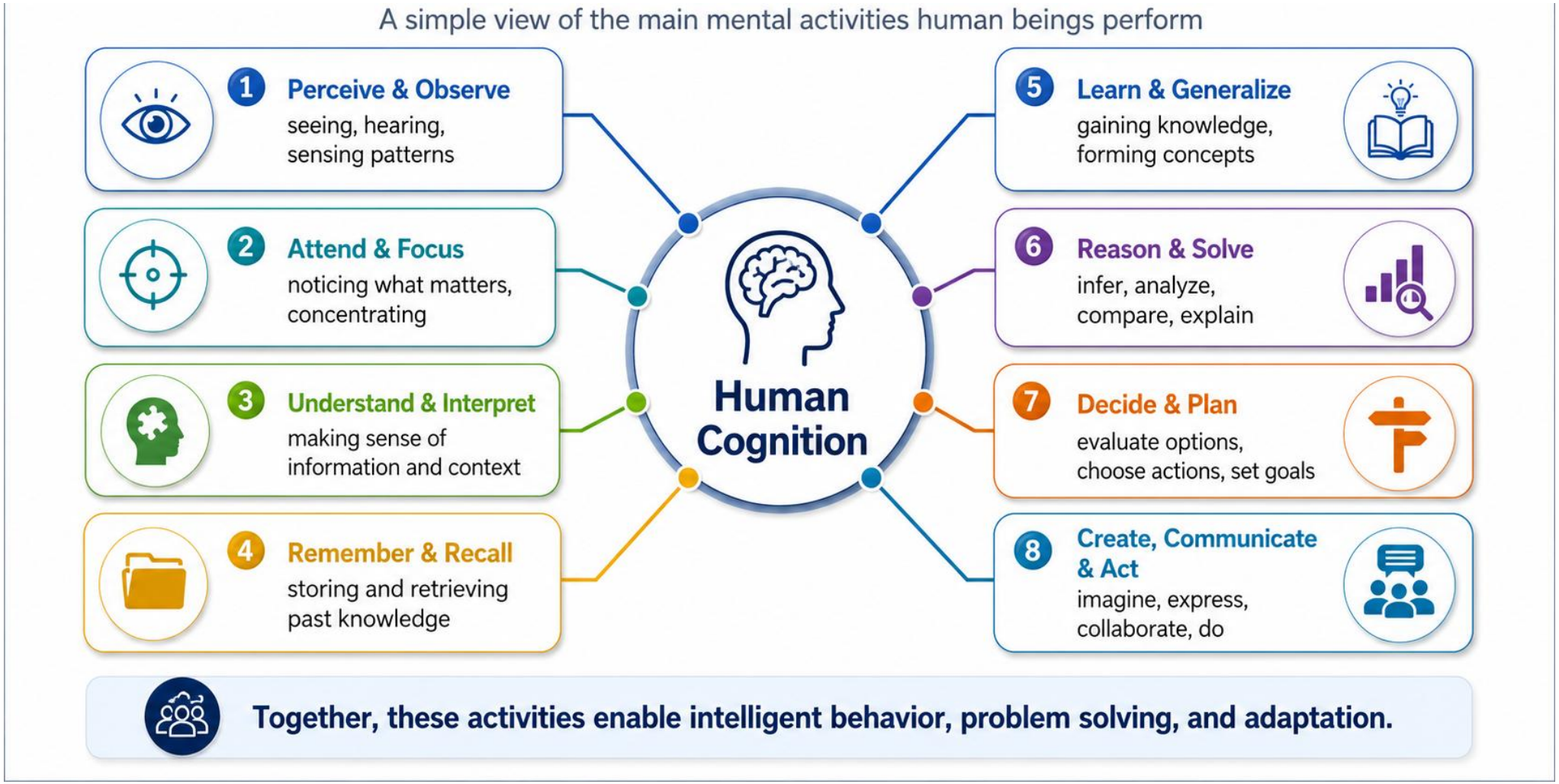
Figure 1. The CBR Cycle



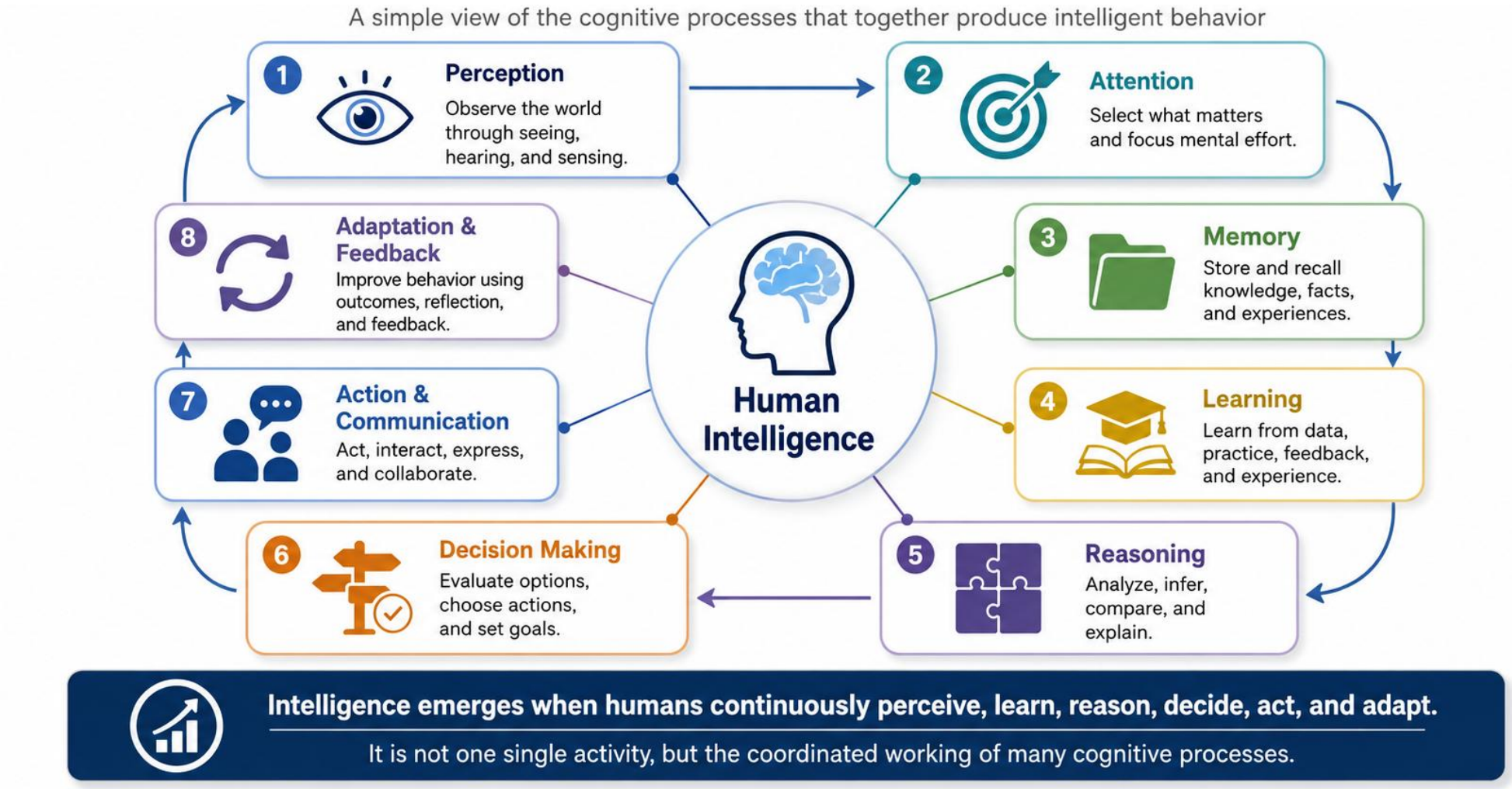


https://medium.com/analytics-vidhya/brief-history-of-neural-networks-44c2bf72eec

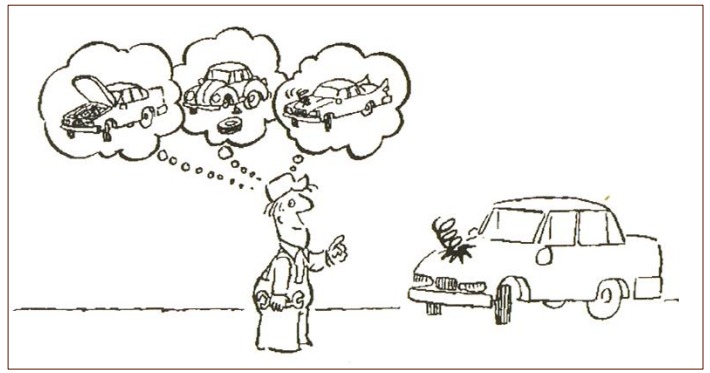
Understanding human cognitive activities

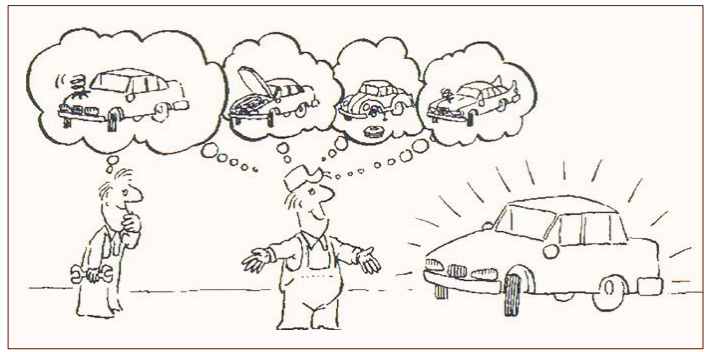


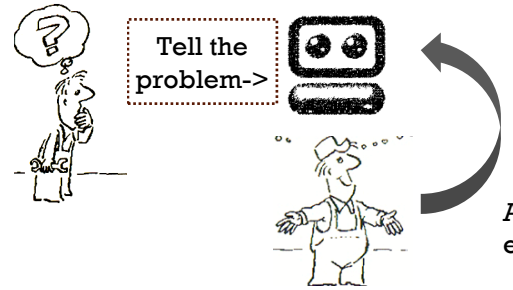
Understanding how humans achieve intelligence

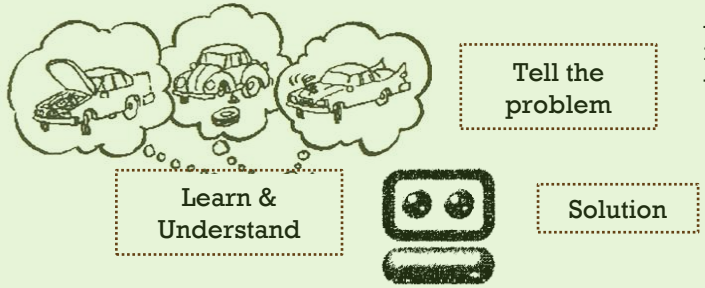


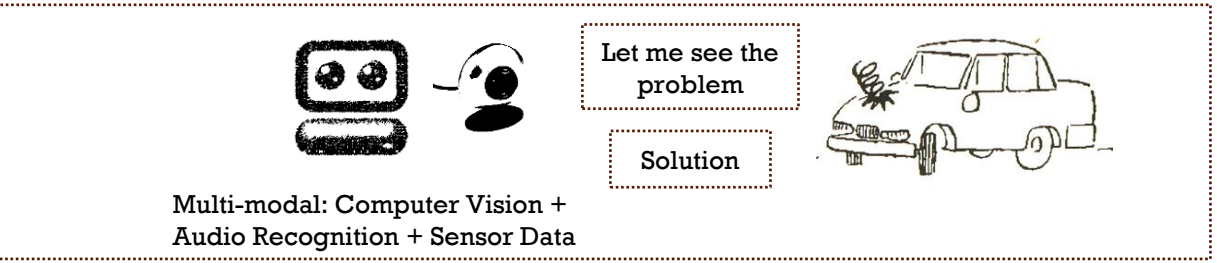
Understanding how human reasoning works & automation levels and options

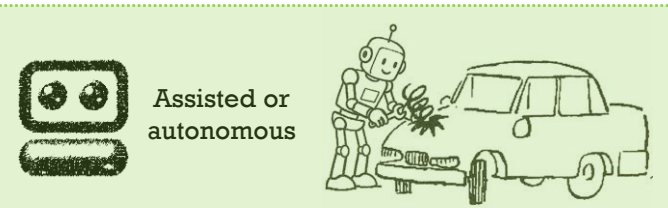


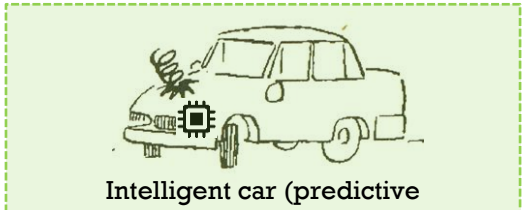






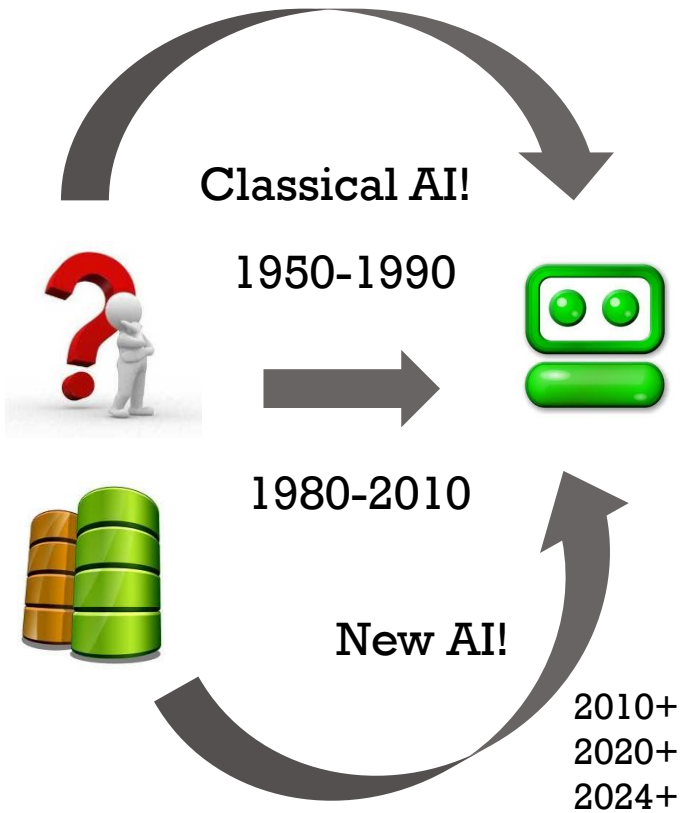






Source: Developing Industrial Case-Based Reasoning Applications, the INRECA Methodology, R Bergmann et al, Springer, 1999. Original diagrams Modified.

AI: from classical to contemporary



1970s

1997

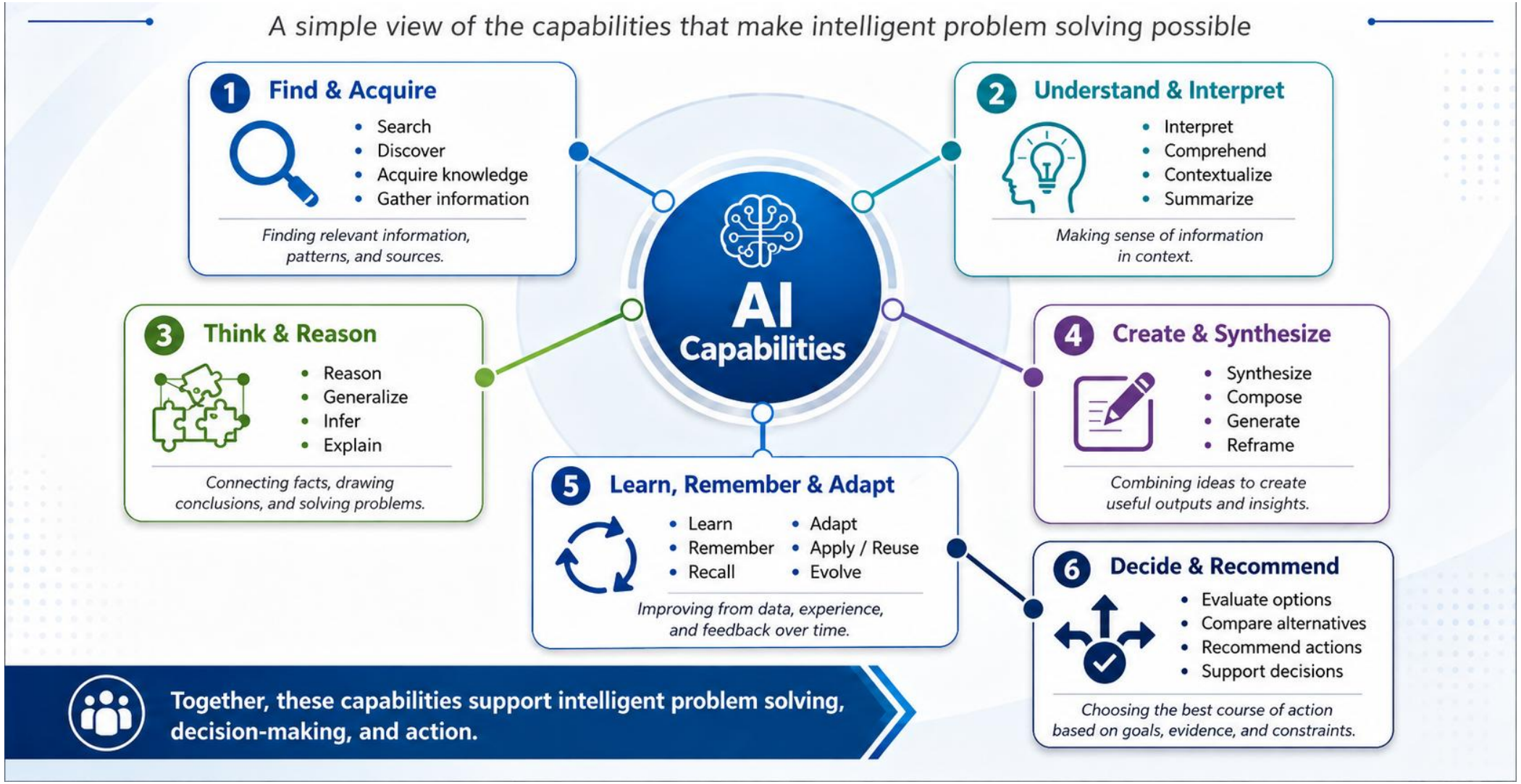
2011

2016

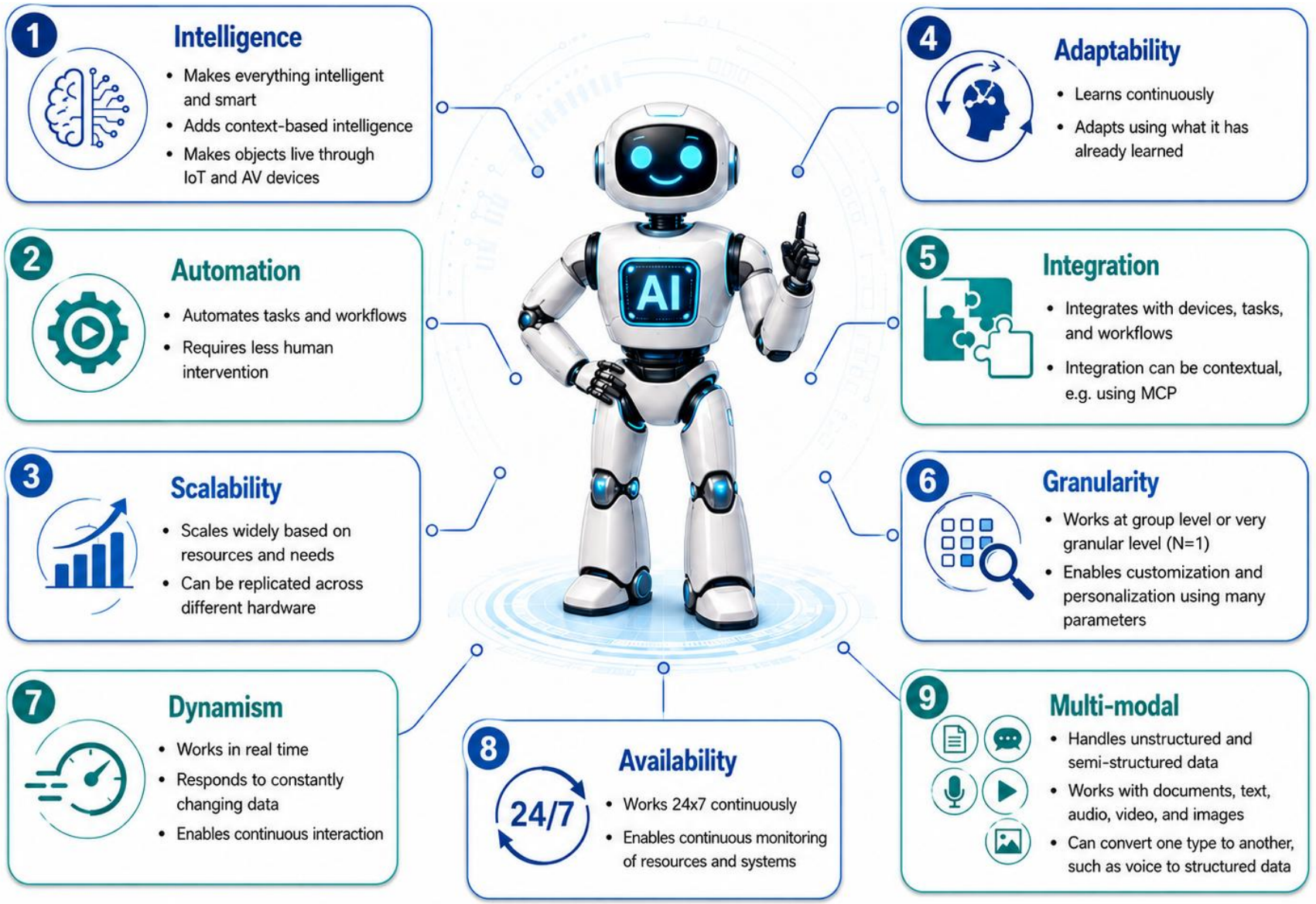
2023

LLMs 2020+ making English (or any natural language) is a new language for everything

AI: what human-like problems?



AI: why?

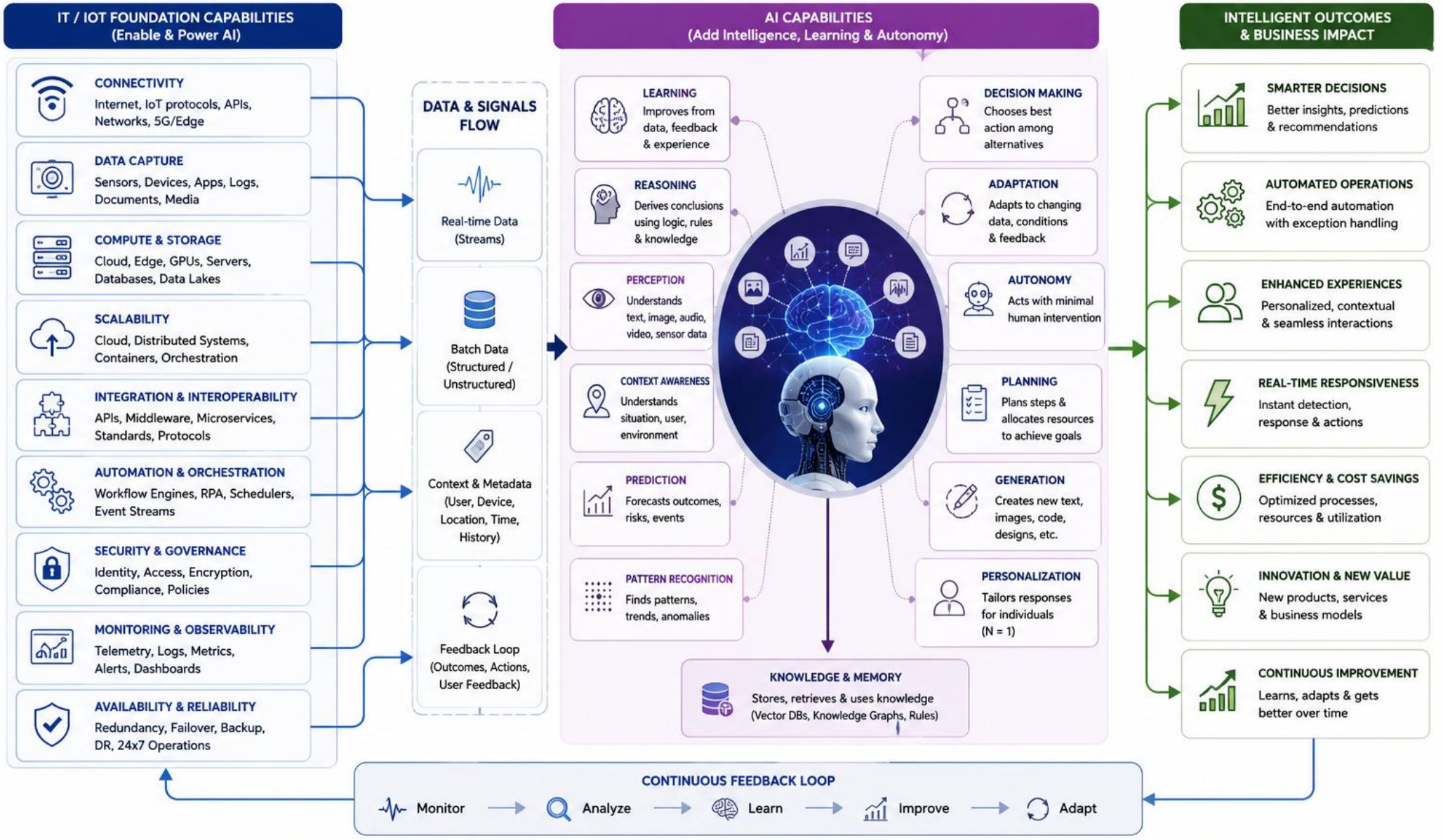


AI inherits characteristics of core IT/IOT, also enhances them

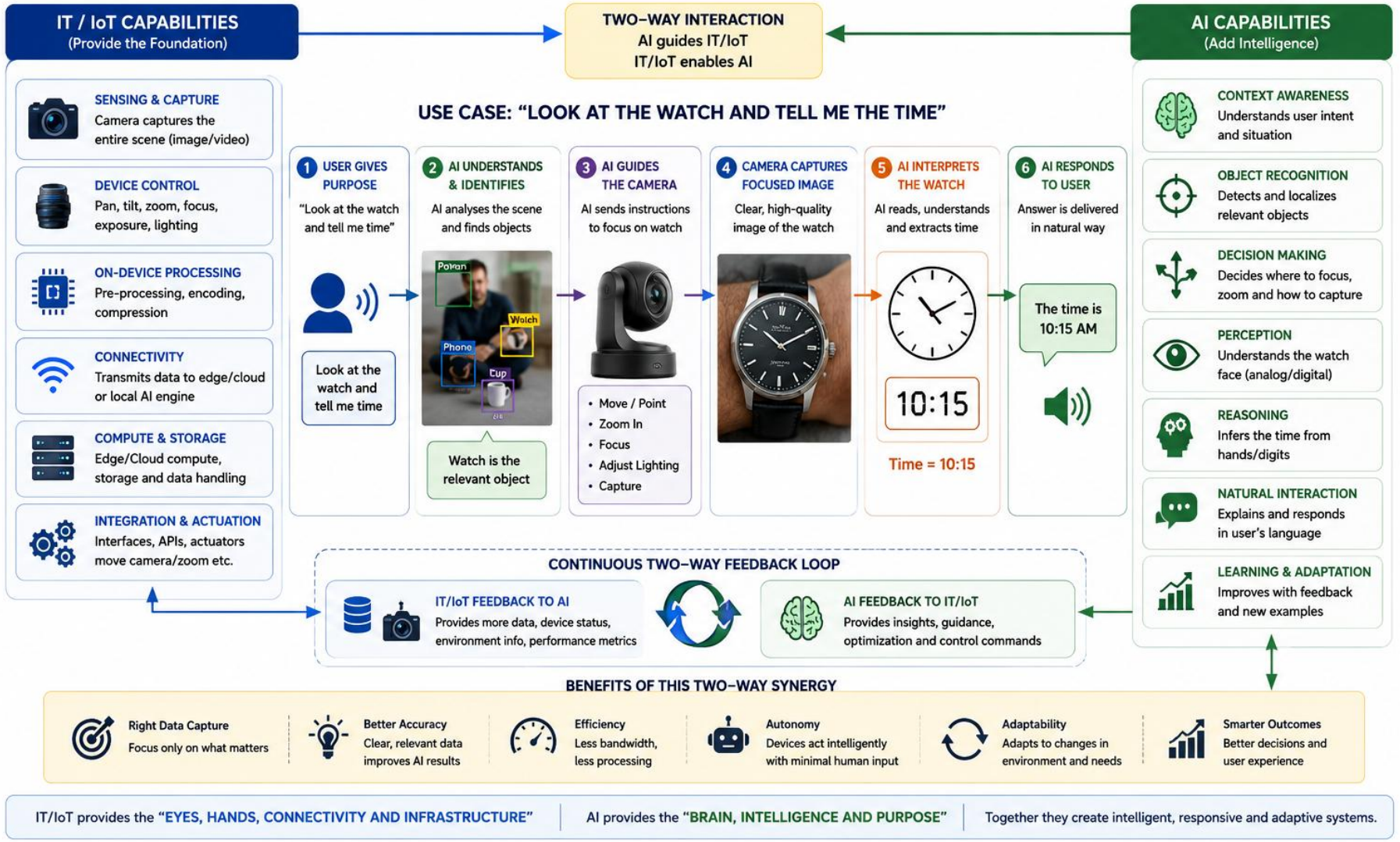
Existing characteristic	Classification
Intelligence	Primarily AI
Automation	IT capability enhanced by AI
Scalability	Infrastructure capability
Adaptability	Primarily AI
Integration	IT capability enhanced by AI
Granularity	Data/IT capability enhanced by AI
Dynamism	IT/IoT capability enhanced by AI
Availability	Infrastructure capability
Multimodality	Data-device capability interpreted by AI

(1) Machines, equipment, vehicles, etc. can have in-built (OEM) sensors and cameras or can be retrofitted to make then live (know everything about them) as well as act on intelligence provided by AI as a brain!

Connecting IT/IOT capabilities with AI capabilities

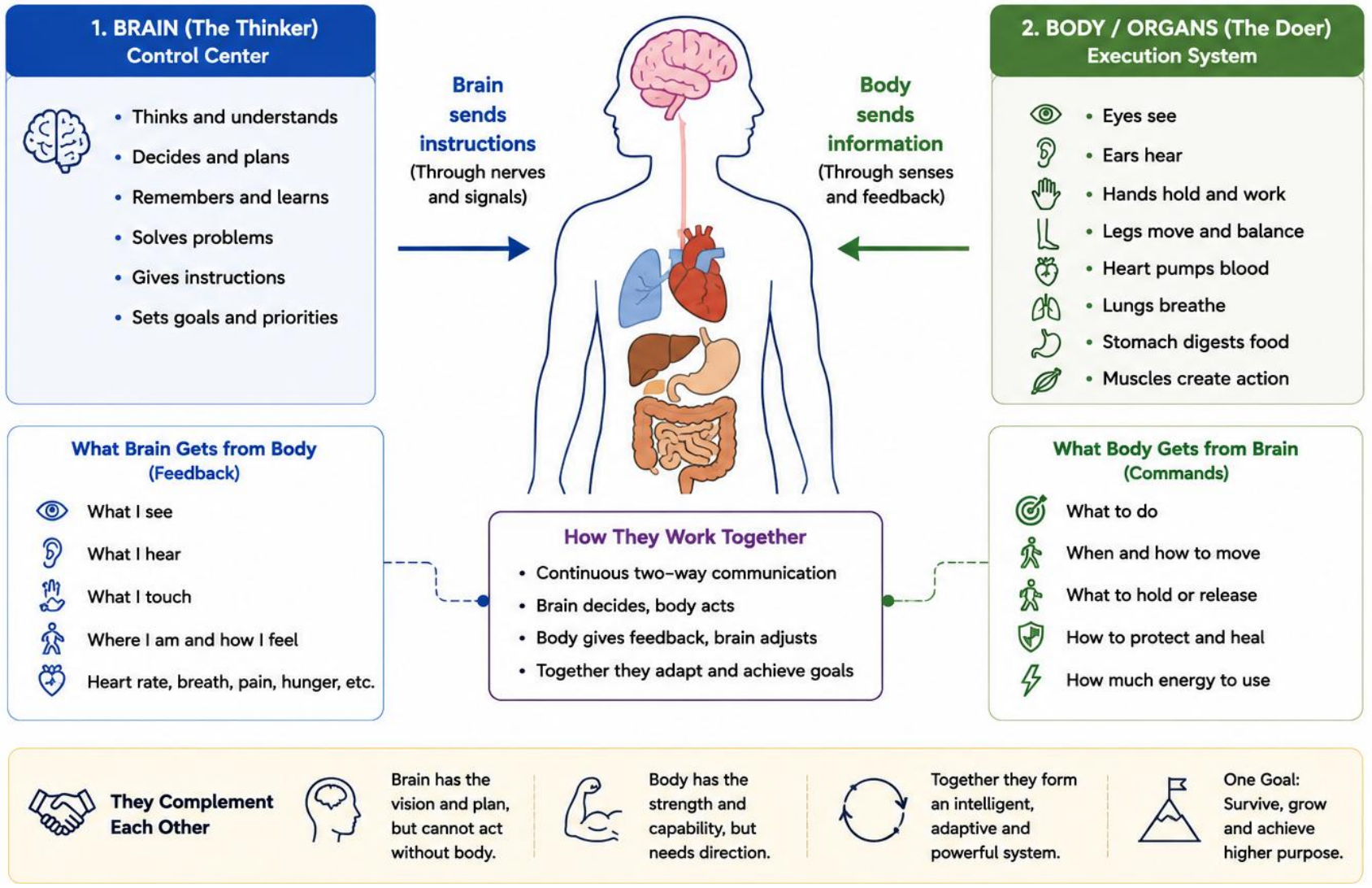


Connecting IT/IOT capabilities with AI capabilities: Example



Think about human body as IT/IOT systems and brain as AI

Two parts. One system. Same purpose: survive, adapt and achieve goals.



Characteristics largely inherited from core IT/IoT

Characteristic	Primarily inherited from	What AI adds
Automation	Software systems, workflow engines, RPA, industrial control	Intelligent selection of actions, exception handling, adaptive automation
Integration	APIs, middleware, microservices, databases, IoT protocols	Context-aware integration, semantic mapping, dynamic tool selection
Scalability	Cloud computing, distributed systems, containers, edge infrastructure	Scalable inference, parallel model execution and intelligent workload allocation
Availability	Reliable IT infrastructure, redundancy, failover, cloud services	Continuous intelligent assistance, monitoring and autonomous response
Real-time operation	Streaming systems, sensors, networks, edge computing	Real-time interpretation, prediction and decision-making
Connectivity	Internet, IoT, APIs, mobile and communication networks	Intelligent coordination among connected devices, systems and agents
Replicability	Software deployment, virtualization, containers and cloud images	Replication of trained intelligence across devices and environments
Interoperability	Standards, APIs, data formats and protocols	Semantic interoperability and understanding across heterogeneous systems
Multimodal data capture	Cameras, microphones, sensors, scanners and AV systems	Understanding and combining text, image, audio, video and sensor information
Monitoring	IoT sensors, telemetry, logs and observability platforms	Anomaly detection, diagnosis, forecasting and proactive intervention
Granularity	Databases, digital identity, sensors and transaction systems	Individual-level prediction, personalization and N=1 decision-making
Ubiquity	Embedded computing, mobile, IoT, cloud and edge systems	Intelligence embedded within devices, workflows and environments
Speed	Computing power, GPUs, networks and optimized software	Rapid inference, search, generation and decision support
Data persistence	Databases, data lakes, logs and storage systems	Learning from historical data and retrieving relevant knowledge
Security controls	Identity management, encryption, access control and network security	Intelligent threat detection, behavioural analysis and adaptive protection

Characteristics that are more distinctly AI-driven

AI-Driven Characteristic	Description
Learning	Improves performance by learning from data, feedback, examples, or experience.
Reasoning	Derives conclusions by applying logic, rules, knowledge, or learned patterns.
Prediction	Estimates future outcomes, risks, probabilities, or likely events.
Perception	Interprets text, images, speech, video, and sensor data.
Context Awareness	Understands the situation, environment, user, and circumstances surrounding a task.
Adaptation	Changes its behaviour or decisions when conditions, data, or requirements change.
Decision-Making	Evaluates alternatives and selects an appropriate action or recommendation.
Generalization	Applies learned knowledge to new, previously unseen situations.
Personalization	Tailors recommendations, responses, or actions to an individual user or case.
Autonomy	Performs tasks and takes actions with limited human intervention.
Generation	Creates new text, images, code, designs, plans, audio, or other content.
Goal-Directed Behaviour	Plans and executes a sequence of actions to achieve a defined objective.
Knowledge Representation	Organizes facts, concepts, relationships, rules, and domain knowledge in a usable form.
Problem Solving	Identifies possible solutions, evaluates them, and selects an effective approach.
Planning	Determines the steps, resources, and sequence required to achieve a goal.
Self-Correction	Detects errors, evaluates outcomes, and modifies its response or behaviour.
Natural Interaction	Communicates with people through language, speech, images, gestures, or conversation.
Pattern Recognition	Detects meaningful structures, relationships, similarities, or anomalies in data.
Creativity	Produces novel ideas, alternatives, designs, or solutions based on learned knowledge.
Explainability	Provides reasons, evidence, or justification for a recommendation or decision.

Operationalising AI: Automation of seamless end-to-end operations: Example

Hi, my customer id is 123456 and have subscribed to PlanX, I want to change to PlanY, will be grateful if you help me.

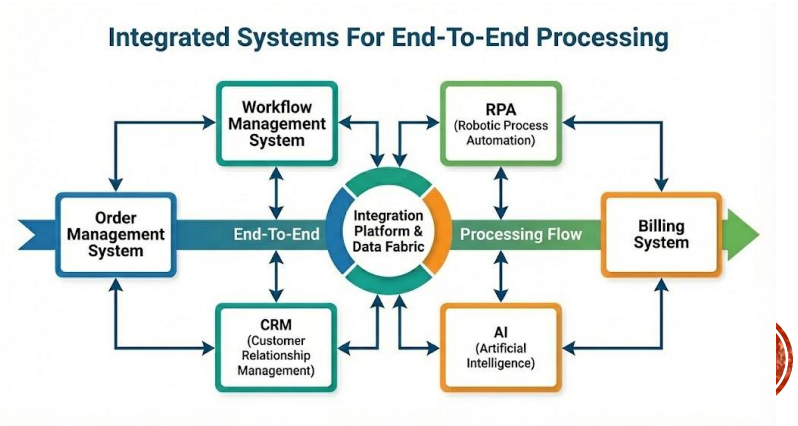
Sure, pls wait while I check how best I can help you.

1. Check credentials of customer if not logged in. If not, ask to Login in.
2. Understand what customer is looking for by extracting information from text
3. Check information provided
4. Check action, next step will be based on action, action is: change the subscription
5. Check eligibility of customer for product PlanY, ask any KYC is required, deposit etc.
6. Compliance check and risk assessment
7. If eligible, then check there any pending dues from PlanX
8. Unsubscribe PlanX and subscribe to PlanY
9. Inform to the customer

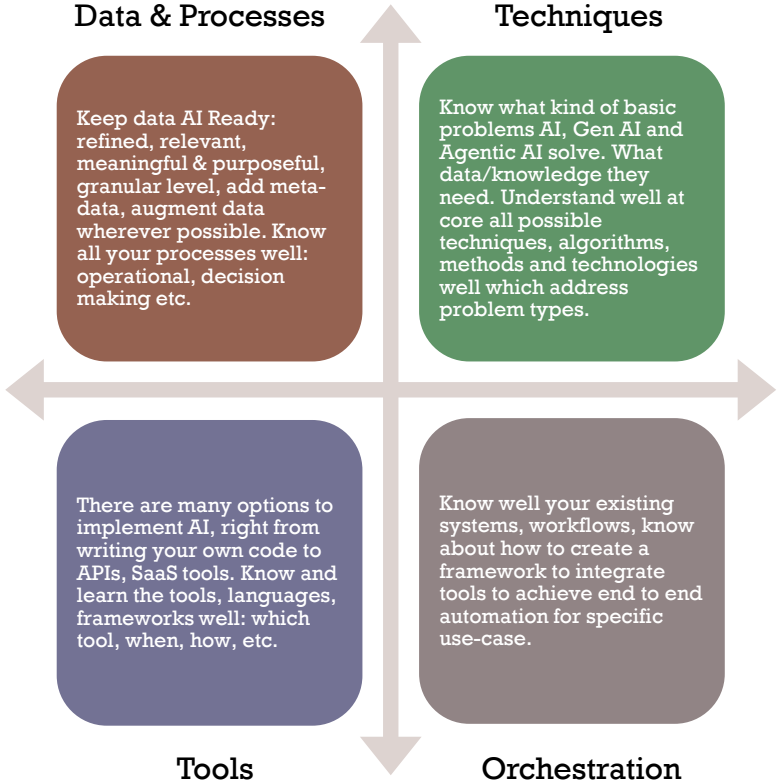
Dr Akshay, we have successfully changed your plan from PlanX to PlanY. Appreciate your feedback on our service [Link]

10. Get feedback

1. Operational task.
2. Gen AI Task: NLP: Named Entity Recognition with action. Decision Making: what needs to be done to change subscription. Ask missing information if not able to decide.
3. Enterprise systems: subscription management: check existing subscription
4. Decision task
5. AI + Operational Tasks: CRM system: KYC status, order management system: compliance, requirements (documents). Invoke workflow system, RPA.
6. Gen AI Task: Compliance check and risk assessment
7. Billing system: Status of payments etc. If dues, invoke payment processing system
8. Order management and billing system: product configuration and subscription
9. Workflow management
10. CRM System: Get feedback



AI: why and what is readiness?



- ❑ Have your digitalization done and have meta-data about everything.
- ❑ Get your data AI-ready in one place or create AI ready data integration layer.
- ❑ Create metadata about data sources, define access policy and data governance (who should access what etc.)
- ❑ Invest in and build teams: ML/AI Engineers, data scientists and developers. Have formal training programs and certifications for them.
- ❑ Have business, domain and tech guys sit together for use-cases.
- ❑ Let the team experiment, build and test models
- ❑ Make AI horizontal, create complete repository of models and share amongst the teams to increase reusability, improve results, reduce duplication of efforts, etc.

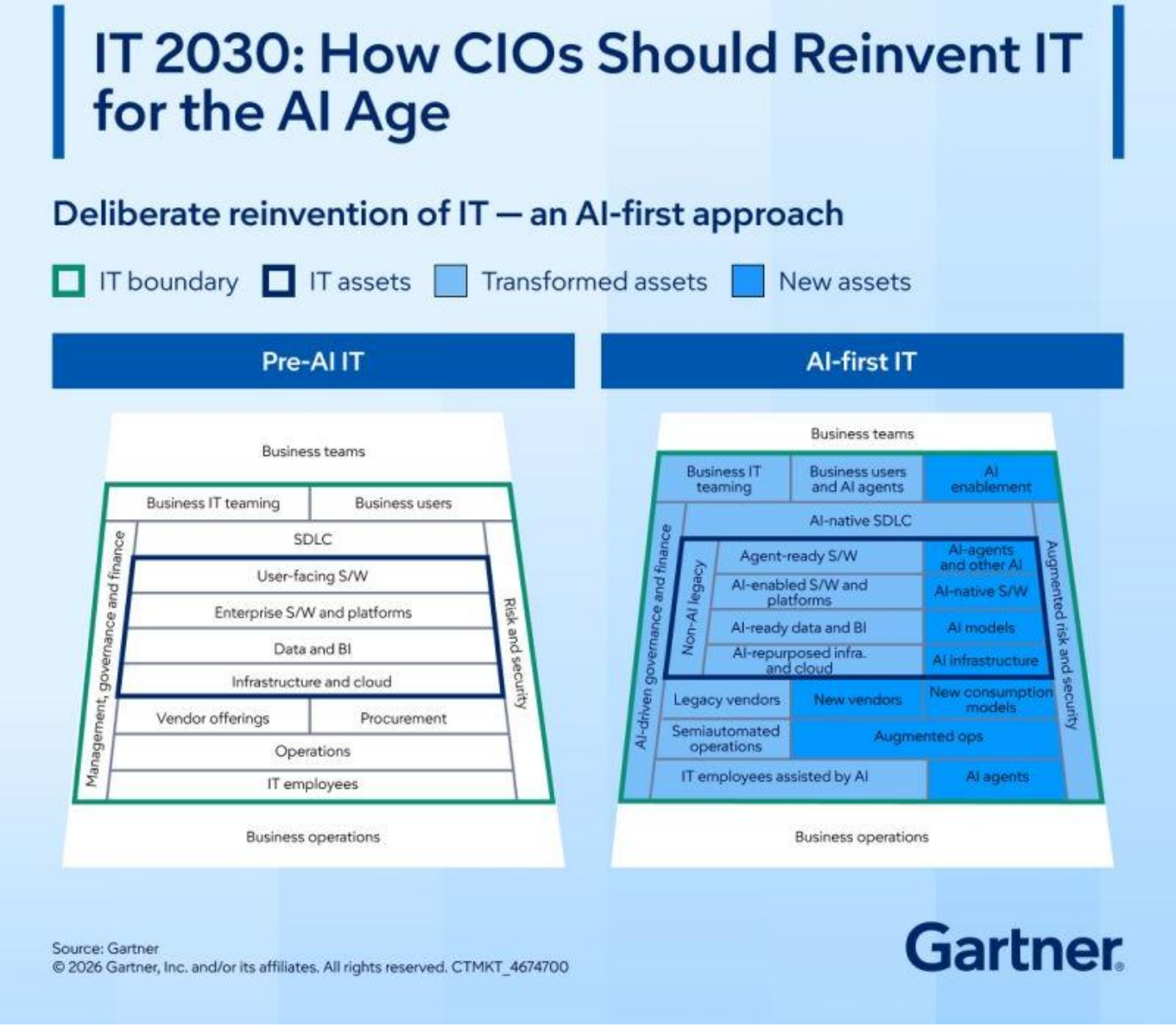
Excerpts from case study: DBS's AI Journey Piyush Gupta, CEO, DBS

Lim drew from Sun Tzu's Art of War, emphasizing the significance of leadership and cultivating the right mindset throughout the organization in order to survive changes and thrive afterward. Having a clear purpose helped DBS overcome various obstacles in developing the right capabilities to compete and deliver value in an increasingly digital world. Shee highlighted that instead of "putting on digital lipstick," or chasing after superficial changes, DBS "integrated digitalization into its DNA" and differentiated itself from competitors.

Him Chuan LIM is their country head, Shee Tse Koon is Group Executive and Group Head of Consumer Banking and Wealth Management

Why Should a Bank Think Like an E-commerce Company

AI first approach?



1. AI is core to the business model
Products and services are built on AI capabilities, not just enhanced by them. Revenue and competitive edge depend on AI performance.

2. Decision-making is AI-augmented
Strategic, operational, and customer-facing decisions leverage AI insights. Leaders treat AI outputs as first-class inputs in planning and execution.

3. Workflows are redesigned around AI
Processes are re-engineered so AI can operate at the center; removing human bottlenecks, not just automating existing steps.

4. Data is a strategic asset
Make data models AI ready. The company actively designs systems to collect, structure, and own proprietary, high-quality data to feed models.

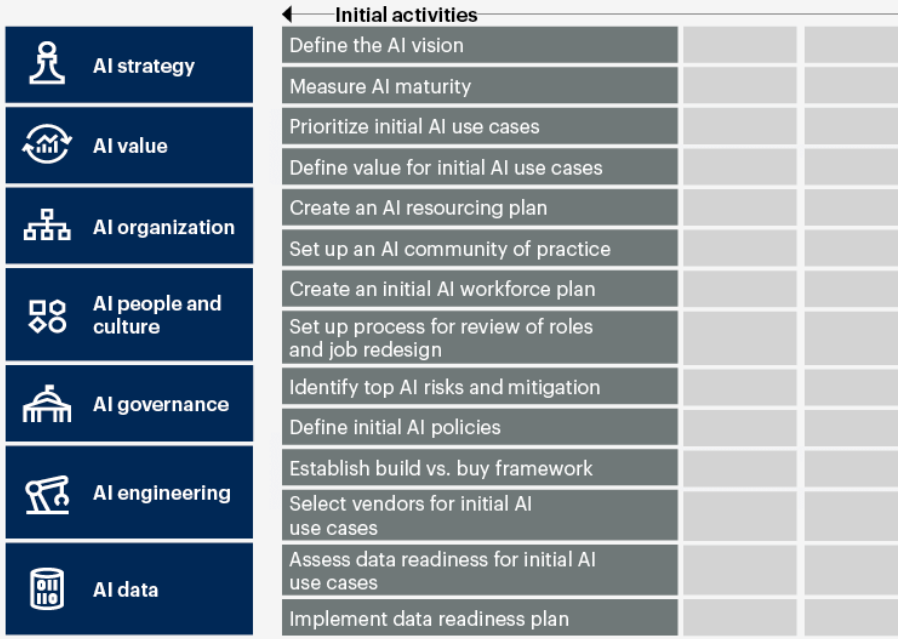
5. Culture is "AI-literate"
Everyone, from executives to front-line staff, understands AI's capabilities, limits, and ethical implications.

what-is-an-ai-first-company

- Understand
  - how intelligence is built!
  - what are the roles of data driven classical AI, semantic AI, Gen AI and agentic AI
  - characteristics of AI
- Think of automating everything
- Do everything in proper English meaningful and purposeful naming convention e.g. artifacts in code, data tables, fields, etc.
- Think data and knowledge sources as backbone of building intelligence
- Think about every piece of data, process, resource and artifact, make it useful and contextual, add value to it by metadata

AI Roadmap?

Gartner AI Roadmap at a Glance



Source: Gartner

Enterprise AI Capabilities for Maturity Model - Tier 1 and 2

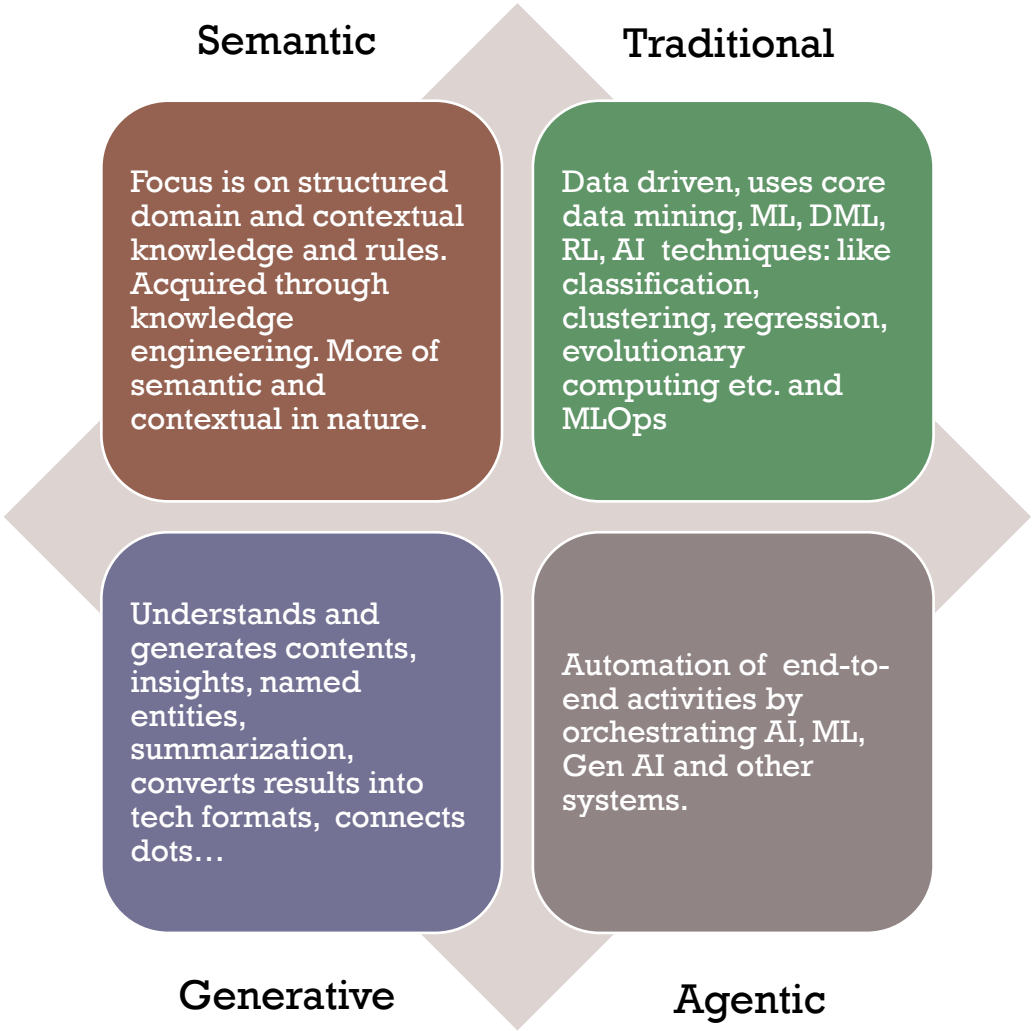
	AI strategy	AI value	AI organization	AI people & culture	AI governance	AI engineering	AI data
Tier 1	Monitor and interpret AI trends	Develop and manage AI use case portfolio	Develop external partnerships and alliances for AI	Manage culture and change for / due to AI	Create, maintain, and evolve AI, governance policies and controls	Design and architect AI applications and systems	Acquire and prepare AI data
	Develop and rationalize AI ambition	Develop AI value propositions and business cases	Manage and balance internal and external ecosystem	Evolve roles and staffing for / due to AI	Develop AI governance operating models and teams	Develop, test and integrate AI applications and systems	Manage and govern AI data
Tier 2	Develop and refine AI strategy	Develop AI product portfolios	Evolve internal operating model	Deliver skills, training, and literacy for AI	Engineer oversight systems and enforce policies	Deploy, operate and scale AI systems	Build AI data utilization and insights
	Develop and coordinate AI strategic roadmap					Manage project, platform and application portfolios	
	Innovate / incubate AI					Develop and manage AI engineering teams	

Source: Gartner
821273_C

AI leadership is different than other tech leadership. It needs complete rethinking and redesigning to great extent, challenges: un-deterministic nature of GenAI (more risks!), non explainability of techniques like neural networks (decision outcomes!), connects to other techs, data driven hence quality and AI ready data and its governance is very important, responsible and ethical use of data and models, etc.

Gartner

Understanding: semantic AI, traditional AI, Gen AI, Agentic AI



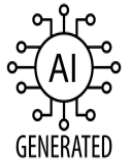
Semantic

Build domain knowledge and knowledge driven systems whenever applicable powered by Gen AI. Semantic AI can address core AI problem types. Get the knowledge validated and verified by domain experts. Semantics. Plays role in managing optimizing and improving traditional AI. It can help to build AI systems without data to great extent (cold-start issues) and common-sense knowledge.

Traditional

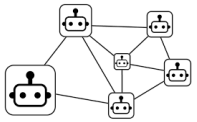
Implements core AI functionalities and models generic problem types. Based on the use-case tasks such as sentiment analysis, recommendations, inventory forecasting using use traditional AI to build ML models. Use metadata to improve accuracy of models. Create a complete repository of models with every minute details: what task, what data used, which techniques experimented, tools/code used, results, accuracy etc. Optimize and make models easily accessible using APIs. Use local NLP, pretrained models to optimize performance and save the cost.

Generative



Use Gen AI for NLP, connecting dots, reasoning tasks, vibe-coding and build multi-modal interfaces especially user facing. Use CS concepts like caching, local sentence transformation to save prompts going to Gen AI every time. A complete shift from building own to creating, generating, operating, managing … using just prompts (text-voice-vision)!

Agentic



Once the all tasks are implemented (may be using AI agents) and other systems are tuned and synchronized to talk to each other, use orchestration framework to have end-to-end automation.

My advice to get best out of AI

What your tech team should learn, know and understand in sequence

IT/CS Fundamentals and Tech Understanding

- Have understanding other techs like IoT, Cloud, AR/VR etc.
- What is called as digital? digitization, digitalization and digital transformation, compute, memory, storage, network
- What is software, apps, software architectures and engineering, workflows, APIs, micro-services, caching, local processing etc.

Data Essentials

- Good data is foundation of good AI. What are the different data formats (CSV/JSON), preprocessing, granular v/s aggregated data, type of data etc.
- What is data management and data schema/vocabulary, SQL
- Meta-data and its role. What is mean by semantics in data and role of domain knowledge.

Basic Analytics/Statistics

- Understand what is descriptive statistics and how to use it
- Understand common-sense, proactive, predictive and prescriptive intelligence, role of visualization, etc.

Core AI

- Core AI/ML/Data mining have been there since more than 3 decades. They still matters.
- Understand what type of core and generic problems AI/ML solve
- What are the generic applications AI addresses.
- Learn and understand what tools, techniques, methods and techs are used, how to model them, what problems they solve, what kind of data/input they need, what are limitations and issues with each of them and so on.

Generative AI

- Understand capabilities of Gen AI, issues and limitations.
- Learn and master science and art of prompt engineering and context engineering. They are is a must.
- Understand role of RAG, fine tuning, vector databases, etc. to build Gen AI backed applications. How to optimize Gen AI backed apps from performance and cost perspective.
- Note, Gen AI cannot not solve all AI problems, not expected to be, it solves specific categories of problems. It takes care of context, semantics and understanding of multi-modal data, helps to understand, integrate various components including core AI.
- Tools are just peripherals and integrate various APIs, micro-services, functions and components, need to spend time on implementing and put them in use-case/application context.

Tools, Languages, Frameworks and Platforms

- There are many tools, frameworks and platforms exists
- Understand them based on what use-case you are going to implement
- Python is one of the widely used language for core AI/ML.
- Be tool agnostic, learn science and art of orchestrating the tools.

Agentic AI | Making Existing Systems Smart

- Understand what is end-to-end automation and workflows
- Agentic AI automates workflows and Gen AI plays very crucial role in implementing cognitive and thinking tasks and integrating tasks in workflows
- Learn tools to implement agentic AI. Better you develop expertise in-house.

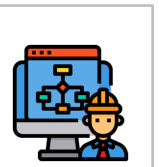
Relevant CS Fundamentals-> AI Ready Data -> AI Fundamentals -> Design, Development & Implementation

Combining AI and human cognition

As AI absorbs tasks related to retrieval and summarization, the locus of human contribution shifts upward towards judgement, direction, expertise, and synthesis. AI is best understood as a powerful ship rather than an independent navigator: it can move faster and farther than any human, but without a *knowledgeable captain* who understands the vessel and the waters they are navigating, it is as likely to drift aimlessly as it is to arrive anywhere useful.

Cognitive workers must therefore supply *deep subject-matter* understanding to frame the right questions, identify meaningful trade-offs, and evaluate outputs critically.

AI *amplifies* the returns to prior knowledge and users who read widely and deeply are better able to steer models and push analysis beyond surface-level synthesis.



Cognitive workers must act as *system architects* rather than task executors. Productive AI use depends on the ability of the human to decompose complex problems, sequence inquiries, impose constraints, and define evaluation criteria. This is less about prompt engineering and more about structured thinking.

AI does not diminish the importance of cognitive workers, but rather it raises the threshold for what is considered *meaningful contribution*. For policymakers, this shift implies that skilling strategies must prioritize domain depth, analytical reasoning, structured problem solving, and continuous learning from an early age.

Economic Survey 2025-26, Government of India Ministry of Finance Department of Economic Affairs

Let us define Intelligence

Intelligence is the ability to acquire, understand, and apply knowledge and skills to solve problems, adapt to new situations, reason, and make decisions.

What is Intelligence?

Intelligence is the ability to acquire, understand, and apply knowledge and skills to solve problems, adapt to new situations, reason, and make decisions.

Acquire → Learning

Memory
Adaptability

Tacit | Explicit | Implicit

Apply

Problem

Think, plan & reason

Solution

	Aspect	Description
1	Learning	The ability to acquire and retain knowledge from experience.
2	Reasoning & Problem-Solving	Applying logic and strategies to solve problems.
3	Adaptability	Adjusting to new situations and environments.
4	Memory	Storing and retrieving information when needed.
5	Creativity	Generating novel ideas and solutions.
6	Social Intelligence	Understanding and interacting with others.
7	Self-Awareness	Recognizing one's own thoughts and emotions (in humans and potentially in AI).

Collective and collaborative Intelligence

Reference: what is intelligence?

Making machines intelligent: Building Intelligence.

What we are going to learn: understand the science, art and engineering of building intelligence.

Acquire → Learning

Memory
Adaptability

Tacit | Explicit | Implicit

Apply

Think, plan & reason

Problem → Solution

Knowledge Engineering (acquisition & codification)

- What are the ways to make machines learn and derive the knowledge?
- Can the machines be fed explicit knowledge?
- What are the data sources required to learn the knowledge?
- Data sources: what are those?, different types, formats, challenges and issues etc.
- What are the techniques, methodologies and technologies used to acquire the knowledge (machine learning)?

knowledge?

- What is mean by tacit, explicit and implicit knowledge and how it is presented, stored, shared, transferred, new knowledge can be added to existing one etc.?
- What is accuracy of knowledge when it used to solve the problem?
- Can the derived knowledge be verified, corrected by humans in loop?
- Can the knowledge stored can be interpreted, explainable, etc. ?

Application of knowledge

- How knowledge acquired can be applied to a given problem to derive the solution?
- What are the formats in which inputs are given and solution outputs are derived or stored?
- What are the tools, techniques, methodologies and technologies used to access and use the knowledge to solve the problem in the current problem context?
- How machines think, plan and reason the problem solving?
- Can the machines explain how they solved problems, derived solutions and explain solutions too?
- Can machine answer the questions related to solutions?
- Can humans take action using solutions suggested by the machines?

Automation of problem solving

- Can the process of knowledge acquisition, management (representation, storage etc.) and application of knowledge be automated end-to-end?
- In practical scenario, how entire problem can be broken into smaller tasks and each task is executed whenever it is required? output of one task can be fed to input to other tasks. How one major task can control/coordinate/orchestrate other tasks?…tasks can be classified as operational tasks, decision tasks, intelligent tasks, etc.
- Can the machines solve end to end problem? Solving all tasks, a task can be operational task or intelligent task? How this can be made possible?

AI tools, techniques and technologies to make machines intelligent and do end-to-end automation

Common-sense and operational intelligence

Acquire → Learning

Memory
Adaptability

Tacit | Explicit | Implicit

Apply

Think, plan & reason

Problem → Solution

25-30% use cases can be solved by just incorporating common-sense and human domain intelligence without AI. This was exactly the focus of semantic AI, capturing, structuring and automating precious human expertise .

Using simple statistics to know customer transaction pattern

Max amount: 50K
Avg amount: 10K
Upfront payment: 1Lakh
Min time bet transactions: 1Hour
Type: Online:80% offline:20%
Locations:[Mumbai, Delhi…]
…

Cold start problem, use stat of demographic/other (wisdom of crowd)

Type, credit limit: Salaried, 5L
Max amount: 70K
Avg amount: 15K
Upfront payment: 1.5Lakh
Min time bet transactions : 1Hour
Type: Online:80% offline:20%
…

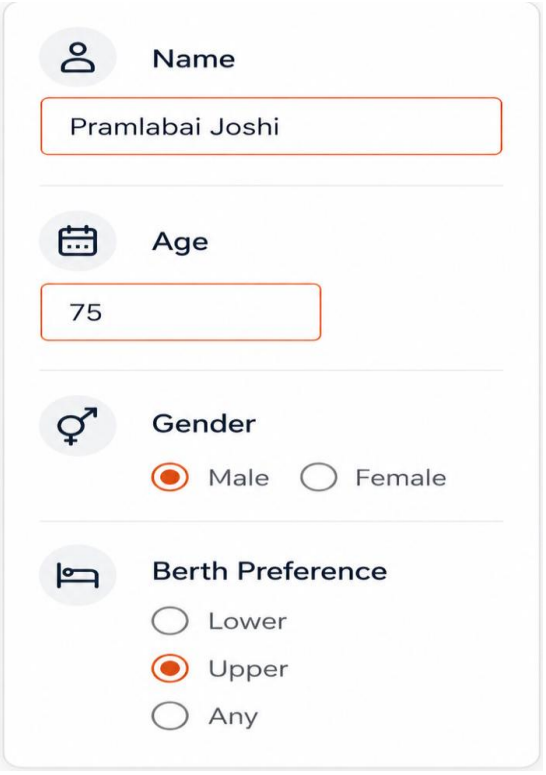
This can help whether 1> a call should be given to the customer or not for *EMI option* based on transaction done, 2> to certain extent, detecting whether transaction is likely fraud (outlier) or not, 3> if it is usage tracking like electricity or gas, can help whether something is gone wrong etc.

Lot of behavioural aspects can be determined from transactions and interactions (through user interfaces e.g. average time taken to enter password by the user), improve customer experience, validate the details etc. Metadata about data can help to categorize, understand things and their relationships better. Relationships help to restrict data entry and dynamic validation of data.

Examples:

- ✓ Cross check usage data for outliers e.g. utility bills
- ✓ Once kind of transaction is authorized remember kinds! e.g. money transfer to same beneficiary
- ✓ Check customer savings balance before making offer for personal or credit card loans!
- ✓ Push last accessed options/filled data on the top reduces search time.
- ✓ Auto populate whatever data available
- ✓ …

Semantic validation of data before submit



Understand every data piece and semantic relationships between them for correctness. Common sense tells lower berth is better for elderly people! Check names with gender selected, etc.

Intelligence: Predictive intelligence using simple explanation using regression

Example: data driven predictive intelligence using simple linear regression, predict sales based on ad spent

Acquire → Learning

Memory
Adaptability

Tacit | Explicit | Implicit

Apply

Think, plan & reason

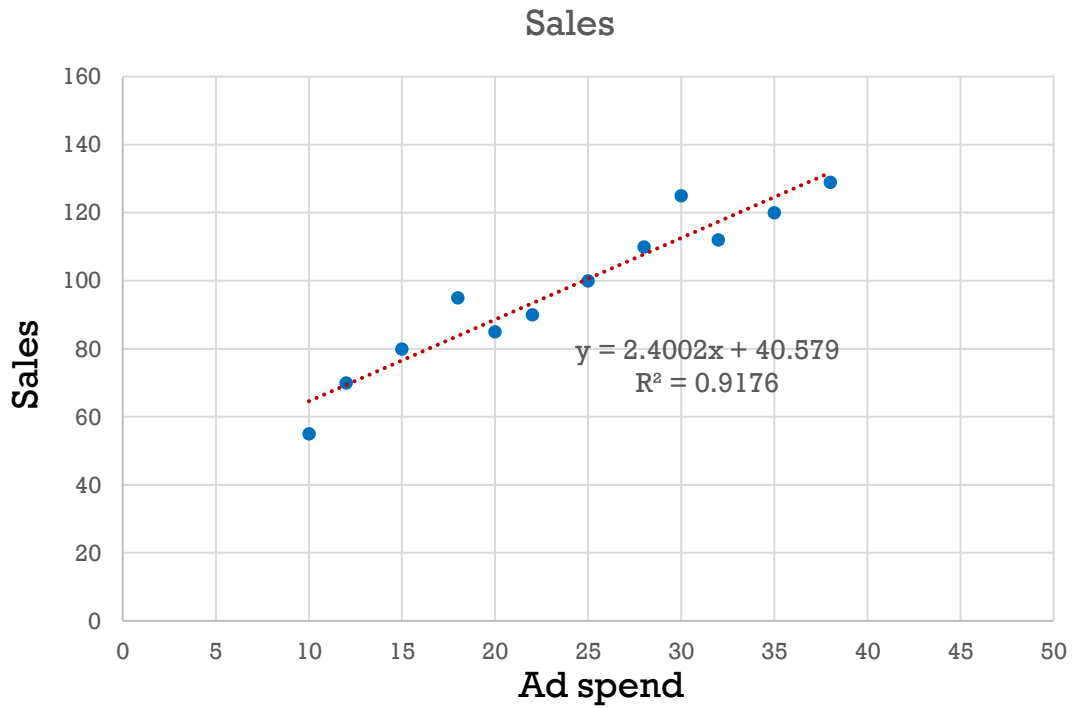
Problem → Solution

Data set: 15 months ad spend and sales data. Divide data into 2 sets, training: 12 months and testing: 3 months. Build model **y=mx+c** using 12 months data.

No	Ad Spend	Sales
1	10	55
2	12	70
3	15	80
4	18	95
5	20	85
6	22	90
7	25	100
8	28	110
9	30	125
10	32	112
11	35	120
12	38	129
13	40	133
14	42	140
15	45	150

Training Data

Testing Data



Equation **y=mx+c** is **acquired** using mathematical formula which minimizes difference between actual and calculated and determine value of x and c. R^2 shows how good the model is.

Model (**knowledge**):
Sales = 2.4xAdd spend + 40.579

Intelligence: Predictive intelligence using simple explanation using regression

Acquire → Learning

Memory
Adaptability

Tacit | Explicit | Implicit

Apply

Think, plan & reason

Problem → Solution

*For every 1 unit increase in ad spend, sales increase by approximately 2.4 units, based on the training data. *If I spend Rs.50 on ads how much of sales I am likely to achieve?*

Predicting data for training set and testing the model using test data set

No	Ad Spend	Sales	Predicted
1	10	55	65
2	12	70	69
3	15	80	77
4	18	95	84
5	20	85	89
6	22	90	93
7	25	100	101
8	28	110	108
9	30	125	113
10	32	112	117
11	35	120	125
12	38	129	132
13	40	133	137
14	42	140	141
15	45	150	149

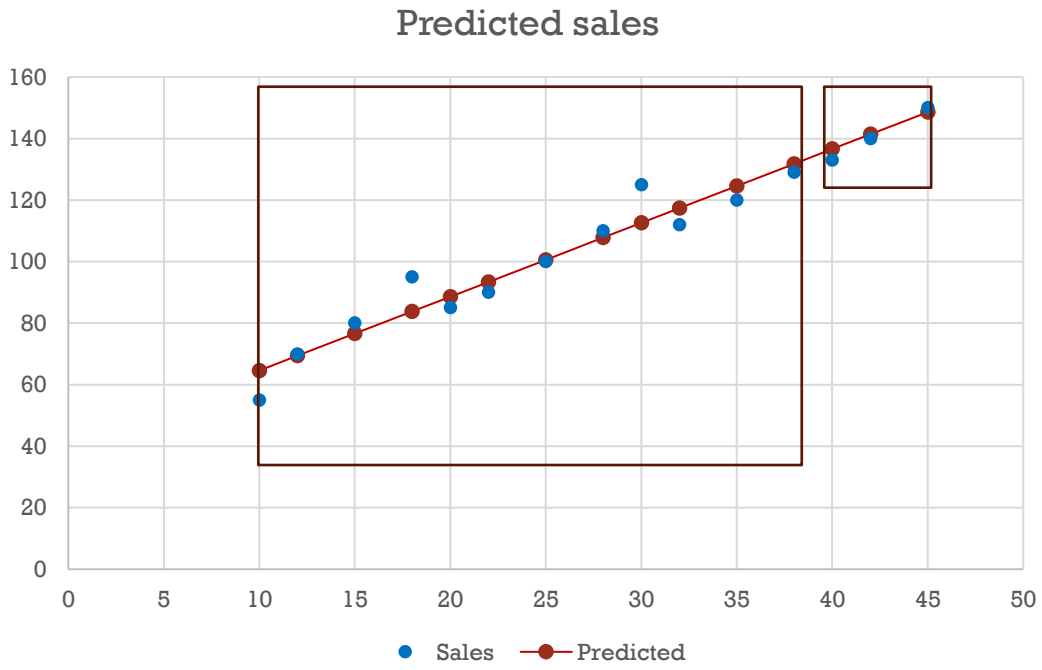
Applying knowledge* to predict sales for ad spend

16	50		161
17	55		173
18	60		185

Solution is predicted sales.

Trained model can be automated through API to know likely sales whenever ad spend is entered/provided. A code can be added to know what ad spend will be required to achieve specific sales. This model can be plugged in and invoked whenever required in agentic systems using API. Even API can be written to retrain model whenever new data is added (AutoML).

Predicted sales=2.4 x Add spend + 40.579



Ad Spend	Sales	Predicted	Squared error
40	133	137	13
42	140	141	2
45	150	149	2
		MSE	6

Model looks good for testing data too as error is 6 (*pls note predicted values and squared errors are rounded*)

Intelligence: Predictive intelligence using simple explanation using regression

Acquire → Learning

Memory
Adaptability

Tacit | Explicit | Implicit

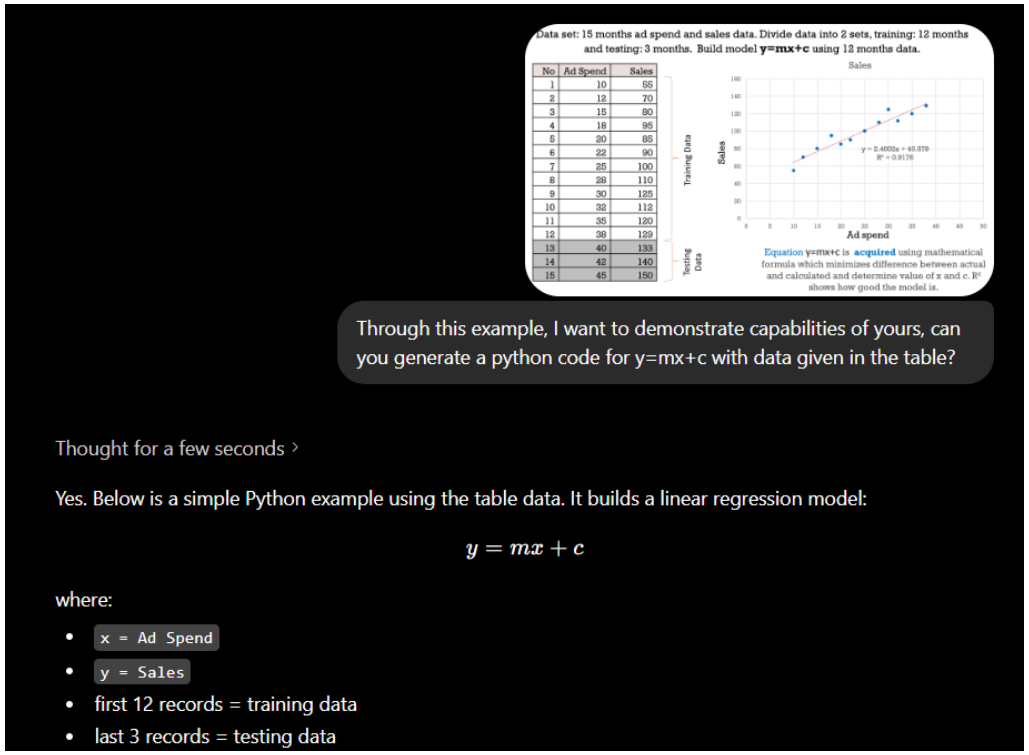
Apply

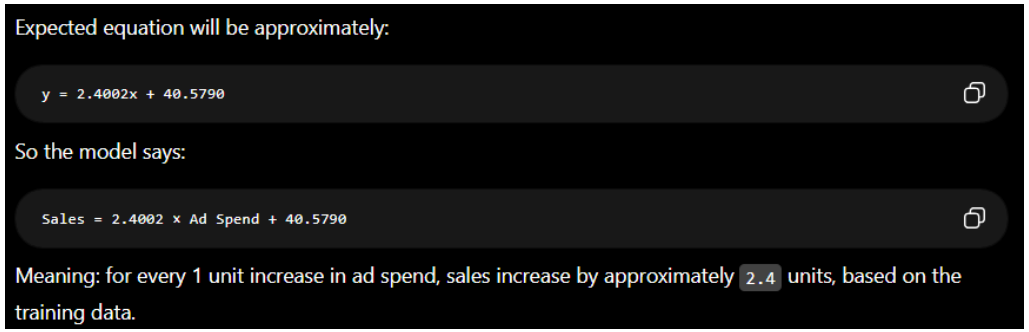
Think, plan & reason

Problem → Solution

Make learning personalized and interesting. Complexity can be simplified.

Using Gen AI to understand and code! ChatGPT outcome





https://chatgpt.com/share/6a23f04d-5f00-83aa-919d-0ce8a519eff5

Intelligence: Predictive intelligence using simple explanation using regression

Using Gen AI to code!

Acquire → Learning

Memory
Adaptability

Tacit | Explicit | Implicit

Apply

Think, plan & reason

Problem → Solution

```python
import numpy as np
import matplotlib.pyplot as plt
from sklearn.linear_model import LinearRegression
from sklearn.metrics import r2_score, mean_absolute_error, mean_squared_error

# ------------------------------------------------
# 1. Dataset
# ------------------------------------------------

ad_spend = np.array([10, 12, 15, 18, 20, 22, 25, 28, 30, 32, 35, 38, 40, 42, 45])
sales = np.array([55, 70, 80, 95, 85, 90, 100, 110, 125, 112, 120, 129, 133, 140, 150])

# ------------------------------------------------
# 2. Split into Training and Testing Data
#    First 12 months = training
#    Last 3 months = testing
# ------------------------------------------------

x_train = ad_spend[:12].reshape(-1, 1)
y_train = sales[:12]

x_test = ad_spend[12:].reshape(-1, 1)
y_test = sales[12:]

# ------------------------------------------------
# 3. Build Linear Regression Model: y = mx + c
# ------------------------------------------------

model = LinearRegression()
model.fit(x_train, y_train)

# Get m and c
m = model.coef_[0]
c = model.intercept_

print("Linear Regression Equation:")
print(f"y = {m:.4f}x + {c:.4f}")
```

```python
# ------------------------------------------------
# 4. Predict Sales for Training and Testing Data
# ------------------------------------------------

y_train_pred = model.predict(x_train)
y_test_pred = model.predict(x_test)

print("\nTesting Data Predictions:")
for ad, actual, predicted in zip(x_test.flatten(), y_test, y_test_pred):
    print(f"Ad Spend: {ad}, Actual Sales: {actual}, Predicted Sales: {predicted:.2f}")

# ------------------------------------------------
# 5. Model Evaluation
# ------------------------------------------------

r2_train = r2_score(y_train, y_train_pred)
r2_test = r2_score(y_test, y_test_pred)

mae_test = mean_absolute_error(y_test, y_test_pred)
mse_test = mean_squared_error(y_test, y_test_pred)
rmse_test = np.sqrt(mse_test)

print("\nModel Performance:")
print(f"Training R² Score: {r2_train:.4f}")
print(f"Testing R² Score: {r2_test:.4f}")
print(f"Testing MAE: {mae_test:.2f}")
print(f"Testing RMSE: {rmse_test:.2f}")

# ------------------------------------------------
# 6. Plot Training Data, Testing Data and Regression Line
# ------------------------------------------------

plt.figure(figsize=(9, 5))

plt.scatter(x_train, y_train, label="Training Data")
plt.scatter(x_test, y_test, label="Testing Data")
```

```python
# Regression line
x_line = np.linspace(min(ad_spend), max(ad_spend), 100).reshape(-1, 1)
y_line = model.predict(x_line)

plt.plot(x_line, y_line, linestyle="--", label=f"Regression Line: y = {m:.2f}x + {c:.2f}")

plt.xlabel("Ad Spend")
plt.ylabel("Sales")
plt.title("Linear Regression: Ad Spend vs Sales")
plt.legend()
plt.grid(True)

plt.show()
```

Intelligence: Predictive intelligence using simple explanation using regression

Using Gen AI to code!

Acquire → Learning

Memory
Adaptability

Tacit | Explicit | Implicit

Apply

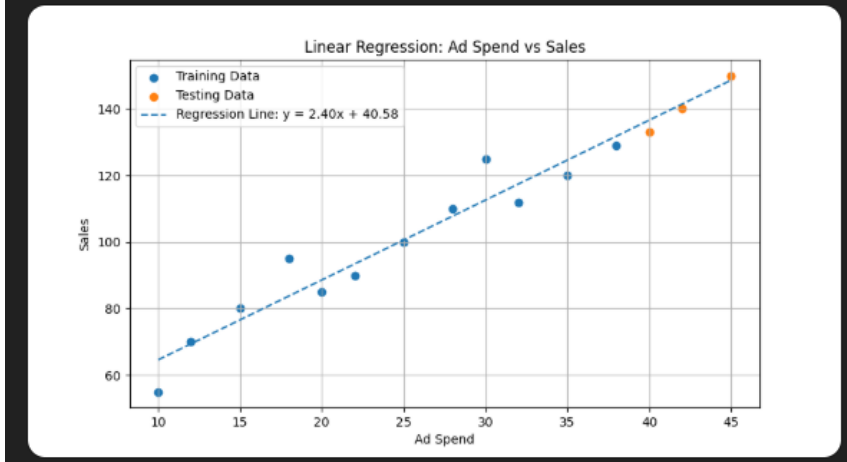
Think, plan & reason

Problem → Solution

```
Linear Regression Equation:
y = 2.4002x + 40.5794

Testing Data Predictions:
Ad Spend: 40, Actual Sales: 133, Predicted Sales: 136.59
Ad Spend: 42, Actual Sales: 140, Predicted Sales: 141.39
Ad Spend: 45, Actual Sales: 150, Predicted Sales: 148.59

Model Performance:
Training R² Score: 0.9176
Testing R² Score: 0.8851
Testing MAE: 2.13
Testing RMSE: 2.36
```



make prediction for 50,55 and 60.

Thought for a second ›

Using the model:

$$y = 2.4002x + 40.579$$

Predictions:

Ad Spend	Predicted Sales
50	160.59
55	172.59
60	184.59

Add this code after model training:

```python
# ------------------------------------------------
# 7. Predict for new Ad Spend values
# ------------------------------------------------

new_ad_spend = np.array([50, 55, 60]).reshape(-1, 1)

new_predictions = model.predict(new_ad_spend)

print("\nNew Predictions:")
for ad, predicted in zip(new_ad_spend.flatten(), new_predictions):
    print(f"Ad Spend: {ad}, Predicted Sales: {predicted:.2f}")
```

Intelligence about?

Acquire → Learning

Memory
Adaptability

Tacit | Explicit | Implicit

Apply

Think, plan & reason

Problem → Solution

Intelligence?

Knowledge

Ability to acquire and apply knowledge

Apply to?

Take actions

Manual | Assisted | Autonomous

?

Add?

Manuals
Process documents
Web Sources
Feedback and reviews
Emails
GenAI

Knowledge About?

Acquire by what means?

Customers
Suppliers/Vendors
Products/Services
Employees
Resources/Assets
Processes
Market

People
Automated (using machines)
Hybrid

Acquire: type of sources?

Tacit
Implicit/ explicit

about?

Data source

?

Reports
Queries
OLAP

Internal operational & external systems

DW/DM/ DL

Training
Chat

IoTs | Metadata | BI/BA | LLMs | RPA | AI, ML, DM… | …

Knowledge acquisition/basic learning types

Acquire → Learning

Memory
Adaptability

Tacit | Explicit | Implicit

Apply

Think, plan & reason

Problem → Solution

Explicit Knowledge Acquisition

Data Driven Learning
(Supervised/Unsupervised, self-supervised/semi-supervised learning)

Knowledge + Data Driven Learning

Evolutionary Computing

Reinforcement Learning

Training based | Supervised (labelled) | Unsupervised | Continuous | Adaptive | Lazy | Automated

Basic learning types

Acquire → Learning

Memory
Adaptability

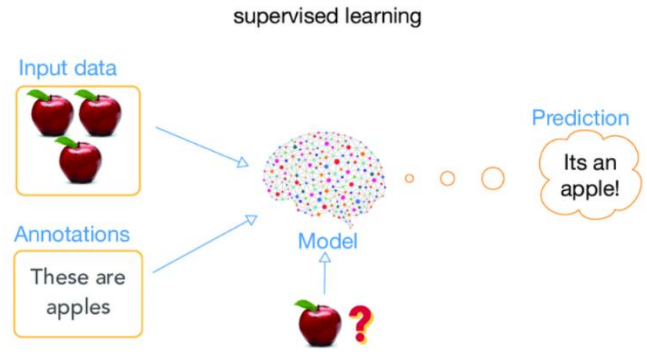
Tacit | Explicit | Implicit

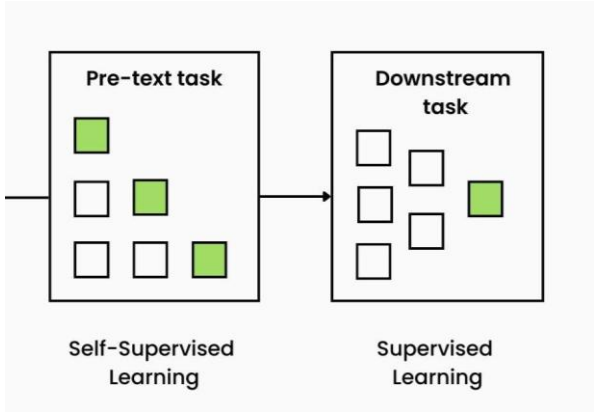
Apply

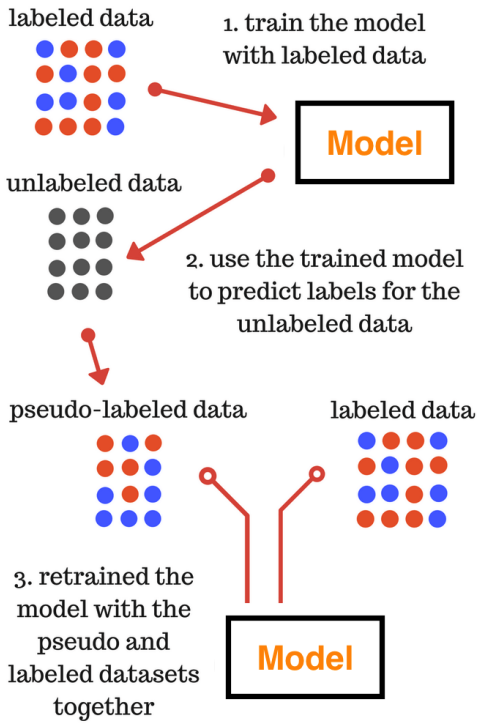
Think, plan & reason

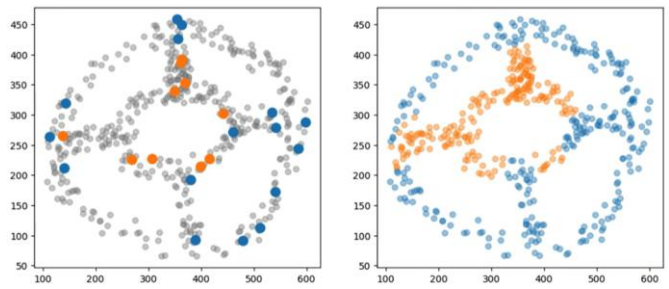
Problem → Solution

Data driven learnings









Semi-supervised learning

[Boosting vs. semi-supervised learning]

[Different types of machine learning models algorithms]

Basic learning types

Acquire → Learning

Memory
Adaptability

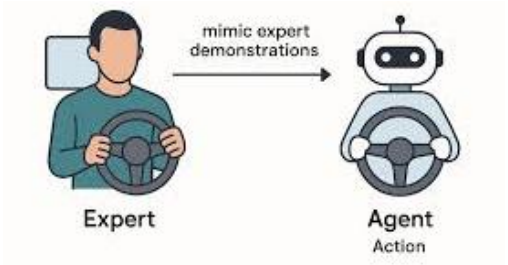
Tacit | Explicit | Implicit

Apply

Think, plan & reason

Problem → Solution

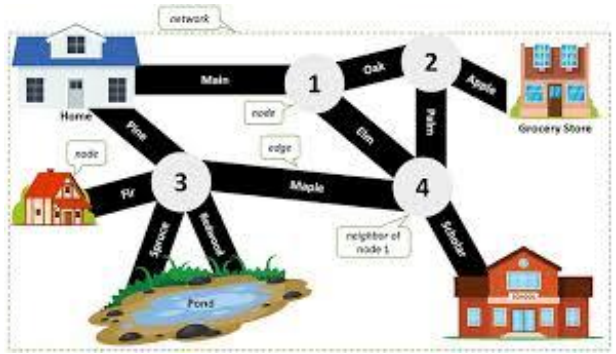
Learning from manuals and modelled behaviours



Learning from experts

Experiential Learning

Reinforcement learning



Learn with optimality

Data types and sources for learning

Acquire → Learning

Memory
Adaptability

Tacit | Explicit | Implicit

Apply

Think, plan & reason

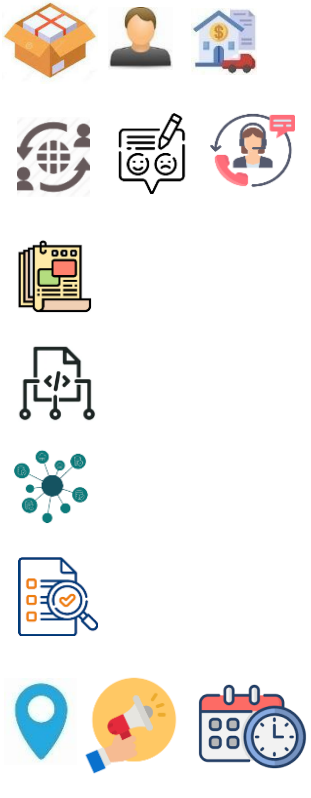
Problem → Solution

Sources

- Documents | Reports
- Internet
- Social Networks
- Sensors and devices
- Past solved cases
- Databases | Data (including multi-media Files
- People
- Automated Systems

Types

- Master Data
- Transactional | Usage
- Documents
- Metadata
- External data
- Augmented data
- Geo, location, time data, etc.



Data types and sources for learning

Acquire → Learning

Memory
Adaptability

Tacit | Explicit | Implicit

Apply

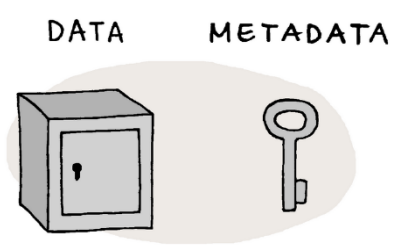
Think, plan & reason

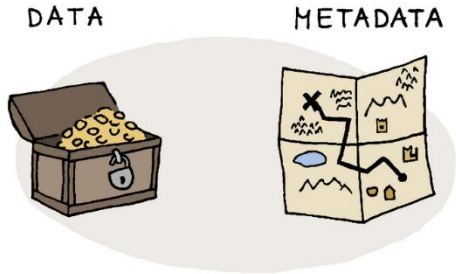
Problem → Solution

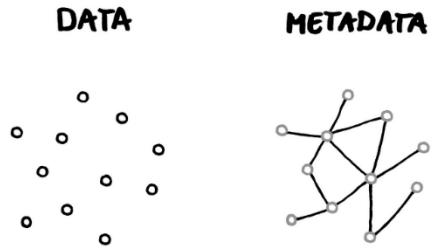
Right metadata, external and augmented data can help to automate most of the common sense and domain knowledge intelligence to connect the dots. It helps to separate apple from oranges or even green apple v/s red apple and build models to achieve higher accuracy.

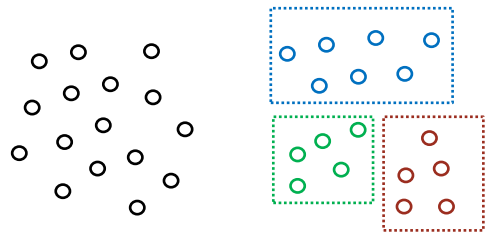
https://dataedo.com/blog/data-vs-metadata

Metadata is defined as "**data about data**". It is structured information that describes, explains, or provides context for other data to make it easier to find, manage, use and govern!



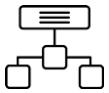




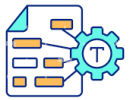


Recall example of predictive intelligence. Can it be modelled contextually to make predictions more accurate? Based on ad channel, demographics?

Descriptive: Used for discovery and identification; includes information like title, author, and keywords.



Structural: Describes how data elements are organized, such as how pages are ordered to form chapters in a book or the layout of an HTML document.



Semantic/contextual: Describes how data elements are related to each other meaningfully (semantics) for example how two qualifications are similar.

Administrative: Provides technical information to help manage a resource, such as when and how it was created, file type, and access permissions.

Role-based Access: who can access, what is allowed – read, update, field access, masking is needed? purpose of access - approval, preview, verify, ML job etc. , mode of access: through app, chatbot, API, when access happened: time-stamp of access and duration

Data types and sources for learning

Acquire → Learning

Memory
Adaptability

Tacit | Explicit | Implicit

Apply

Think, plan & reason

Problem → Solution

Structured | Semi-structured | Unstructured | Standard (e.g. RDF) and formats (Binary | CSV | Json/XML)

Raw form | Volume | Granularity | Features | Metadata | Continuous | Noisy | Uncertainty | Preprocessing | Imbalanced | Missing Values

Can be augmented using image processing, NLP, GenAI etc.

Data can be augmented from various sources

Can be retrieved from GPS/GIS

More structured, standard and granular data is always better. Data can be ingested easily, interoperability is taken care of, models and apps can be developed faster without any patch work or intermediate steps like conversion, transformation, etc.

Data source formats for learning

Acquire Learning

Memory
Adaptability

Tacit | Explicit | Implicit

Apply

Think, plan & reason

Problem

Solution

Image

Video

Text

Numeric

Audio

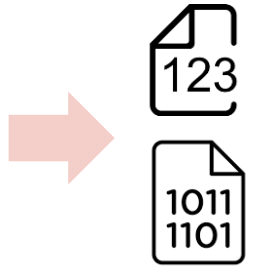
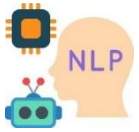


Image Processing



Natural Language Processing

Speech Processing

Data source formats for learning

Acquire → Learning

Memory
Adaptability

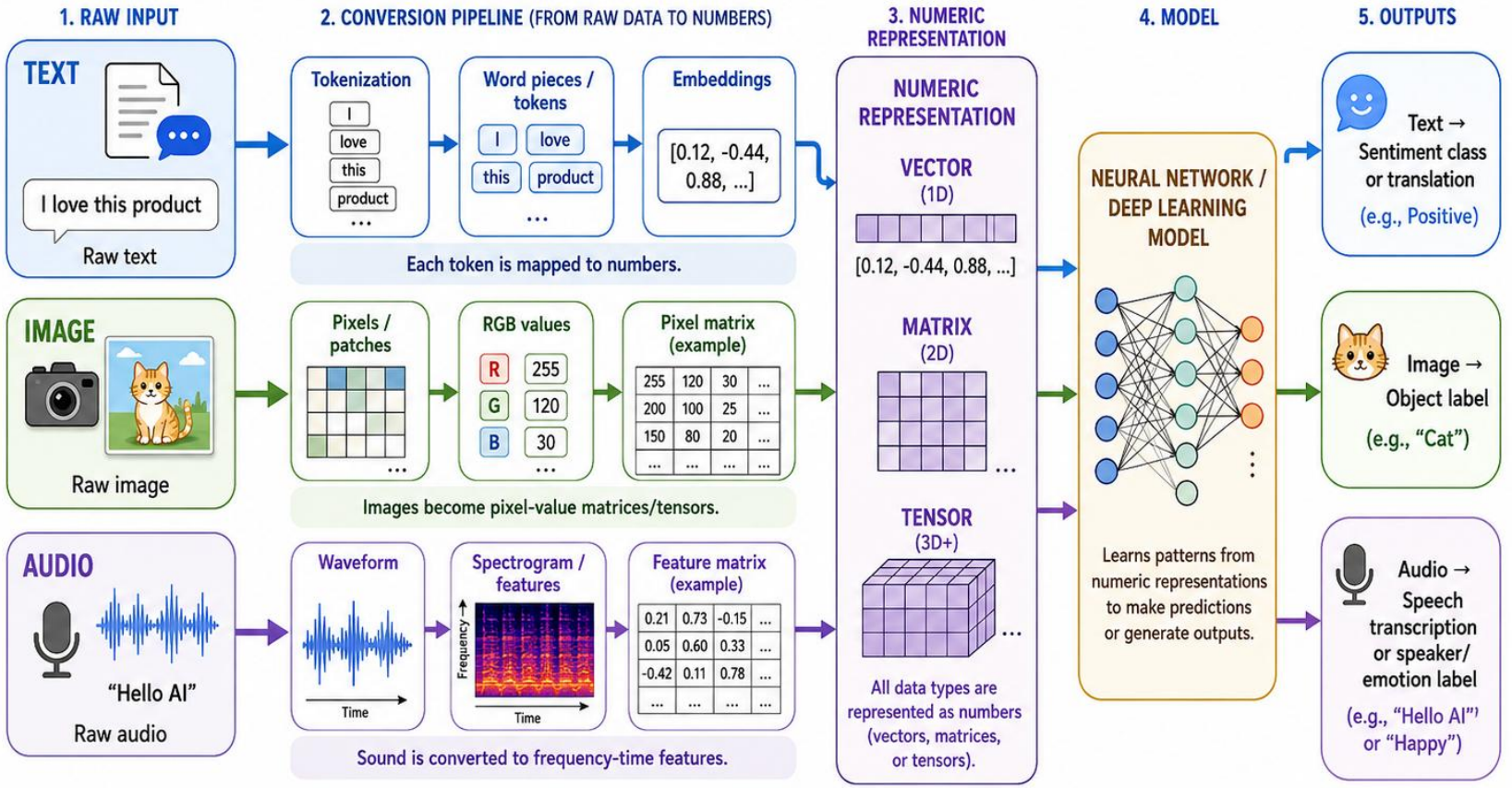
Tacit | Explicit | Implicit

Apply

Think, plan & reason

Problem → Solution

Neural networks do not directly understand raw text, pictures, or sound.
They process numeric representations such as vectors, matrices, and tensors.



Key takeaway: Different data types use different preprocessing, but all are transformed into **numbers** before being processed by neural networks.

Data sources e.g. Banking application

Acquire → Learning

Memory
Adaptability

Tacit | Explicit | Implicit

Apply

Think, plan & reason

Problem → Solution

More data, better analysis in 360 degree. Takes care of dimensions like what (e.g. transaction), who, where (channel, location), when (time), why (reason)…

Category	Description
Demographics (including KYC)	Demographic and KYC-related information of the customer.
Customer Data (Product-based)	Customer data captured at the time of availing/subscribing to a product (e.g. loan application data).
Current Data	Current status of customer products such as savings balance and outstanding loan amounts.
Preferences	Customer preferences such as transaction limits, transaction type, transaction time, etc., captured via conversations, interactions or explicit input.
Product Meta-data	Meta-data of products/services such as loan types, features, terms and conditions, etc.
Purchases	Products or services subscribed to, availed or purchased by the customer.
Transactions	Transaction data including payments, repayment history, participants, type of transaction, location and time.
Interactions	Customer interaction data such as feedback, customer care interactions and resolution status.
Document Meta-data	Meta-data of documents such as TOCs, taxonomies, keywords, tags, etc., derived using NLP.
Locations	Location-related data including location meta-data.
Time Data	Time-based and event-based data such as holidays, festivals and special events.
Task-based Data	Data related to specific tasks such as fraud cases and NPA cases.
External Data	Data from external ecosystem players such as real-estate agents, builders, projects, social data and CIBIL scores.

Making data models and data AI ready

Acquire → Learning

Memory
Adaptability

Tacit | Explicit | Implicit

Apply

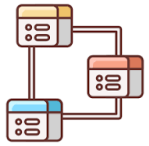
Think, plan & reason

Problem → Solution

Attributes are also referred as 1> features (e.g. in ML), 2> parameters (e.g. modelling), 3> variables (in coding), 4> fields (in database/SQL), 5> columns (spreadsheets, DB/SQL) etc. based on context.

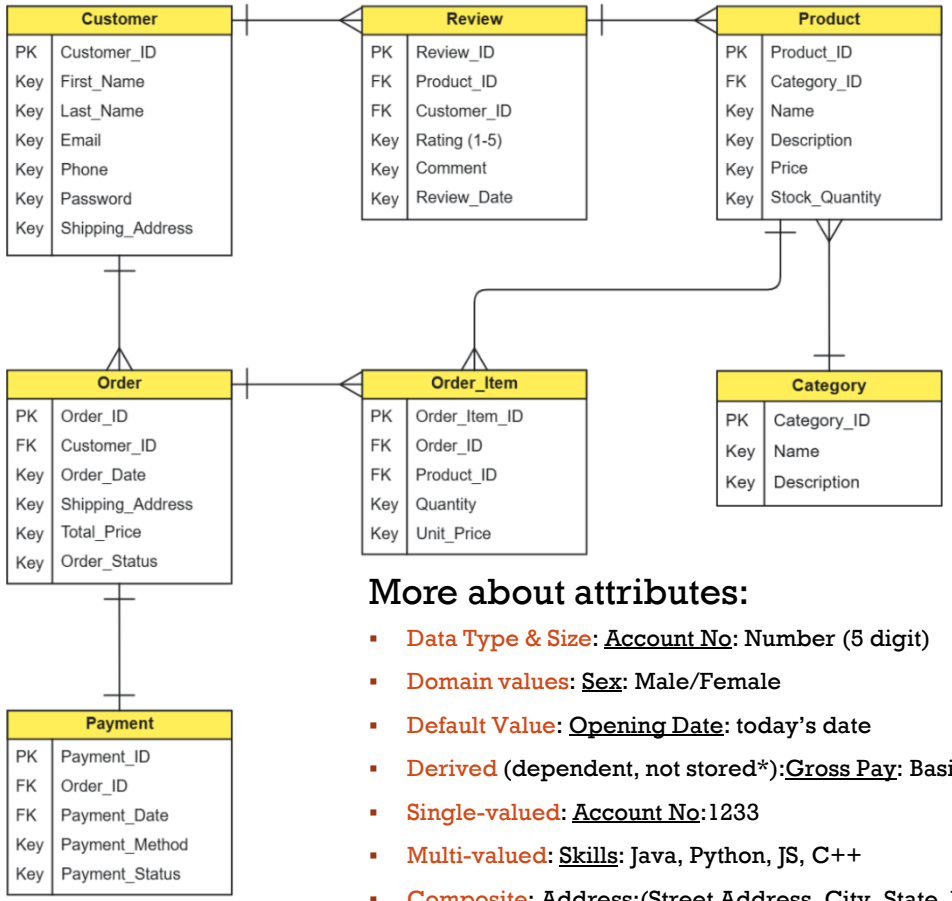
Data model helps to understand structural relationships, constraints between various entities and more about each and every entity in detail.

https://www.template.net/edit-online/498431/e-commerce-system-erd



Data model is a blueprint or abstract representation that organizes data, defines its relationships, and establishes rules (constraints) for how it's stored and used, acting as a foundation for databases and systems to ensure consistency, clarity, and efficient management of information for analysis and applications

E-commerce System ERD Template



Well thought and designed data models help in managing (updates, retrieval, etc.) data based on business logic using SQL (structured query language). It helps Gen AI to convert natural language text into SQL queries helping business users to see the reports, dashboards based on their requirements without knowing SQL.

More about attributes:

- Data Type & Size: Account No: Number (5 digit)
- Domain values: Sex: Male/Female
- Default Value: Opening Date: today's date
- Derived (dependent, not stored*):Gross Pay: Basic + DA + HRA + CCA
- Single-valued: Account No:1233
- Multi-valued: Skills: Java, Python, JS, C++
- Composite: Address:(Street Address, City, State, Pin)
- Key/connected attributes: Account no

Making data models and data AI ready

Data model for AI Ready Data

Acquire → Learning

Memory
Adaptability

Tacit | Explicit | Implicit

Apply

Think, plan & reason

Problem

Solution

LLM

Apps, agents, analytics

Run Time

Configurable Components | Models

Customized UI/UX & vibe coding

Solution data & meta-model

Data | Vocabulary

Detailed Business SRS

Data and LLM driven
Apps, Analytics, AI and Agents

Conventional data models were meant for transactional and operational systems, enterprise systems like ERPs, CBS etc. Taking care of storing every details required from transaction details, accounting, audit, compliance, reconciliation etc.

AI ready data needs to be clean, granular (normalized) to the extent possible, refined, meaningful and purposeful, highly normalized, connected, supported by meta-data, augmented and value-added data etc. Hence the data model required for AI ready data can be different than traditional data models.

These data models should be designed in such a way that they leverage Gen AI capabilities to build and manage everything on top of them like ML models, code, analytics, agents etc. Means models should be more semantic driven, standard and well understandable by Gen AI.

Good AI ready model serves two purposes: takes care of business requirements, structural relationships, constraints, business rules etc. at the same time can make more semantic, meaningful and purposeful.

Newcomers can create data model which serves both the purposes operational as well as AI-readiness. Really can leverage analytics, AI, ML and agentic AI. With GenAI, it is time to the organizations can reimagine own all inclusive software solutions.

Making data models and data AI ready

Acquire → Learning

Memory
Adaptability

Tacit | Explicit | Implicit

Apply

Think, plan & reason

Problem → Solution

AI scales only when data is strategic, unified, trusted and cost-efficient.

[5 steps to build strong data foundations for gen AI | Google Cloud Blog]

01 Build an AI-first data strategy

Align people, processes, policies and technology to strategic priorities.

02 Establish a single source of truth

Unify data access across formats and silos so AI can use reliable inputs.

03 Accelerate data-to-AI with AI

Use AI agents and conversational analytics to improve discovery and analysis.

04 Govern and secure the AI lifecycle

Manage quality, lineage, access, observability and compliance end-to-end.

05 Improve efficiency to reduce costs and scale

Optimize storage, compute, ingestion and operations with managed services and FinOps.

Strategic asset → unified access → AI-assisted use → trusted lifecycle → scalable economics

Value-adding knowledge (e.g. meanings/semantics): Taxonomy

Acquire → Learning

Memory
Adaptability

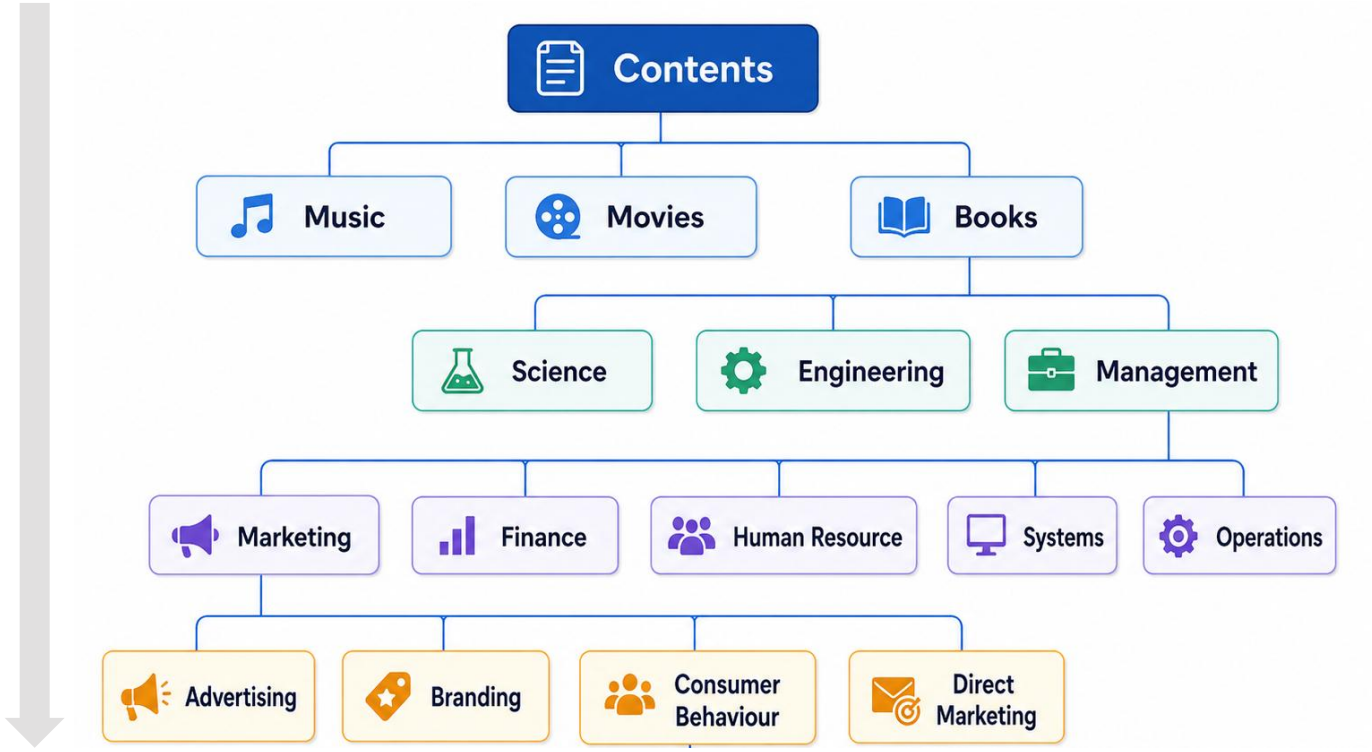
Tacit | Explicit | Implicit

Apply

Think, plan & reason

Problem

Solution



Taxonomies can help 1> to reduce search space, 2> make search contextual and bottom-up search, 3> recommend products across categories, 4> create/build specific models with better accuracy (e.g. forecasting only specific kind of products e.g. Business v/s Consumer etc.), 5> shows how contents are similar ….

Semantic nets

Semantic Net / Semantic Network: A semantic net is a way of representing knowledge using a graph structure.
It represents:
Objects / concepts as nodes
Relationships between them as links / edges
So, instead of storing knowledge only as text, we store it as connected concepts.

Acquire → Learning

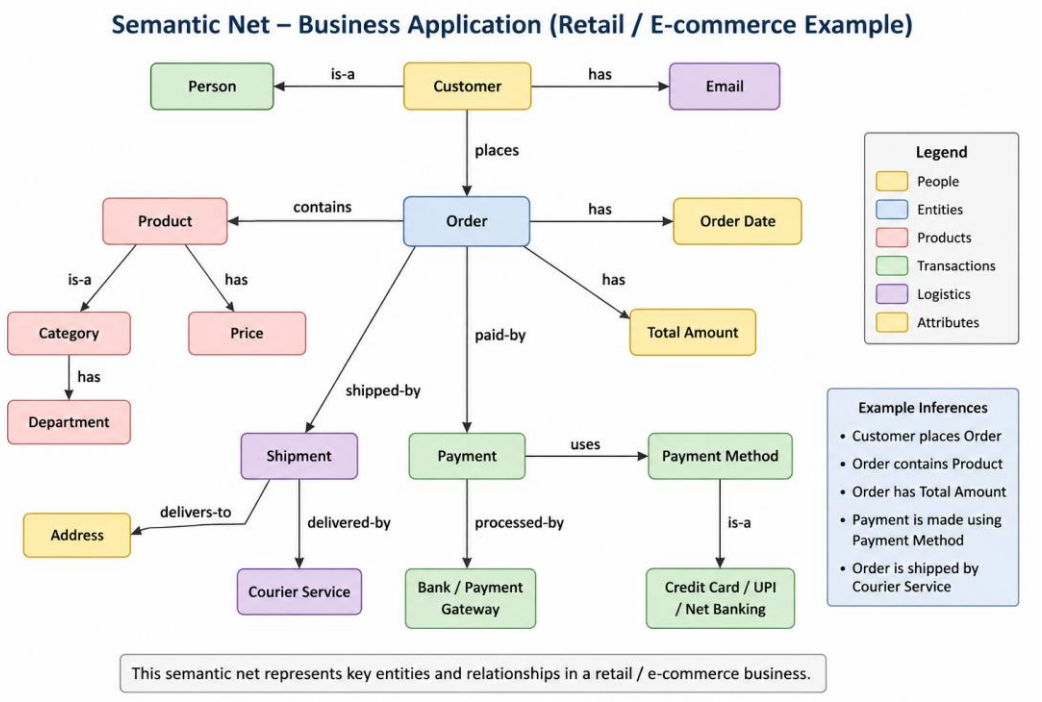
Memory
Adaptability

Tacit | Explicit | Implicit

Apply

Think, plan & reason

Problem → Solution



Relationship	Meaning	Example
is-a	Class/category relation	Dog is-a Animal
instance-of	Object belongs to a class	Tommy instance-of Dog
part-of	Component relation	Wheel part-of Car
has-property	Attribute relation	Apple has-property Red
can-do	Capability/ action	Bird can-do Fly
causes	Cause-effect relation	Rain causes Wet Road

Given relationships:	We can infer:	Suppose:	Then the system can infer:
Customer places Order Order contains Product Product is-a Category Category has Department	Customer buys Product Customer is interested in Category Customer may belong to Department-level preference segment	Raj places Order O101 Order O101 contains Running Shoes Running Shoes is-a Footwear Footwear has Department Fashion	Raj bought Running Shoes Raj is interested in Footwear Raj may be targeted for Fashion-related offers

What Is a Knowledge Graph? Definition & How It Works | Timbr.ai

Knowledge Graphs

Acquire → Learning

Memory
Adaptability

Tacit | Explicit | Implicit

Apply

Think, plan & reason

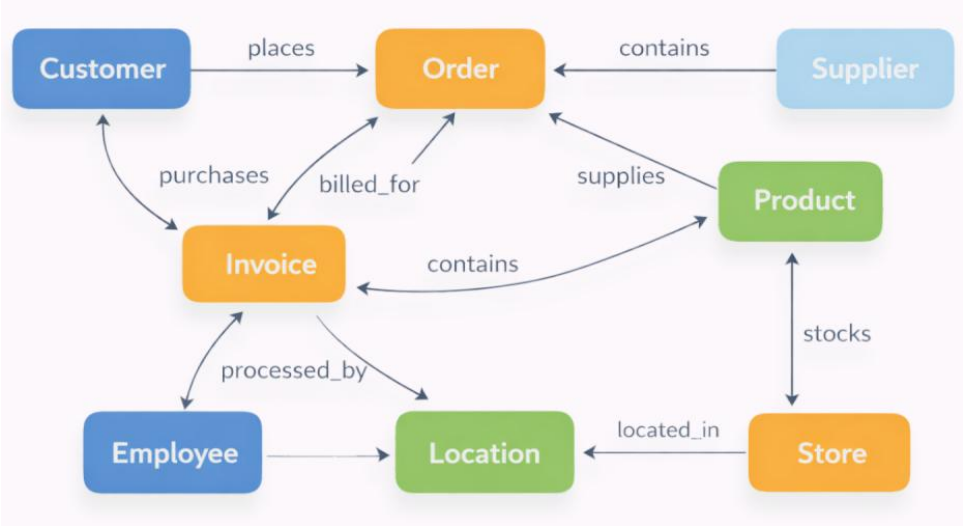
Problem → Solution

A knowledge graph (also referred as semantic network) is a structured, semantic network that connects real-world entities—people, places, concepts, or objects—using nodes and edges to represent relationships. It acts as a database, transforming raw, siloed data into interconnected, machine-understandable information to provide context for AI, search engines, and analytics

Nodes (Entities): These represent the "things" in the graph, such as a specific customer, a product, or a location.

Edges (Relationships): These define how the nodes are connected (e.g., "Customer X *purchased* Product Y").

Labels/Properties: These provide additional descriptive details about the entities and their relationships, such as a person's birthdate or the timestamp of a transaction.



Semantic Net	Knowledge Graph
Older AI knowledge representation technique	Modern large-scale semantic representation
Mostly conceptual	Often used in real systems
Nodes and links	Entities and relationships
Used in expert systems and AI reasoning	Used in search engines, enterprise AI, recommendation, GenAI grounding

[What Is a Knowledge Graph? Definition & How It Works | Timbr.ai]

Knowledge Graphs

Acquire → Learning

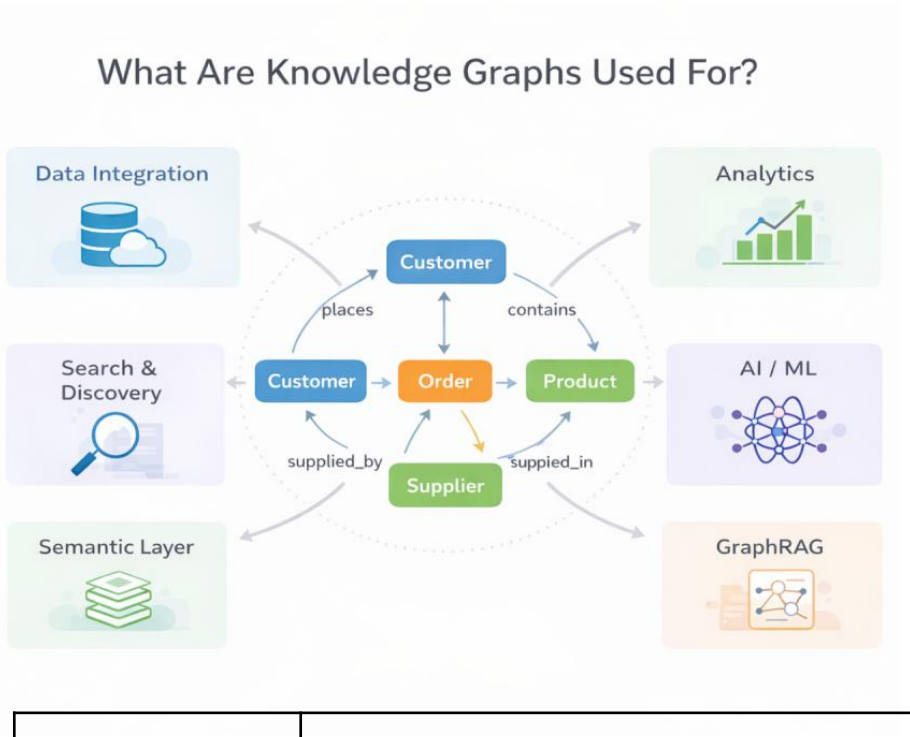
Memory
Adaptability

Tacit | Explicit | Implicit

Apply

Think, plan & reason

Problem → Solution



Data Integration and Unification	Organizations use knowledge graphs to link data across different systems without forcing it into a single physical schema. Entities and relationships can be defined over existing sources while each system remains intact.	A customer in a CRM is connected to their orders in an e-commerce system and their support tickets in a service platform without copying or reshaping the underlying data.
Analytics and Decision-Making	By modeling relationships directly, knowledge graphs make it easier to ask complex questions that span multiple dimensions, without repeatedly embedding business logic in joins or transformations.	An analyst examines how product returns correlate with suppliers, regions, and delivery timelines through a single query path.
Search and Discovery	Knowledge graphs enhance search by understanding intent and relationships rather than just keywords, enabling more accurate and context-aware results. This works whether the graph is materialized or resolved dynamically at query time.	A search for a customer surfaces related contracts, active orders, and recent issues instead of unrelated records with similar names.
AI and Machine Learning Support	Knowledge graphs provide structured and governed context that AI systems can rely on. By grounding models in explicit relationships and definitions, they reduce ambiguity and improve reliability of AI-driven analysis.	An AI assistant answering revenue questions uses defined relationships between customers, orders, and invoices instead of inferring meaning from raw tables.

What Is a Knowledge Graph? Definition & How It Works | Timbr.ai

Knowledge Graphs

Acquire → Learning

Memory
Adaptability

Tacit | Explicit | Implicit

Apply

Think, plan & reason

Problem → Solution

DB schema v/s knowledge graphs

Aspect	Normalized DB Schema	Knowledge Graph
Data Model	Relational (tables, rows, foreign keys)	Graph (nodes, edges, labels)
Purpose	Efficient storage, reduce redundancy	Semantic relationships and inference
Relationships	Implicit via foreign keys	Explicit via labeled edges
Query Language	SQL	SPARQL / Cypher
Flexibility	Rigid schema	More flexible and extensible
Semantics	Usually lacks semantics	Rich with ontology and metadata

SPARQL (protocol RDF query language) is the query language used to query RDF-based knowledge graphs and semantic web data. It is a W3C standard for querying RDF graphs.

subject -> relationship/predicate->object
e.g. customer C101 places order ORD1011
ORD1011 contains products PRD001…

Which products did Customer C101 buy?

Cypher is a graph query language mainly used with Neo4j graph database.

(Customer)-[:PLACES]->(Order)

[What Is a Knowledge Graph? Definition & How It Works | Timbr.ai]

Knowledge Graphs

Acquire → Learning

Memory
Adaptability

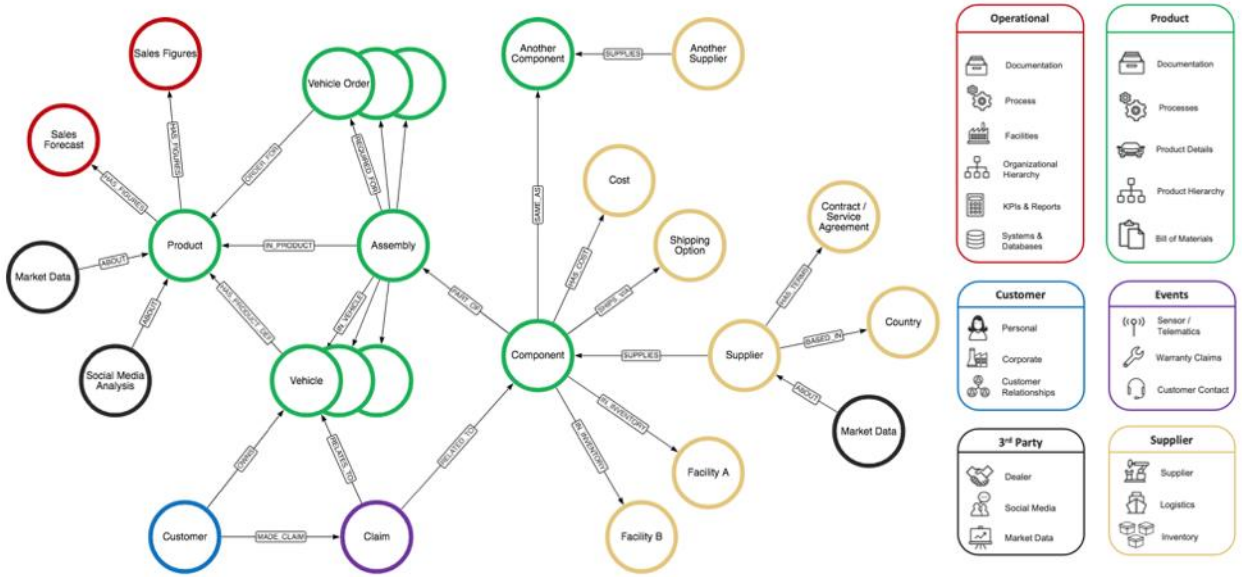
Tacit | Explicit | Implicit

Apply

Think, plan & reason

Problem → Solution

Example: The World's Leading Companies Use Neo4j to Manage Supply Chains, Boost Resilience and Ensure Business Continuity



Caption: An example of the variety of data types within an automotive industry supply chain. When analyzing these entities in a Neo4j graph database, the interconnections – and thus the vulnerabilities – of the supply chain become clear.

Leading Companies Use Neo4j to Manage Supply Chains

Ontologies

An ontology defines the structure, meaning, rules, and vocabulary of a domain.

Acquire Learning

Memory
Adaptability

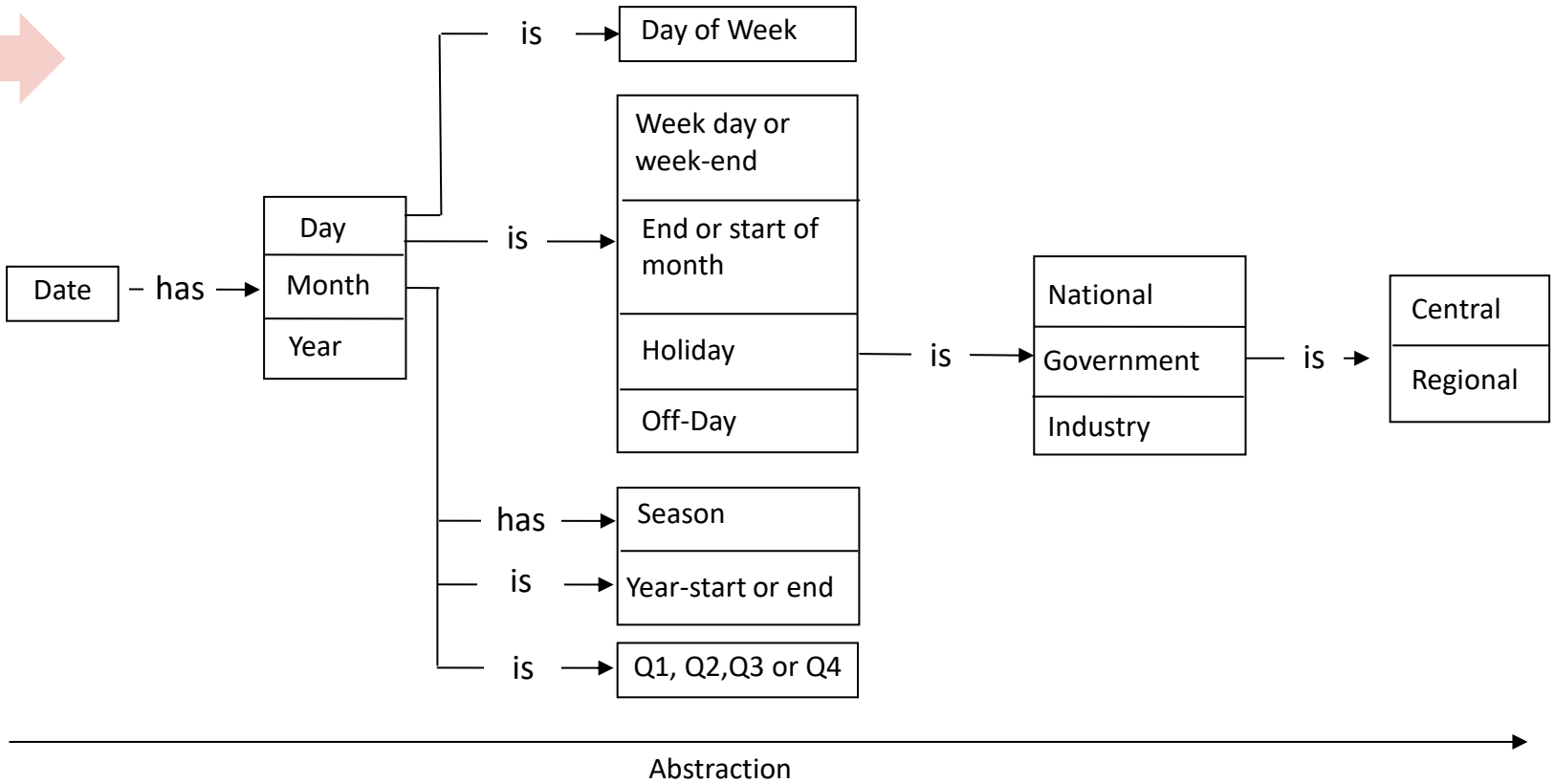
Tacit | Explicit | Implicit

Apply

Think, plan & reason

Problem

Solution



Ontology = Schema + meaning + rules
Knowledge Graph = Actual data represented as a graph

Taxonomies, Knowledge-graphs and Ontologies

Acquire → Learning

Memory
Adaptability

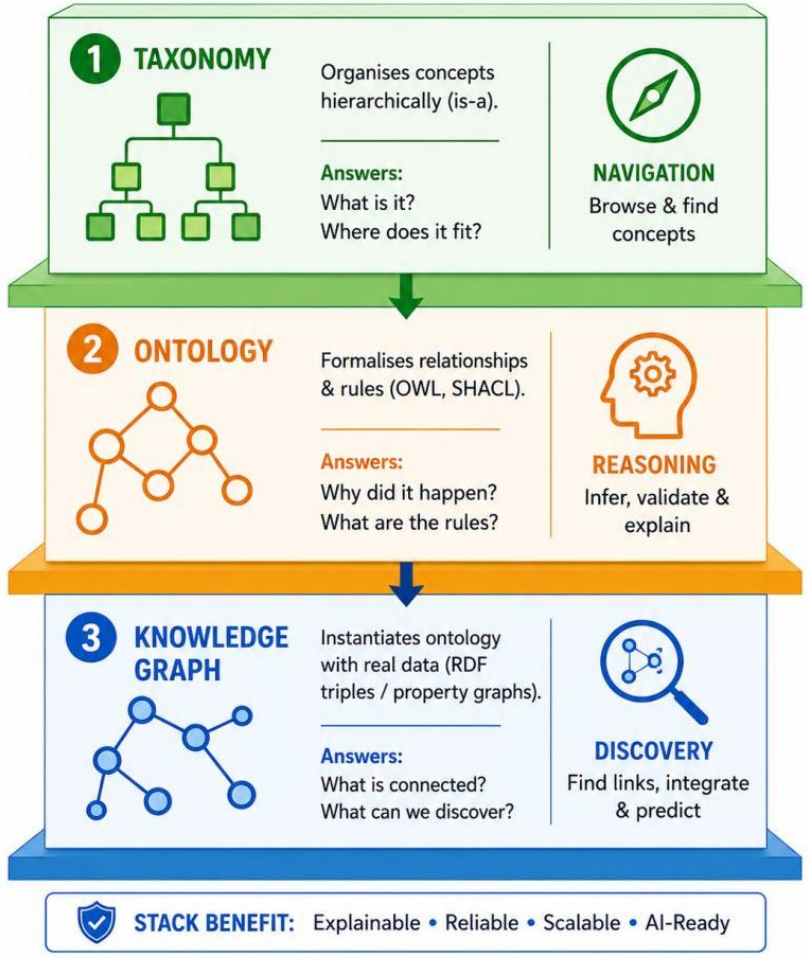
Tacit | Explicit | Implicit

Apply

Think, plan & reason

Problem

Solution



(25) Knowledge Management, in Cyber Physical Human system, via Knowledge graphs, Ontologies, Taxonomies | Groups | LinkedIn

Modelling general data entities

Entities and profiling (brief about entities)

Acquire → Learning

Memory
Adaptability

Tacit | Explicit | Implicit

Apply

Think, plan & reason

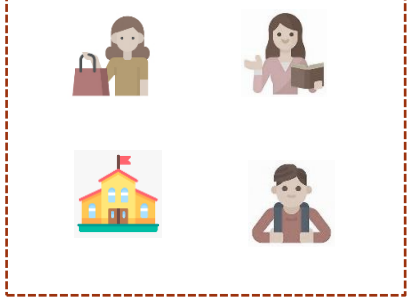
Problem → Solution

Modelling core entities (and related ones) like stakeholder, product, resource, transaction, location and calendar, time etc. keeps the data AI ready and easier to model generic product, customer and transaction agnostic ML models and AI.

Items

Products, contents, resources/ assets like machines, service, process, designs, drawings …

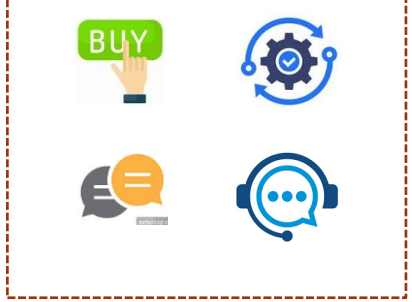
Users | Parties



Players and stakeholders

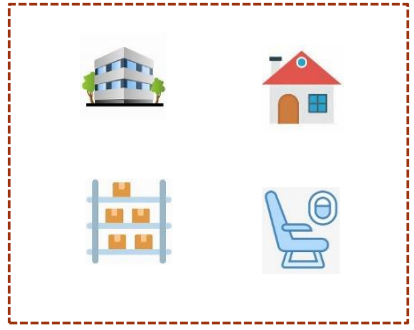
Customers, consumers, vendors, partners…

Transactions (Interactions ,Usage , etc.)



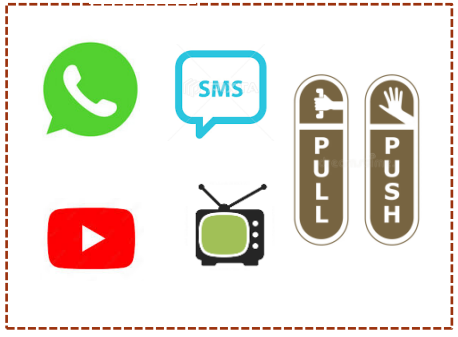
Posting and recording of tasks and transactions

Locations



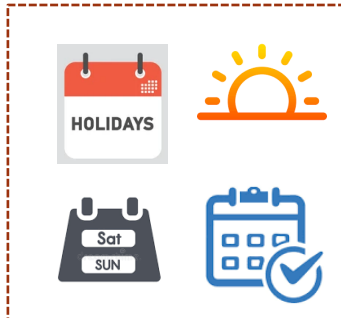
Locations of items & players (users)

Channels | Touch points

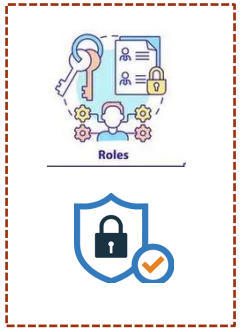


Promotion & communication channels

Time



Time of transactions, schedules



Roles and access, security and privacy

Modelling general data entities

Acquire → Learning

Memory
Adaptability

Tacit | Explicit | Implicit

Apply

Think, plan & reason

Problem → Solution

Entities and profiling (brief about entities)

Entity	Sub-type	Examples
Party (who)	customer	customer, client, user, visitor, student, employee
	Provider	supplier, employee, agent, firm, expert, business partner
Item (what)	Product	book, article, chapter, electronic gadgets (laptop, digital camera, TV) , banking products , insurance products (life, auto, general), telecom products/plans, software, book, movie, music, album, jewellery, furniture, automobile, flower, grocery, property, vehicles (car, two-wheeler), property etc.
	Content	article, news, review, document, blog, video clip, job profile
	Resource/ Asset	office space, machine, equipment, car
	Service	courier service, web service, subscription service, car on rent
	Promotion	campaign, discount offers, advertisement
	Event	show, TV programme, conference, workshop
Transaction (activity/task/usage) (what, when, how, how much)		buy, sell, call, message, use, read, listen, watch, click, download, visit, review, rate item, send feedback, book, rent, reserve, blog, feedback, respond, reply, debit account, credit account, withdraw cash, pay, borrow, lend, transfer, create, share, usage
Location, Channel & Time	Physical	area, city, state, country
	Online	website, WAP, URL,
	Contact	phone no, email, fax
	Mobile	train, flight, bus, ship

Based on requirement each entity can be profiled based on meta-data, transactions, location etc.

To leverage AI better make sure you have AI ready data model (and database) of all your stakeholders like employees, customers, suppliers, partners, products and services, events, campaigns, resources (machines, equipment, software, vehicles, office rooms, etc.), transactions, usage and interactions (based product type/services), feedbacks, logs, processes, locations, calendars, etc. supplemented by meta-data and value-adding data etc. Generic models make AI/ML solutions agnostics, extensible and increase reusability.

Modelling general data entities

Product Types and ecosystem: Entertainment contents like movies

Acquire → Learning

Memory
Adaptability

Tacit | Explicit | Implicit

Apply

Think, plan & reason

Problem → Solution

Production house, producer, director, writer (script), actors, lyricists, singers etc.

Production House

Content creation

Individuals creating video contents

User-generated

Content

Meta-data, format, resolution, plot, songs, scenes, Features: skip, jump, embed ads, etc.

Promotion

Streaming

Streaming servers, bandwidth, resolution etc. ads, etc.

Platform

Content availability, search, wish list, likes, reviews, feedback, ads, P&R, support on many platforms etc.

Consumption

Usage (what, when, how), device, time etc.

Distributors

Cinema Theatres

What is advantage of owning whole ecosystem v/s 1> only content, 2> only streaming platform. Streaming platforms have more insights from the usage of contents by the customers! Sit on huge customer experience data.

Modelling general data entities

Product Types and ecosystem: Machines, equipment

Acquire Learning

Memory
Adaptability

Tacit | Explicit | Implicit

Apply

Think, plan & reason

Problem

Solution

Machine, equipment Manufacturer

Machine, equipment

Manufacturer machines

Manufacturer, firms using machines

Use machines for manufacturing.

IoT Platform

Makes machines live to track usage of machines, manage and monitor machines well

Digital twin/ Analytics/AI Platform

Easy onboarding, Provide insights on usage of machines, predictive maintenance, etc.

Adds value, optimizes usage and downtime, reduces cost

Modelling general data entities

Acquire → Learning

Memory
Adaptability

Tacit | Explicit | Implicit

Apply

Think, plan & reason

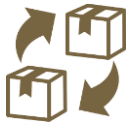
Problem → Solution

Similar way, other entities like party (related parties, friends, colleagues, employee etc.), transaction (linked transactions, sub-transactions, time etc.), usage, location can be modelled

More about item entities

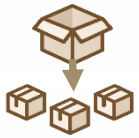
A item can be combination of integrated components (items). Each component can be reused, put into other product etc. A food dish is combination of various ingredients. Understanding ingredients helps to understand likes of customer in general and helps in better personalization.

A item can be a bundled item consisting of multiple items. This can be applicable when products are offered in bundle may be as a part of campaign. Products can be bundled together using market-basket analysis.

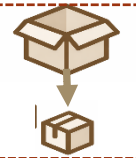


An item can be optional related item (accessory) for other item. E.g. wireless keyboard and mouse for laptop. This can help to recommend accessories when customers are buying base product.

An item can be mandatory related item (accessory) for other item. E.g. charging plug is must for mobile devices. This can help in packaging, managing inventory well.



An item can be broken into multiple items. E.g. a movie can be smaller contents like songs, action scenes etc. There is opportunity to create different contents for customers e.g. song album of movie, action scenes etc.



An item can be converted/customized into small item e.g. summarizing text or summarizing meeting discussion.

Digital item can be broken into small parts, item can be searched, usage can be tracked at detailed level, customized, personalized, etc.

Digital Contents

Acquire → Learning

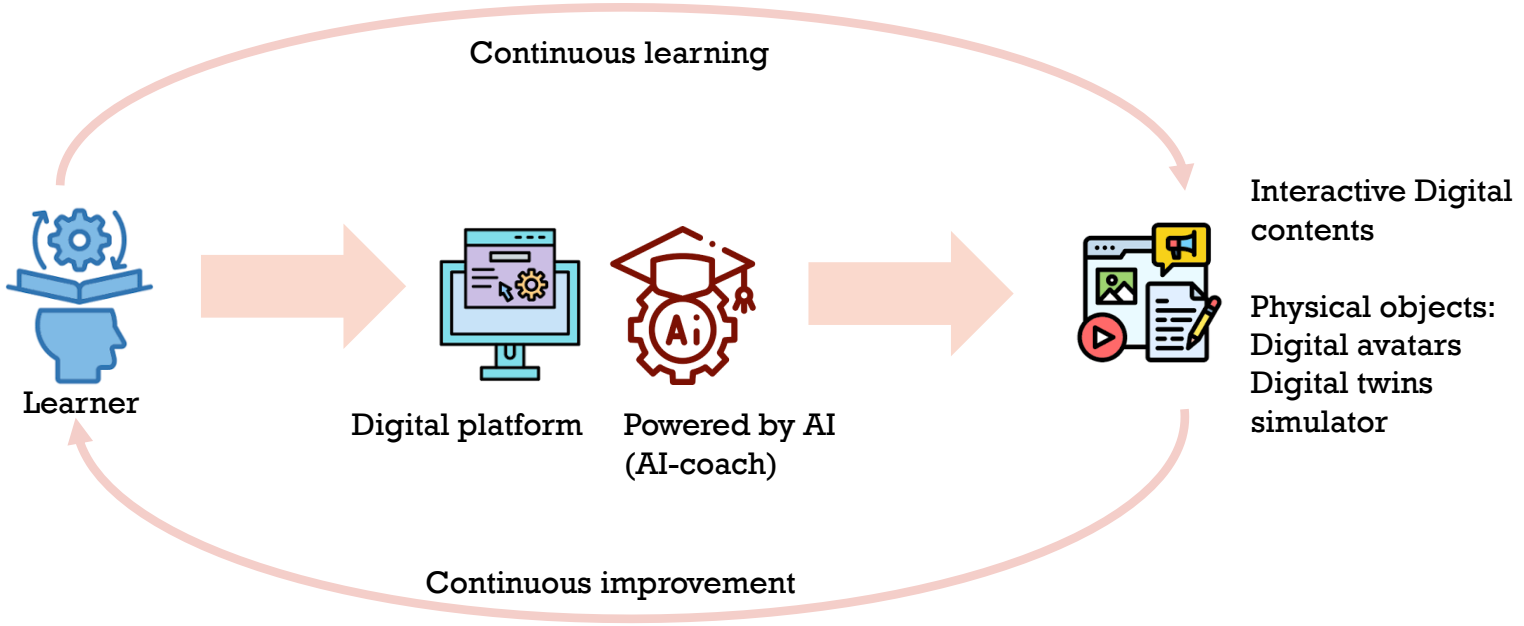
Memory
Adaptability

Tacit | Explicit | Implicit

Apply

Think, plan & reason

Problem → Solution



- ✓ From anywhere at anytime (through multi modal options)
- ✓ Can benchmark learning progress with set standards
- ✓ Help to improve based on deviations, suggest how improvements can be achieved
- ✓ Can personalize learning based on learner's pace of learning

- ✓ Can be broken into smaller parts making more modular and granular
- ✓ Different aspects can be set (e.g. voice quality, tone, emotional expression, consistency for singing)
- ✓ Difficulty levels can be set (novice, intermediate, expert, etc.)
- ✓ Needs experts while creating the contents.

Modelling general data entities

Product modelling (it can be at category level) using Transactions/usage

Acquire Learning

Memory
Adaptability

Tacit | Explicit | Implicit

Apply

Think, plan & reason

Problem

Solution

How product is used by users? e.g. content subscribers frequently skipped/jumped contents? Which one?

Is product in repeated (repeat buy, watched again), at what frequency?

How long it takes customer to decide? What events (e.g. ratings and reviews, specifications, etc.) customers goes through to take the decision?

When customers prefer to buy/watch product?

What is overall sentiment about the product?

What other products bought along with the product, before and after?

What way customers come to know about this product?

Modelling general data entities

Acquire → Learning

Memory
Adaptability

Tacit | Explicit | Implicit

Apply

Think, plan & reason

Problem → Solution

Interactions with generic entities

Who?

With whom?

For/on what, how, in response to what?

When?

By what channel and responded to what?

Where (amazon website?

Money?

e.g. Netflix: Who watched (id, profile)? what content (content id)? when (time)? from where (location), how (duration, skips, intermittent, etc.) ? on what device (type, resolution etc.)? how content was found (search, recommendation?), by what channel in-app, email, sms, etc.

Rajendra

Amazon

Laptop

1 Jan 2025 14:00

Amazon (10% discount campaign)

Amazon web site From home

INR 50K?

Rajendra bought laptop X from **Amazon app** at **Rs.50K** in response to **10% discount campaign** from home (**IP address**) at **1st Jan 2025 14:00**

Customer transaction product from **provider** at **location** at **price** in response to **campaign** from **location** at **date-time**

What **Rajendra buys** from **Amazon app**? **When?** — Personalized insights

What kind of **laptops** customers like **Rajendra buy** from **Amazon?** — Wisdom of crowd

Modelling generic entities with meta-data, augmented data, external data helps to understand them better, great degree of insights, patterns can be derived by connecting them to transactions/interactions amongst the entities to help wow customer service, bring efficiency, offer innovative products etc.

Modelling general data entities

Generic entities: Connecting dots

Acquire → Learning

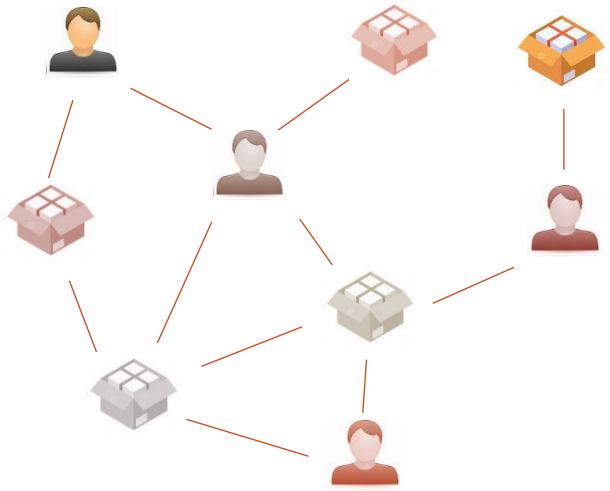
Memory
Adaptability

Tacit | Explicit | Implicit

Apply

Think, plan & reason

Problem → Solution



- Structural meta-data, behavioural profile
- Using common-sense, domain knowledge
- Using ML (e.g. association rule mining, content-based filtering)
- Using transactions and usage
- Location and time

- Understanding connection
- What makes them to connect?
- Is connection suspicious (e.g. related party trans?)
- …

N=1 Analysis: Generic data model and basic entities:

Dimensions: party (*who*), item (*what*), location and time (*where* and *when*)

Acquire → Learning

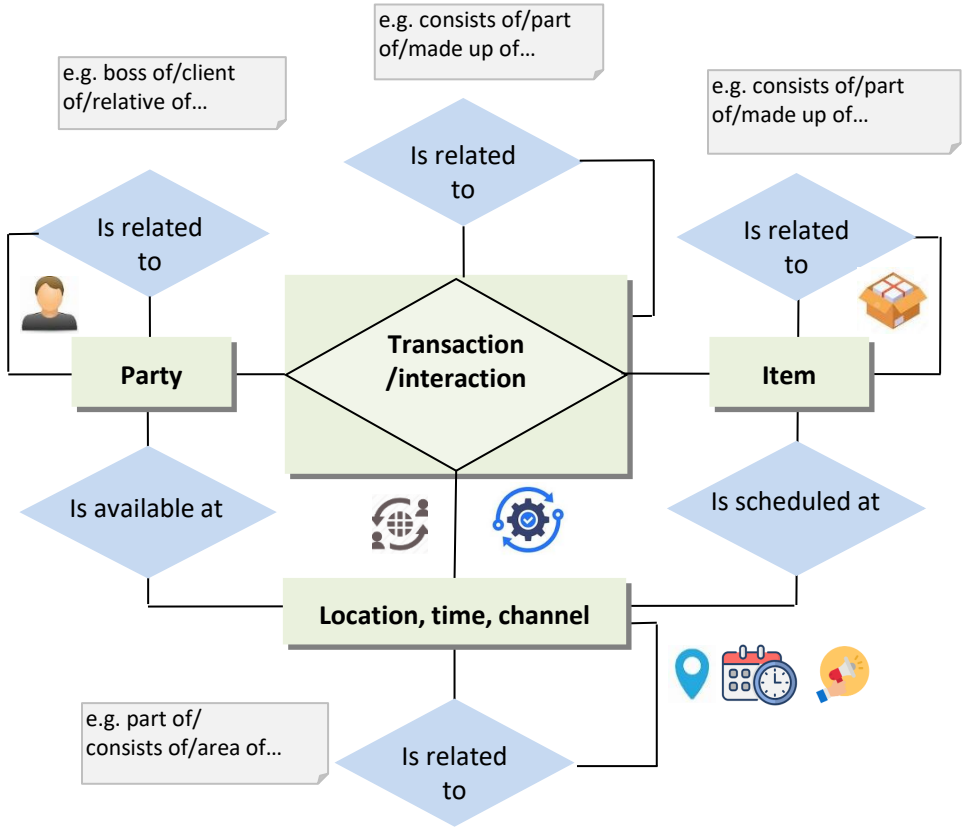
Memory
Adaptability

Tacit | Explicit | Implicit

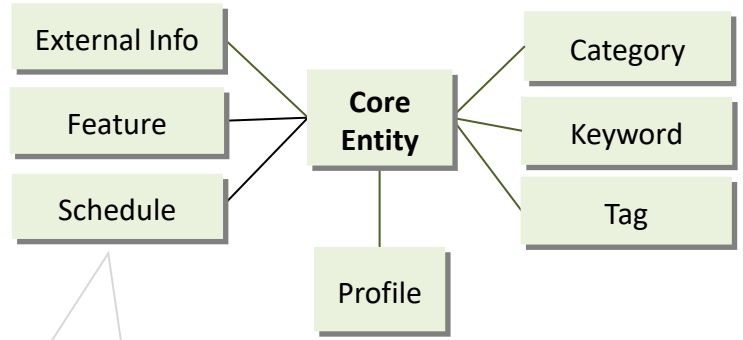
Apply

Think, plan & reason

Problem → Solution



Meta-data/knowledge



Connected entities can help to derive insights to augment data e.g. customer profile can be created based on brought bought e.g. if a product is targeted or positioned for certain gender, age group and a customer has bought it, age group and gender can be augmented in customer database.

Entities, usage, algos, value and strategies

Acquire → Learning

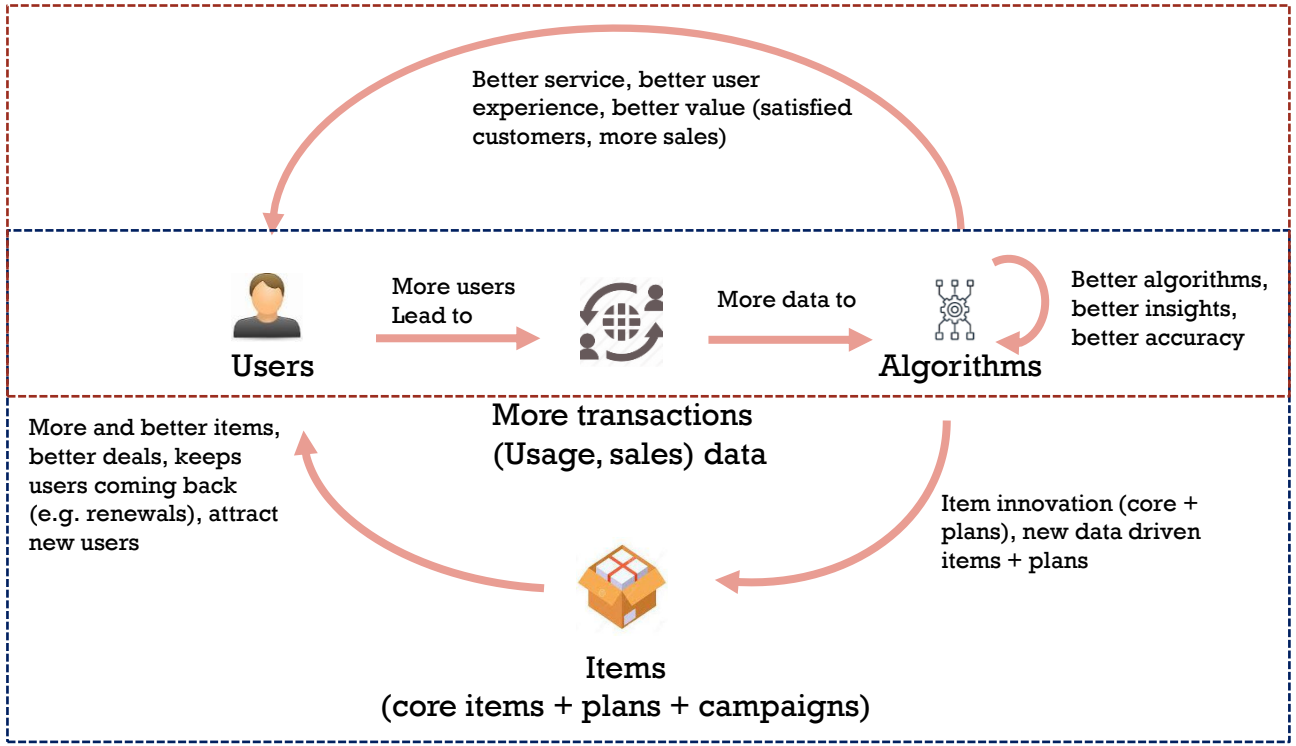
Memory
Adaptability

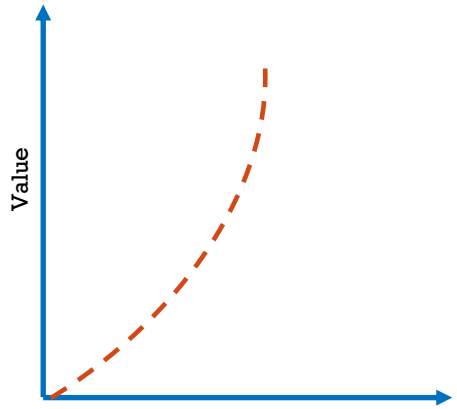
Tacit | Explicit | Implicit

Apply

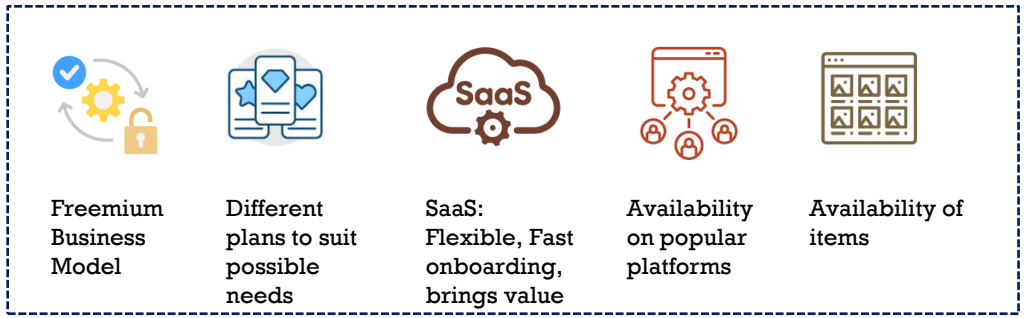
Think, plan & reason

Problem → Solution





Strategies to increase users apart from platform features



Increase → Clients/Users

Entities, usage, algos and value

Acquire → Learning

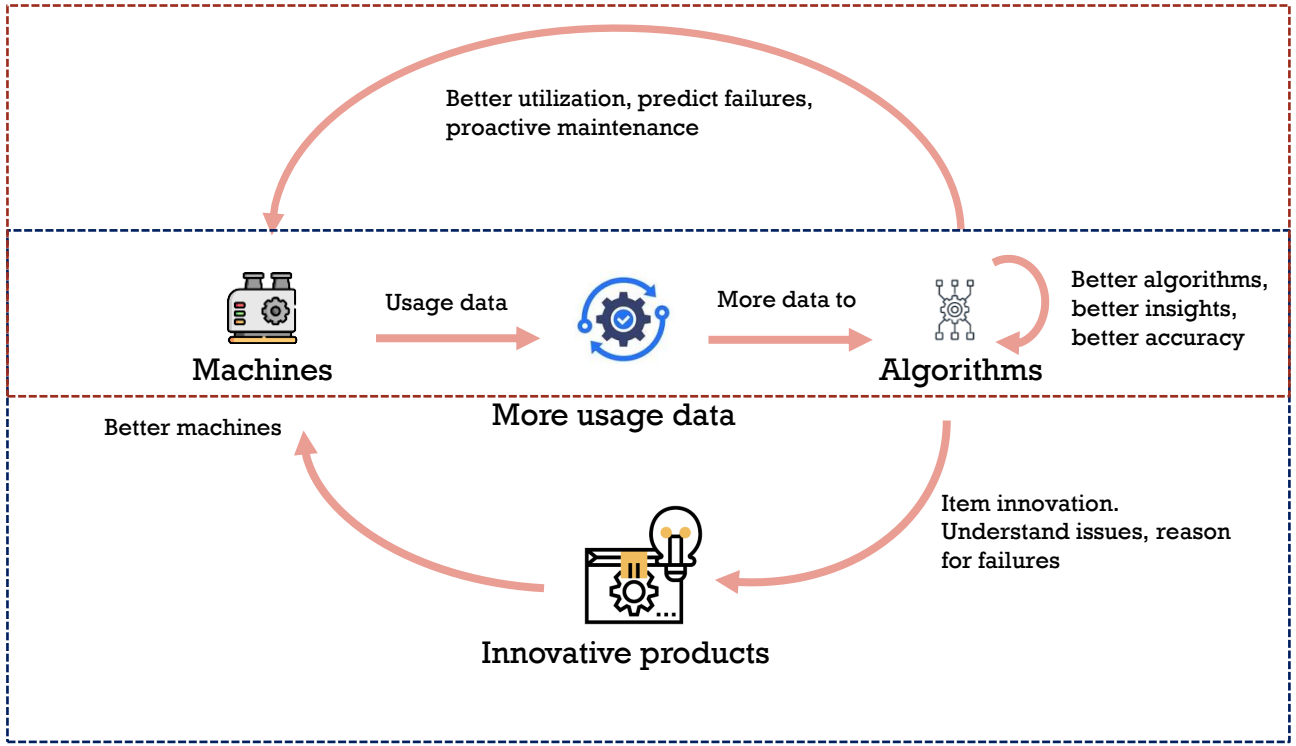
Memory
Adaptability

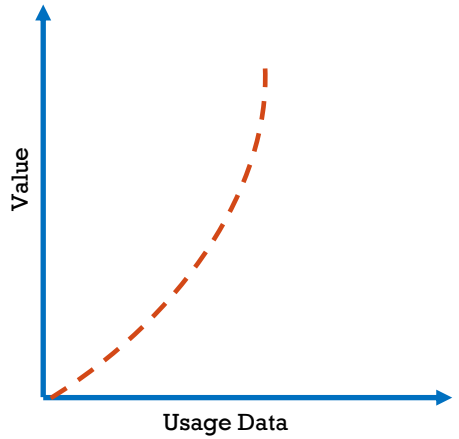
Tacit | Explicit | Implicit

Apply

Think, plan & reason

Problem → Solution





Entities, usage, algos and value

Acquire → Learning

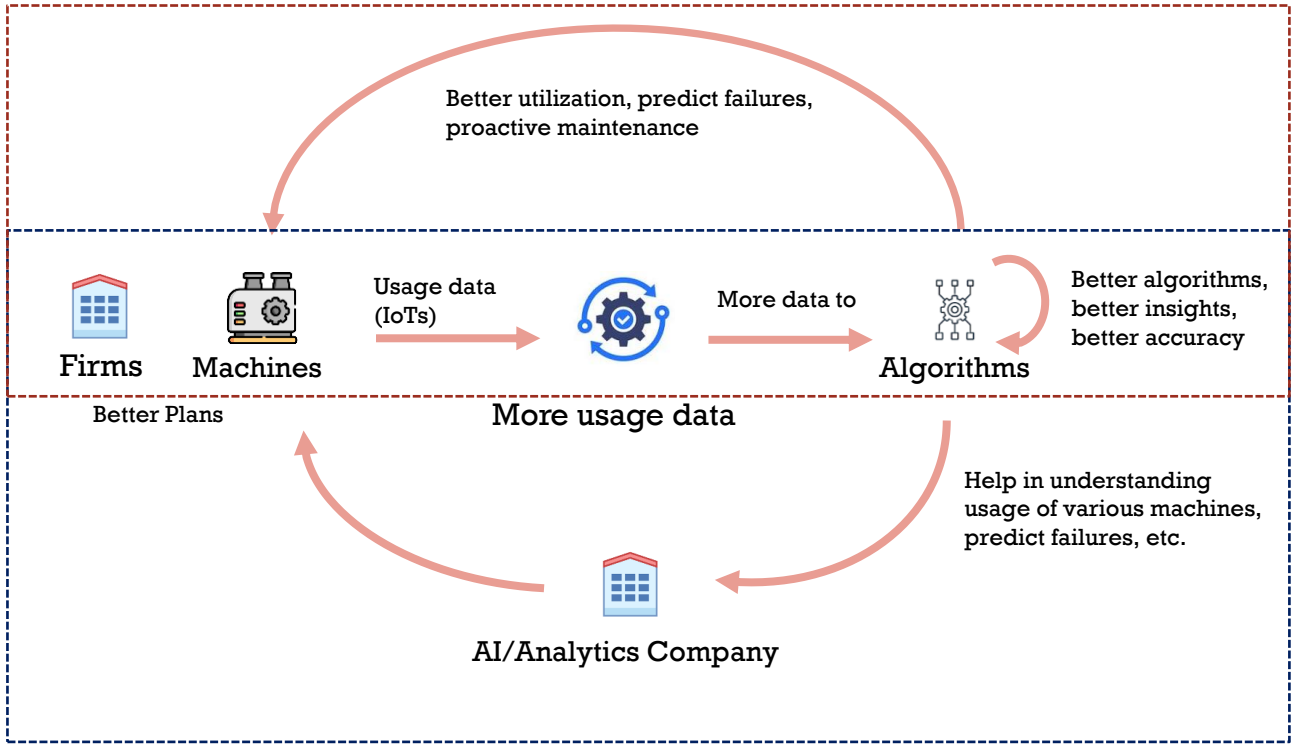
Memory
Adaptability

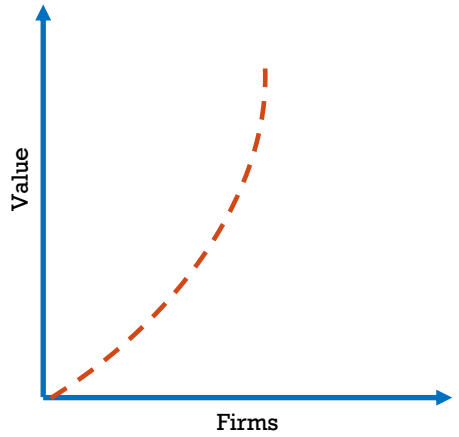
Tacit | Explicit | Implicit

Apply

Think, plan & reason

Problem → Solution





Entities, usage, algos and value

Acquire → Learning

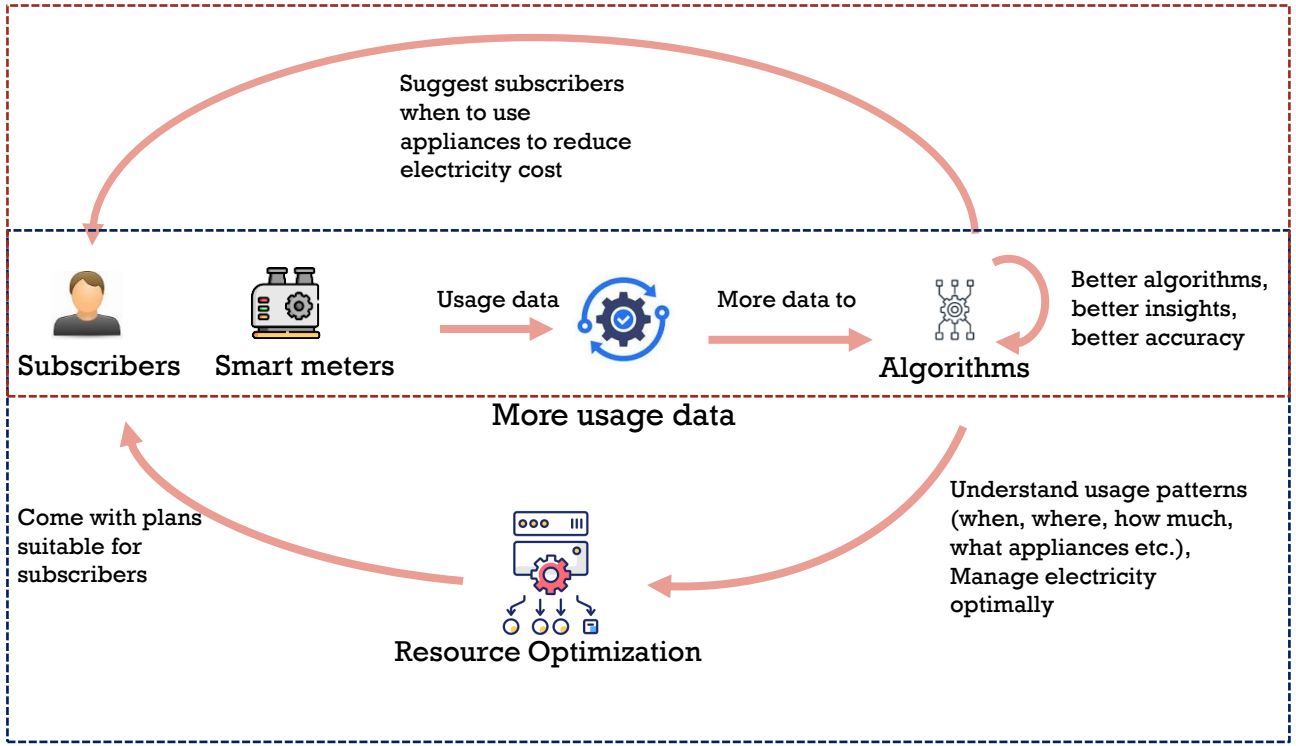
Memory
Adaptability

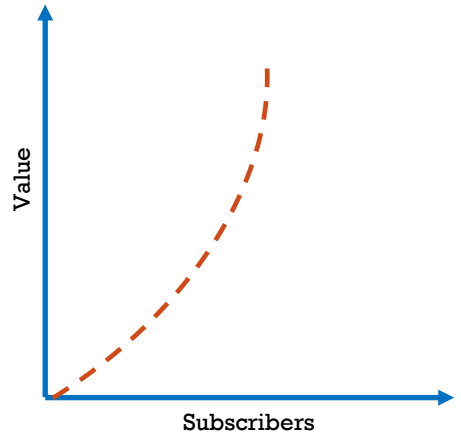
Tacit | Explicit | Implicit

Apply

Think, plan & reason

Problem → Solution





Metadata and value-adding data (information)

Acquire Learning

Memory
Adaptability

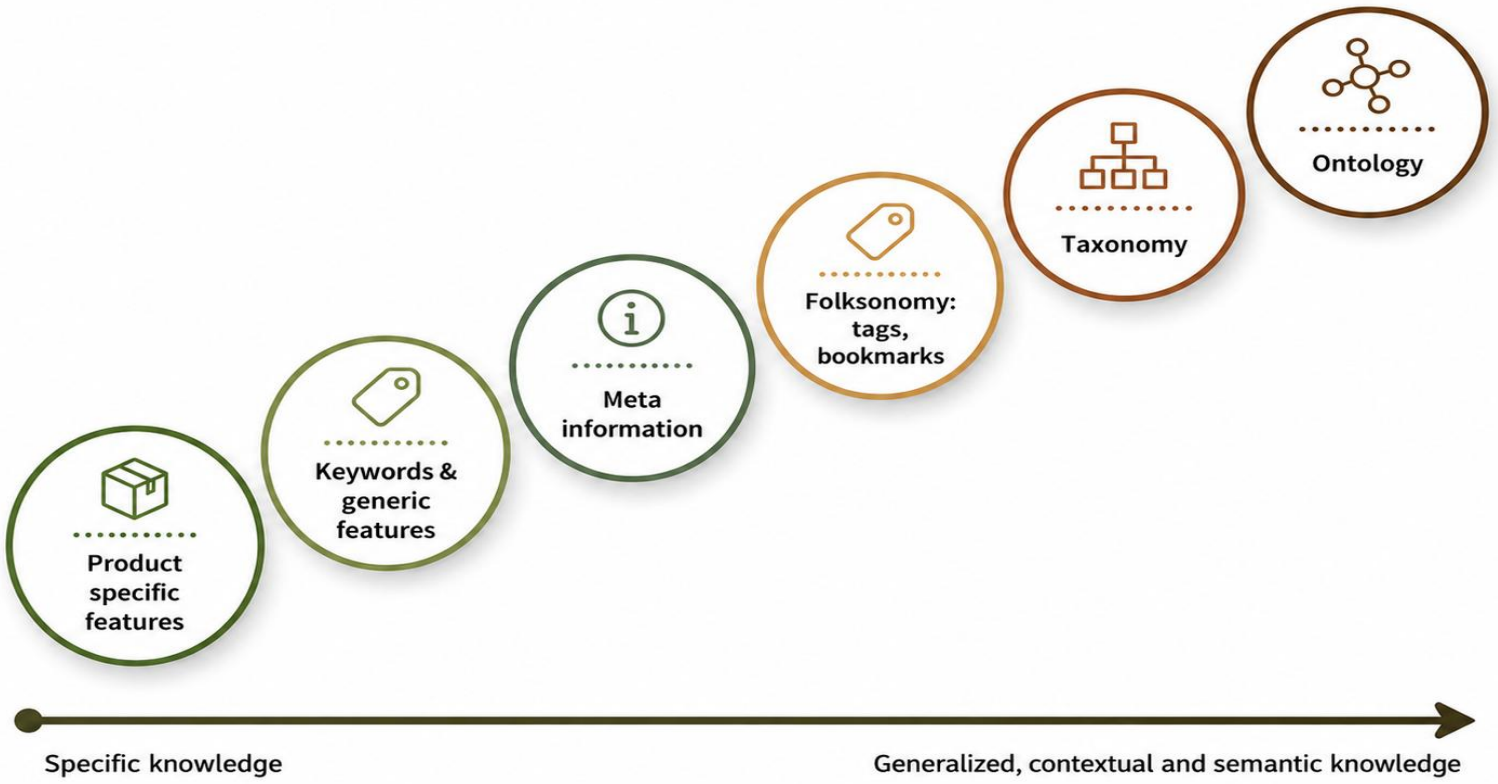
Tacit | Explicit | Implicit

Apply

Think, plan & reason

Problem

Solution



Adding knowledge to existing core entities value adds to core entities greatly, helps in building contextual, customized and personalized AI models.

Modelling general data entities

Acquire → Learning

Memory
Adaptability

Tacit | Explicit | Implicit

Apply

Think, plan & reason

Problem → Solution

Creating generic entities

Entity can be item (service, product, offer, event, campaign, content, etc.), party (customer, segment, provider) etc.

Meta-data and knowledge about entity (e.g. product features, keywords, genre, etc)

Entity

Features
Keywords
Ontologies
Behaviour & intent
Schedule

Profile can be global or local (specific to category, location and time).

Meta-data & knowledge
Calendar/ Inventory
Activities
Demographics
Preferences

General or specific behaviour (patterns) and intent of entity: e.g. what a customer does, when, where, how much, ratings received etc.

Availability of entity (e.g. item inventory or schedule of a customer: when and where?)

Domain model

Business logic: profiles are generated based on this model

Metadata: example : movie

Acquire → Learning

Memory
Adaptability

Tacit | Explicit | Implicit

Apply

Think, plan & reason

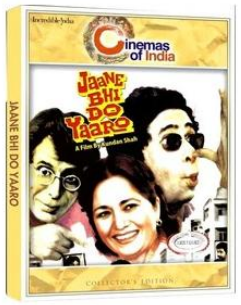
Problem → Solution

Such metadata will make AI to understand what customer likes: category, language, actors, old or new movies, etc.

Connecting dots: language customer knows….

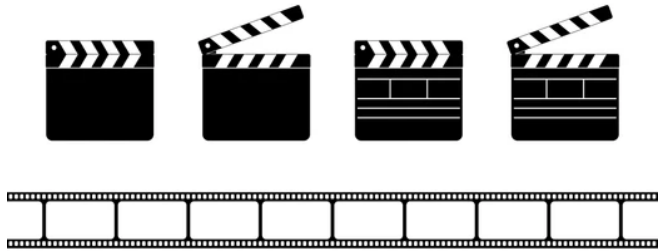
Details of Movie – Jaane Bhi Do Yaaro (Collector's Edition)

Actor:	Naseeruddin Shah, Om Puri, Ravi Baswani, Satish Shah
Director:	Kundan Shah
Writer:	Ranjit Kapoor, Satish Kaushik
Producer:	National Film Development Corporation
Title:	Jaane Bhi Do Yaaro (Collector's Edition)
Category:	Movies
Format:	DVD
Language:	Hindi
Year:	1983
Manufacturer:	Shemaroo
Running Time:	110 minutes
Video Encoding:	NTSC All Regions
Number of Discs:	2
Subtitles:	English, Arabic
Genre:	Comedy



Knowing script with time stamp will help to understand what viewer likes in movie.

Can also help creating new content by extracting relevant contents: action scenes, songs, etc. (using vision/audio ML it is possible in automated way)



What more data can be added?

- Singers
- Music Directors
- Plot
- Songs
- …

What data can be augmented through ML?

- Sentiment
- Average ratings
- Preferred time/language
- Age profile (if demographics available)

Metadata: example : recipe: Kadai Paneer

Acquire → Learning

Memory
Adaptability

Tacit | Explicit | Implicit

Apply

Think, plan & reason

Problem → Solution

Ingredients

- Paneer - 300 gms
- Oil - 3-4 tbsp
- Ginger-Garlic Paste - 1 tsp
- Cumin Seeds - 1 tsp
- Ginger - 1 pc (an inch long) (chopped)
- Garlic - 2-3 pcs (chopped)
- Onion - 2 pcs (large) (chopped)
- Turmeric Powder - 1/2 tsp
- Red Chili Powder - 1 tsp
- Coriander Powder - 1 tsp
- Tomato Puree - 2 pcs (large) (pureed)
- Salt - 1 tsp
- Kadai Masala - 2 tsp
- Coriander - as needed

Category: Veg/North Indian/Punjabi

Tags: Spicy, Main course

Nutrient	Value
Yield Quantity	400 gms
Calories	358
Protein	36 g
Carbohydrate	26 g
Fiber	6 g
Fat	16 g
Cholesterol	0 mg
Sodium	10 g
Potassium	763 mg

Detail	Value
Preparation Time	10 min
Cook Time	30 min
Ready Time	45 min
Yield	400 gms
Servings	3–4

Such metadata will make AI to understand what customers from multiple perspective: 1> what category 2> what ingredients, 3> nutrient preferences (healthy, non-healthy), 4> if cook then time preference etc.

Meta-data: example

Acquire → Learning

Memory
Adaptability

Tacit | Explicit | Implicit

Apply

Think, plan & reason

Problem → Solution

Mechanical & Electrical Specifications

Attribute	Sample Value
Spindle Power	22.4 kW
Max Spindle Speed	8,100 RPM
Axis Configuration	3-Axis (X, Y, Z)
Tool Changer Capacity	24 tools
Drive Type	Servo Motors
Rated Voltage	415V, 3-Phase
Control System	Haas CNC Controller

Sensor & IoT Metadata

Sensor Type	Location	Sampling Rate
Vibration Sensor	Spindle housing	5 kHz
Temperature Sensor	Spindle bearing	1 Hz
Current Sensor	Servo motor line	100 Hz
Acoustic Sensor	Tool head	44 kHz
RPM Encoder	Spindle shaft	Event-based

Machine Identification Metadata

Attribute	Sample Value
Machine ID	CNC-MILL-VM-001
Machine Type	Vertical CNC Milling Machine
Manufacturer	Haas Automation
Model	VF-2
Year of Installation	2019
Location	Plant 1 – Shop Floor A
Asset Criticality	High
Operating Mode	Continuous (3 shifts)

Operational Metadata

Attribute	Sample Value
Avg Daily Runtime	18 hours
Avg Load Factor	72%
Duty Cycle	High
Operating Temperature Range	18–45°C
Coolant Type	Water-based
Typical Job Type	Precision metal machining

Meta-data: adds context to multi-modal data

Acquire → Learning

Memory
Adaptability

Tacit | Explicit | Implicit

Apply

Think, plan & reason

Problem

Solution

1 Deep ML to understand pattern is valid or not

2 Understand Semantics (inside extracted components): corrections, validations etc.

Label Pattern e.g. DOB **AAA** for auto correction: 00B corrected to DOB
Valid values e.g. Gender: MALE/FEMALE/…
Sequence of information e.g. name, dob and then gender
Value patterns e.g. Aadhar no: **9999 9999 9999**
Font e.g. BHIM Aadhaar is Gotham family.
Validation rule: e.g. PVC card has 2 Aadhar numbers & should be same
Identification of doc e.g. Government of India + number pattern etc.
Fuzzy matching e.g. Name style

Rule Engine

RULES

Auto populati on or cross check

3 Verification

Real/ Fake/ Valid/ Verified

ML should be looked in context rather than silos. If number plate is read in silos would never know somebody is misusing valid number!

1 Deep ML to understand pattern is valid

Should be validated with type of vehicle, colour, logo, model etc.

2 Semantics

RULES

Number plate rules, state, Type of vehicles and series

3 Verification

Stolen / Real/ Fake/ Valid/ Verified

Reading number plate using OCR in silos won't help to understand if somebody is misusing it, unless it is looked in the context of vehicle to which the number plate has been issued. At least some deterrence.

Deriving meta-data: using image processing

Acquire → Learning

Memory
Adaptability

Tacit | Explicit | Implicit

Apply

Think, plan & reason

Problem → Solution

LLM

```
{
  "brand": "Skechers",
  "category": "Hiking Shoes/Trail Shoes",
  "primary_color": "Olive/Green",
  "accent_color": "Black",
  "water_repellent": true,
  "insole": "Goga Mat",
  "outsole": "Goodyear Rubber",
  "design_series": "D-Lux Trekker",
  "closure_type": "Lace-Up",
  "comfort_technology": "D-Lux Comfort",
  "typical_uses": ["Outdoor Activities", "Casual Wear"]
}
```

Visual search via vision-> Text-> SQL

Value-adding data: Similarity domain knowledge

Acquire → Learning

Memory
Adaptability

Tacit | Explicit | Implicit

Apply

Think, plan & reason

Problem → Solution

	C	C++	Java
C	1	0.8	0.5
C++	0.8	1	0.6
Java	0.5	0.6	1

	Java	C++	Visual Basic
Java	1	0.6	0.4
C++	0.6	1	0.3
Visual Basic	0.4	0.3	1

similarity matrix overall from the skill-sets point of view

Gen AI can help to build such meaningful knowledge based on the domain context. This is from software context to match the skill set required v/s available.

Similarity between 0-1. 1 similar 0 no similarity, symmetric v/s asymmetric similarity, higher is better, lower is better,....

Value-adding data: Similarity domain knowledge

Acquire → Learning

Memory
Adaptability

Tacit | Explicit | Implicit

Apply

Think, plan & reason

Problem → Solution

	Computer Engineering	Mechanical Engineering	Information Technology	Civil Engineering
Computer Engineering	1	0.3	0.8	0.2
Mechanical Engineering	0.3	1	0.3	0.5
Information Technology	0.8	0.3	1	0.2
Civil Engineering	0.2	0.5	0.2	1

Similarity between 0-1. 1 similar 0 no similarity

Value-adding data: Domain values to meaningful numbers

Acquire → Learning

Memory
Adaptability

Tacit | Explicit | Implicit

Apply

Think, plan & reason

Problem → Solution

Designation	Numeric Value
CGM (Chief General Manager)	25
GM (General Manager)	20
DGM (Deputy General Manager)	15
AGM (Assistant General Manager)	11
Chief Manager	8
Manager	4
Deputy Manager	0

Designation	Numeric Value
Ph.D.	22
Post-Graduate	18
Graduate	16
Diploma	15
HSC	12
SSC	10

Value-adding data: augmenting data

Acquire → Learning

Memory
Adaptability

Tacit | Explicit | Implicit

Apply

Think, plan & reason

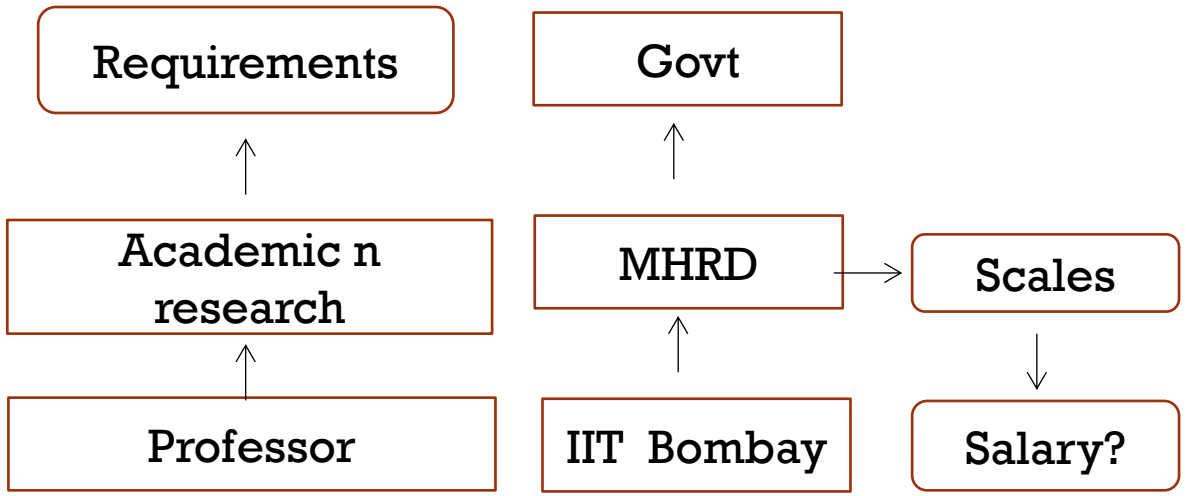
Problem

Solution

Rajendra M. Sonar

Professor of Information Systems/Technology,
Shailesh J Mehta School of Management,
Indian Institute of Technology
Powai, Mumbai-400076, India
Ph. +91-22-2576 7741 (O) Intercom Ext: 7741(O), 8741(R)
Email: rm_sonar [AT] iitb.ac.in
Skype: raj.m.sonar

- Designation
- Organization
- Location



Value-adding data: connecting data

Acquire → Learning

Memory
Adaptability

Tacit | Explicit | Implicit

Apply

Think, plan & reason

Problem → Solution

Raheja Builders → Builder → Real estate

Loan Products → Home Loan → Real estate

There is connect between a transaction – cross selling opportunity

Knowledge?

Acquire → Learning

Memory
Adaptability

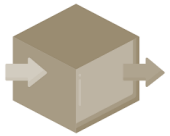
Tacit | Explicit | Implicit

Apply

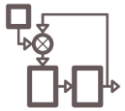
Think, plan & reason

Problem → Solution

Rules | Meta-data | Code | Knowledge Graphs

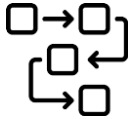


Black-box Models



Equations | Models

Decision Trees



Associations

Centralized | Decentralized (group->individual entity) | Accuracy | Understandability

Key takeaways

AI leadership is about combining technology understanding, data readiness, human judgement and end-to-end redesign.

AI is a spectrum
Semantic, traditional, generative and agentic AI solve different parts of the problem.

Data + knowledge are backbone
AI-ready data, metadata, generic entities, domain knowledge and clean process understanding matter most.

Gen AI is powerful, not universal
Use it for language, content, reasoning support and multimodal interfaces — with limits in mind.

AI-first mindset
data • knowledge • tools • governance

Agentic AI needs orchestration
End-to-end automation comes from coordinated tasks, systems, workflows and controls.

AI-first means redesign or add layers
Move beyond tool adoption: redesign workflows, decisions, teams and governance around AI. Bottom-up approach with semantic knowledge can help greatly.

Humans move up the value chain
Human role shifts to judgement, framing, expertise, validation and responsible use.