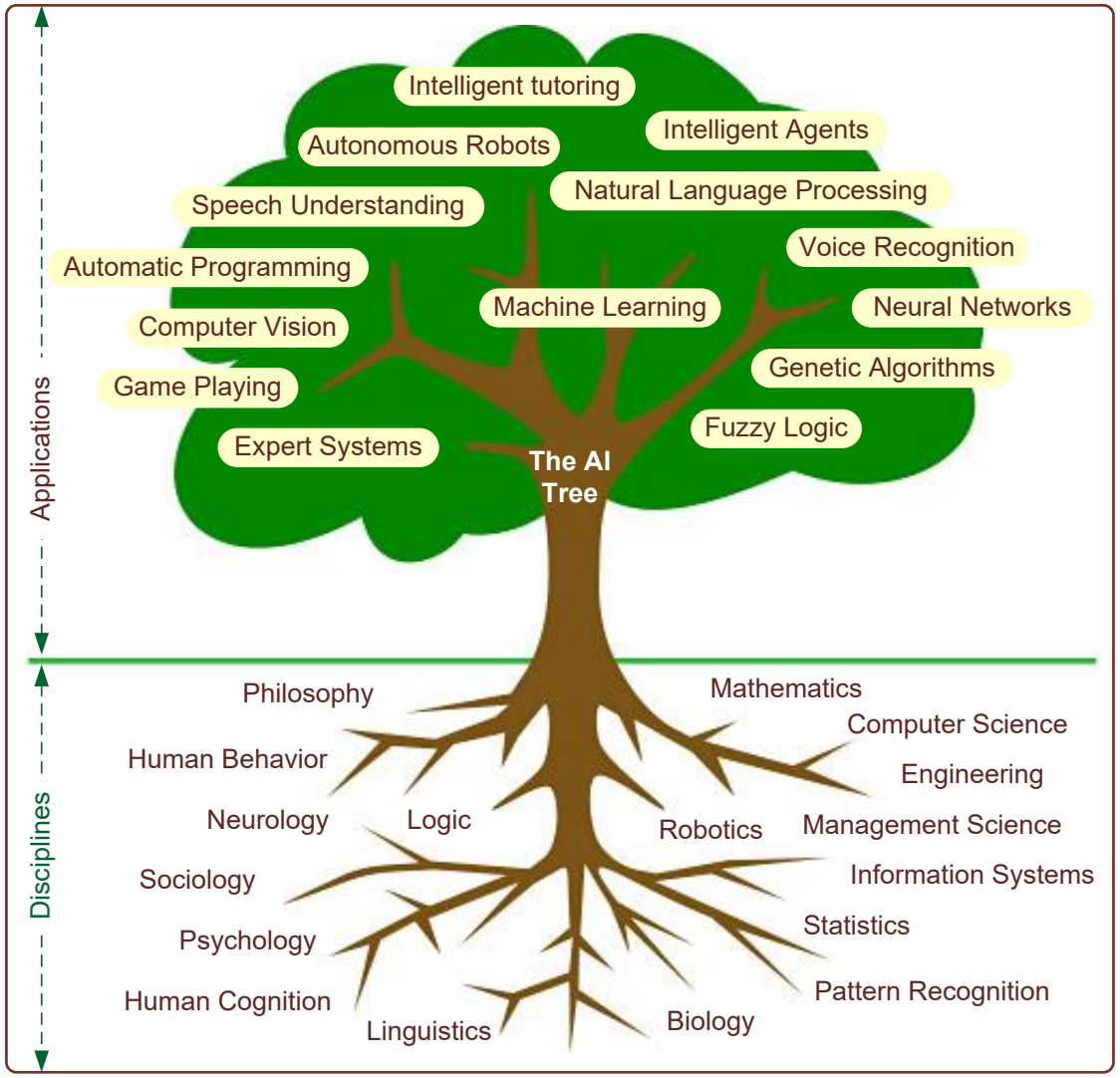
AI: What makes machines intelligent?



Source: Turban et al. (2011), Decision Support and Business Intelligence Systems

Code — Normal Programming Code

- Smart coders can include lot of intuitive and common sense for **operational intelligence** while writing the code in UI, workflows, processes etc.

Algos — Classical AI algorithms

- **Programs/Algorithms** to reach/search solutions e.g. min-max algorithm

KBS — Knowledge Based Systems

- Expert and Fuzzy Systems: model explicit **human** domain **knowledge** and solve problems using that expertise

NLP — Natural Language Processing

- Read, understand and interpret text/audio in human languages and **making sense** of what is going on

ML — Artificial Neural Networks/ML/Data Mining

- Data driven, **learn** relationships, patterns, associations etc. ANNs model human brain

GA — Genetic Algorithms

- Based on evolution principle: **evolve** solution to the problem, classified as evolutionary computing

CBR — Case Based Reasoning

- Model human problem solving: remember, recall, adapt, reuse past **experiences,** lazy learning

MBR — Model Based Reasoning

- Expects things to work according to **modelled behaviour**. Monitors/detects deviation and helps in diagnosis

RL — Reinforcement Learning

- **Self learning systems**: learn problem solving on their own

GAN|LLMs — Generative AI | Large Language Models

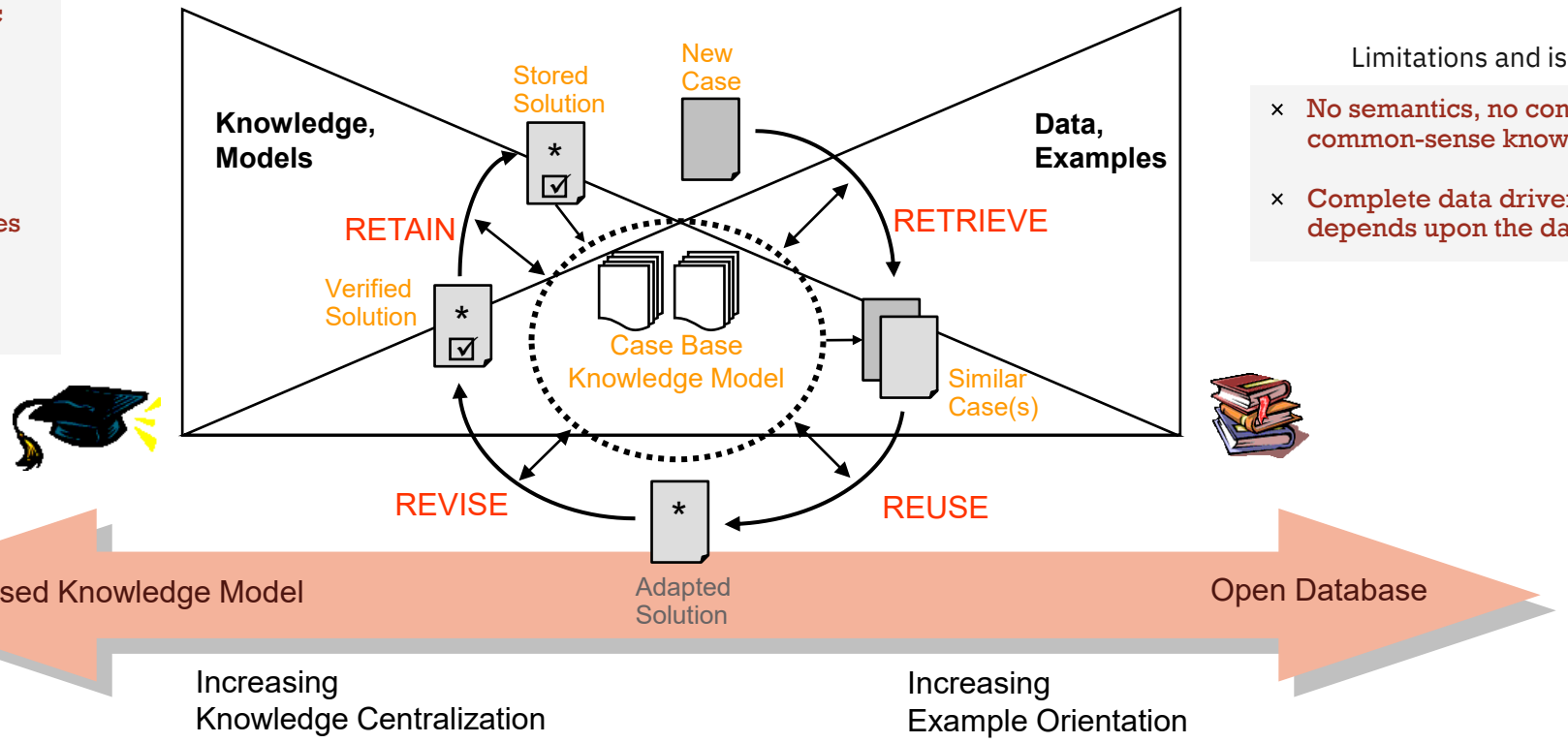
- Generates new contents, summarizes, translates,…

Together they solve many kinds of generic problems. Use effective combinations of hybrid of these technologies powered by LLMs

AI: Knowledge Driven and Data Driven Systems

Limitations and issues

- × Knowledge acquisition, modelling and knowledge engineering
- × Managing dynamic business logic and contexts
- × Modelling complex domain knowledge can be challenging
- × Mostly textual knowledge and no multimedia form like audio, images etc.
- × No data driven intelligence



Limitations and issues

- × No semantics, no contextual and common-sense knowledge
- × Complete data driven, accuracy depends upon the data

Expert Systems
(completely knowledge driven)

CBR Systems
(knowledge + data driven)

Databases
(completely data driven)

Classical Intelligent Agents!

Intelligent Agents: Theory and Practice

Michael Wooldridge
Department of Computing
Manchester Metropolitan University
Chester Street, Manchester M1 5GD
United Kingdom
M.Wooldridge@doc.mmu.ac.uk

Nicholas R. Jennings
Department of Electronic Engineering
Queen Mary & Westfield College
Mile End Road, London E1 4NS
United Kingdom
N.R.Jennings@qmw.ac.uk

Submitted to *Knowledge Engineering Review*, October 1994.
Revised January 1995.

Michael Wooldridge is a prominent computer scientist known for his work on intelligent agents, particularly in the field of multi-agent systems. He defines intelligent agents as computer systems situated in an environment and capable of flexible, autonomous action to meet design objectives, according to Jennings and Wooldridge (1998). His research focuses on the theoretical foundations, architectures, and languages for building intelligent agents, often employing techniques from computational logic, game theory, and social choice theory

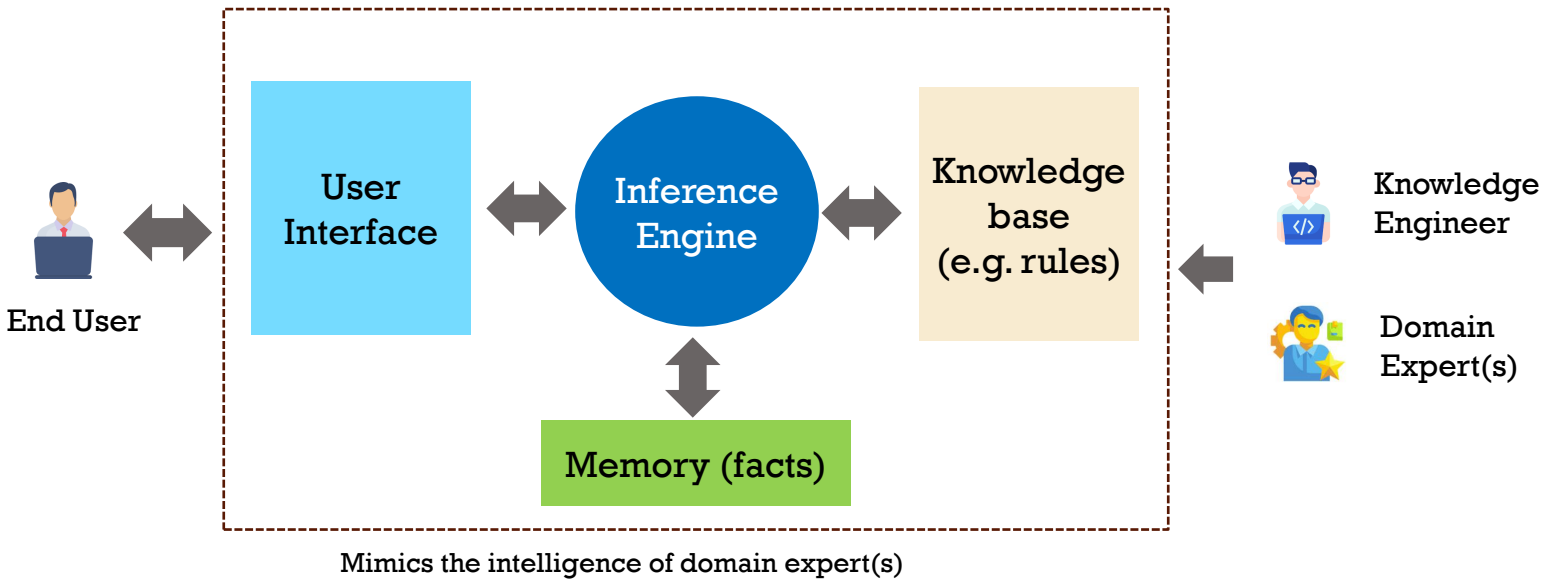
Abstract: The concept of an agent has become important in both Artificial Intelligence (AI) and mainstream computer science. Our aim in this paper is to point the reader at what we perceive to be the most important theoretical and practical issues associated with the design and construction of intelligent agents. For convenience, we divide these issues into three areas (though as the reader will see, the divisions are at times somewhat arbitrary). Agent theory is concerned with the question of what an agent is, and the use of mathematical formalisms for representing and reasoning about the properties of agents. Agent architectures can be thought of as software engineering models of agents; researchers in this area are primarily concerned with the problem of designing software or hardware systems that will satisfy the properties specified by agent theorists.

PDF
Intelligent Agents: Theory and Practice

Classical Intelligent Agents!

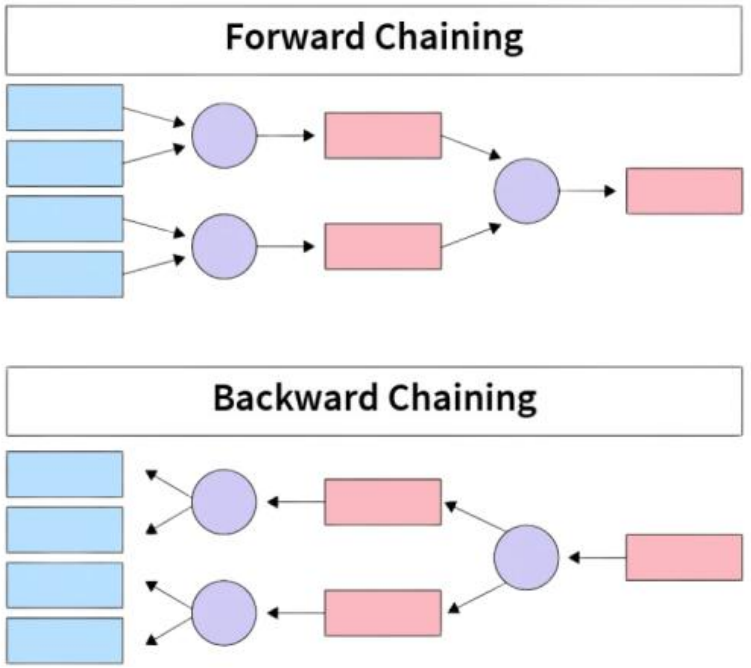
Expert system tech as enabler

Basic Expert System Components



Mimics the intelligence of domain expert(s)

Rule-based reasoning



Why expert systems?

- ✓ They work like intelligent and smart agents by mimicking intelligence of human experts.
- ✓ Inference engine and knowledge base are separate making it easier to manage dynamic business logic.
- ✓ Ask only relevant inputs based on the context and heuristics
- ✓ Rules are easier to understand. Outcomes can be explained (explainability of reasoning)

Limitations and issues with expert systems

- × Knowledge acquisition, modelling and knowledge engineering
- × Managing dynamic business logic and contexts
- × All knowledge cannot be accommodated in if-then-else rule format

Images from different sources are acknowledged

Classical Intelligent Agents, an example.

Intelligent Agent-Based Expert System Architecture for Generating Work Plan Marshalling Station

Abstract - The present approach for generating work plan is difficult to meet the dynamic and intelligent requirement, and it has not had the function of assistant decision yet. An agent that can simulate the reasoning process of human has the properties such as adaptability and intelligence, which can be used to solve the problems in making plan process. This paper has presented the initial architecture for an intelligent agent-based expert system for generating of work plan in marshalling station known as TAES-OPMY. IAES-OPMY can support the station controller's decisions on generating plan and in turn improve the validity, encash ratio and automatization of plan. Furthermore, it can speed the turnover of wagons and bring great profits.

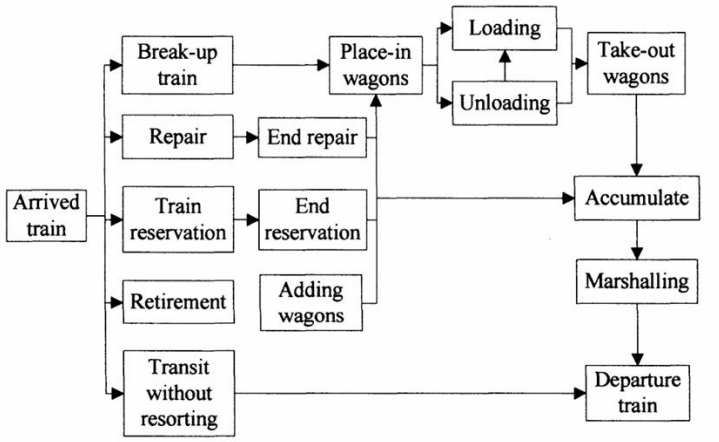


Fig. 2. Technique work operation process in marshalling station

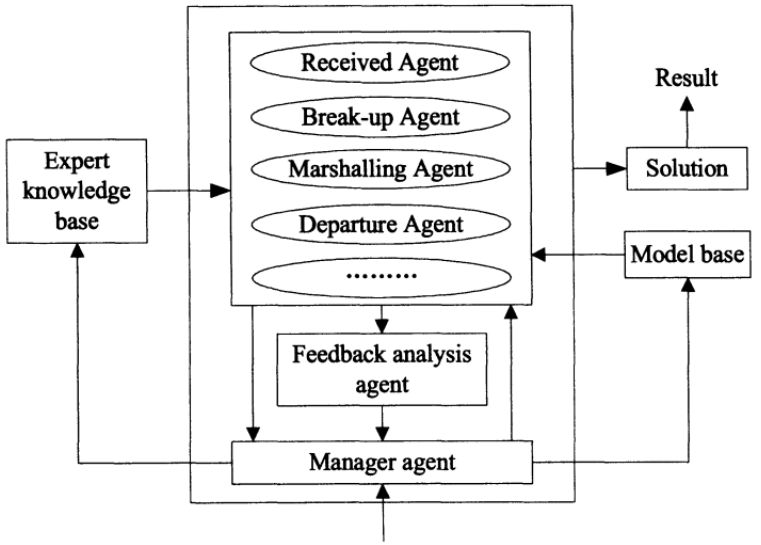


Fig. 3. Architecture of IAES-OPMY

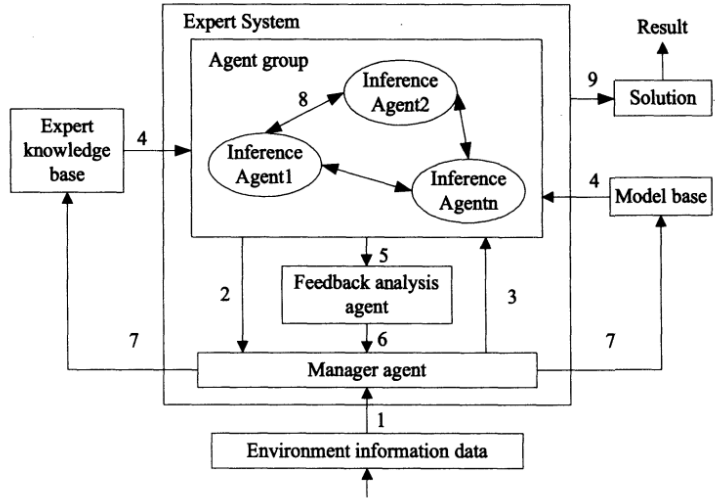
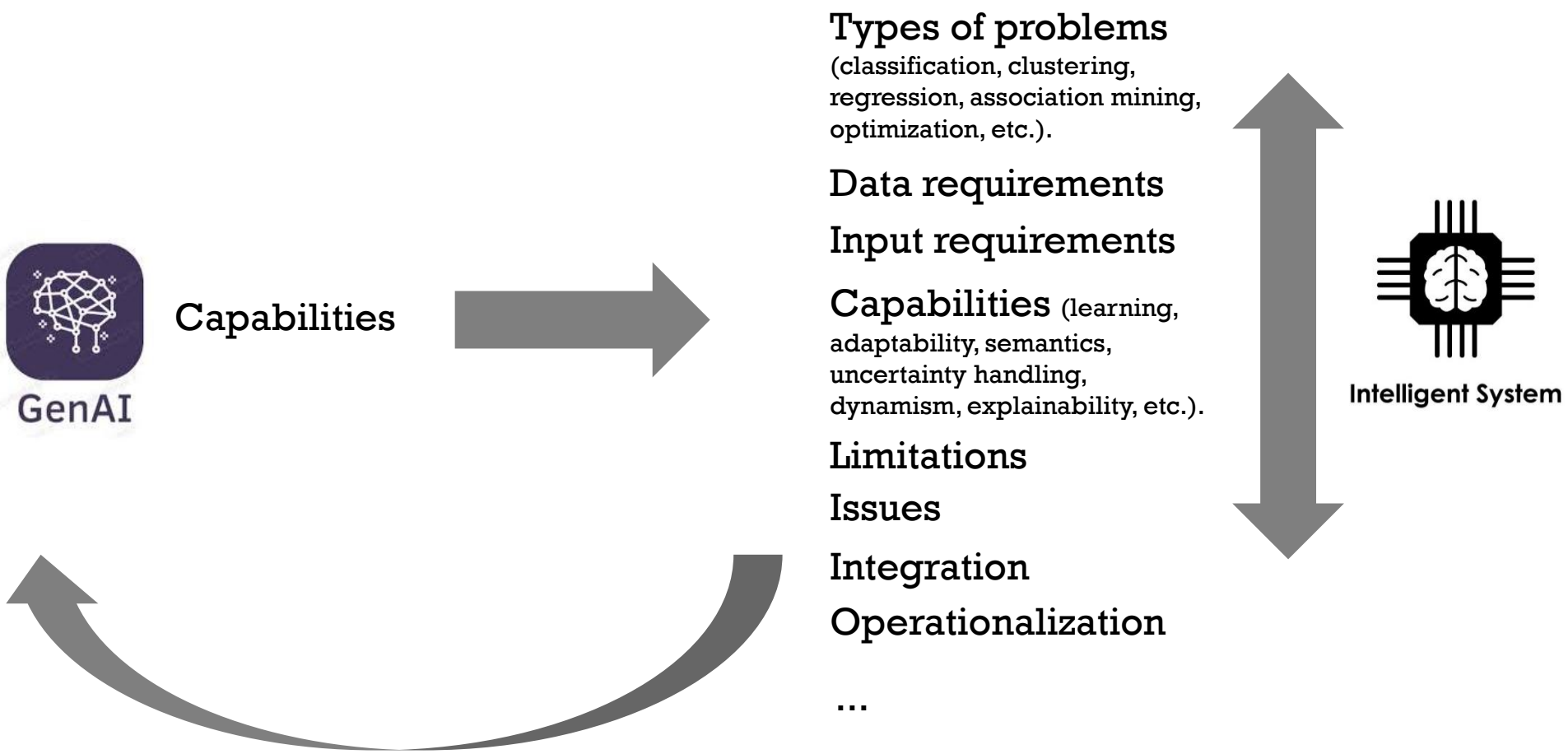


Fig. 1. IAES architecture

0-7803-9017-2/05/$20.00C2005IEEE (2005)

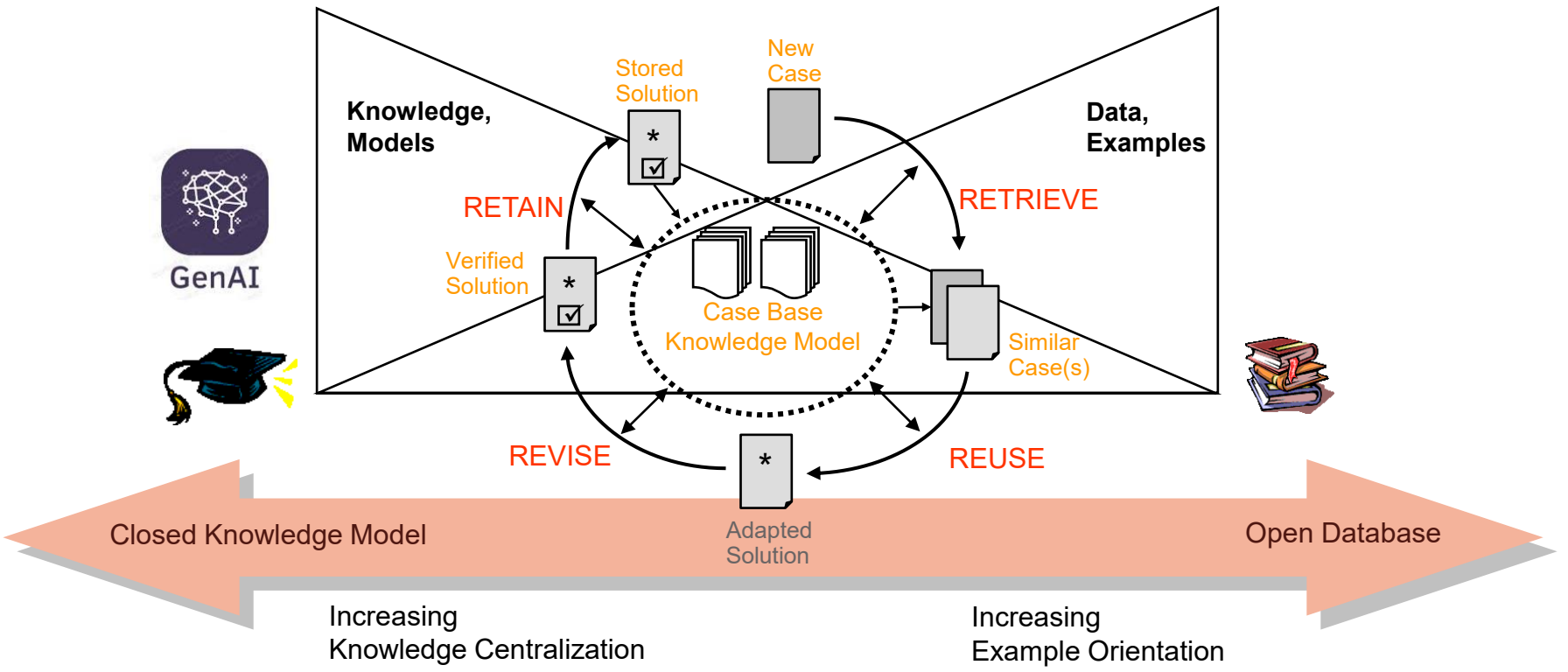
Making GenAI work for AI/ML Systems



GenAI can take care of issues and limitations of intelligent systems. It can help to integrate and operationalize these systems making it possible to leverage classical AI technologies in the current context.

Images from different sources are acknowledged

AI: Knowledge Driven and Data Driven Systems

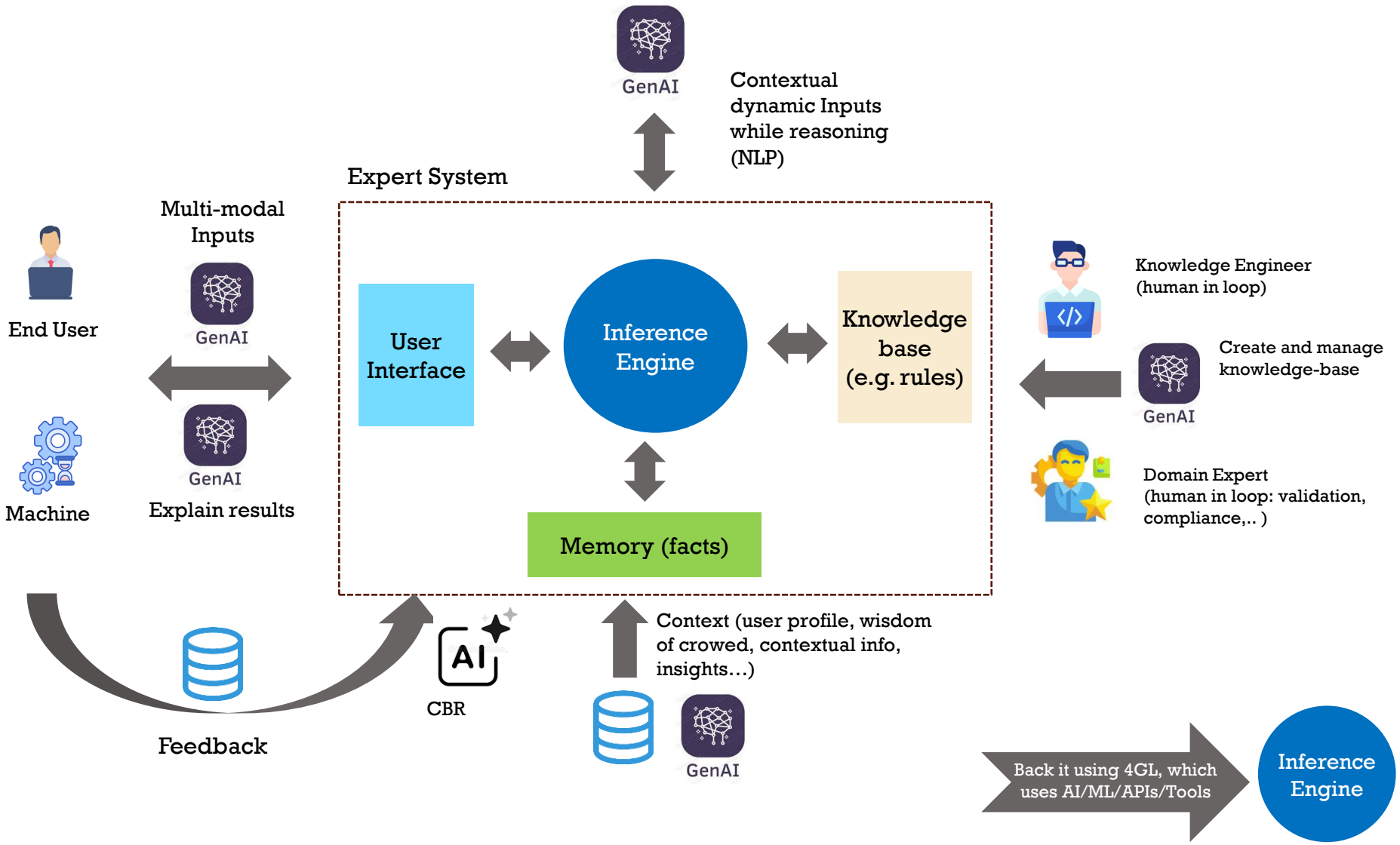


Expert Systems
(completely knowledge driven)

CBR Systems
(knowledge + data driven)

Databases
(completely data driven)

Framework for AI Agents/Agentic AI: Expert system as core tech*



Solutions back by expert system

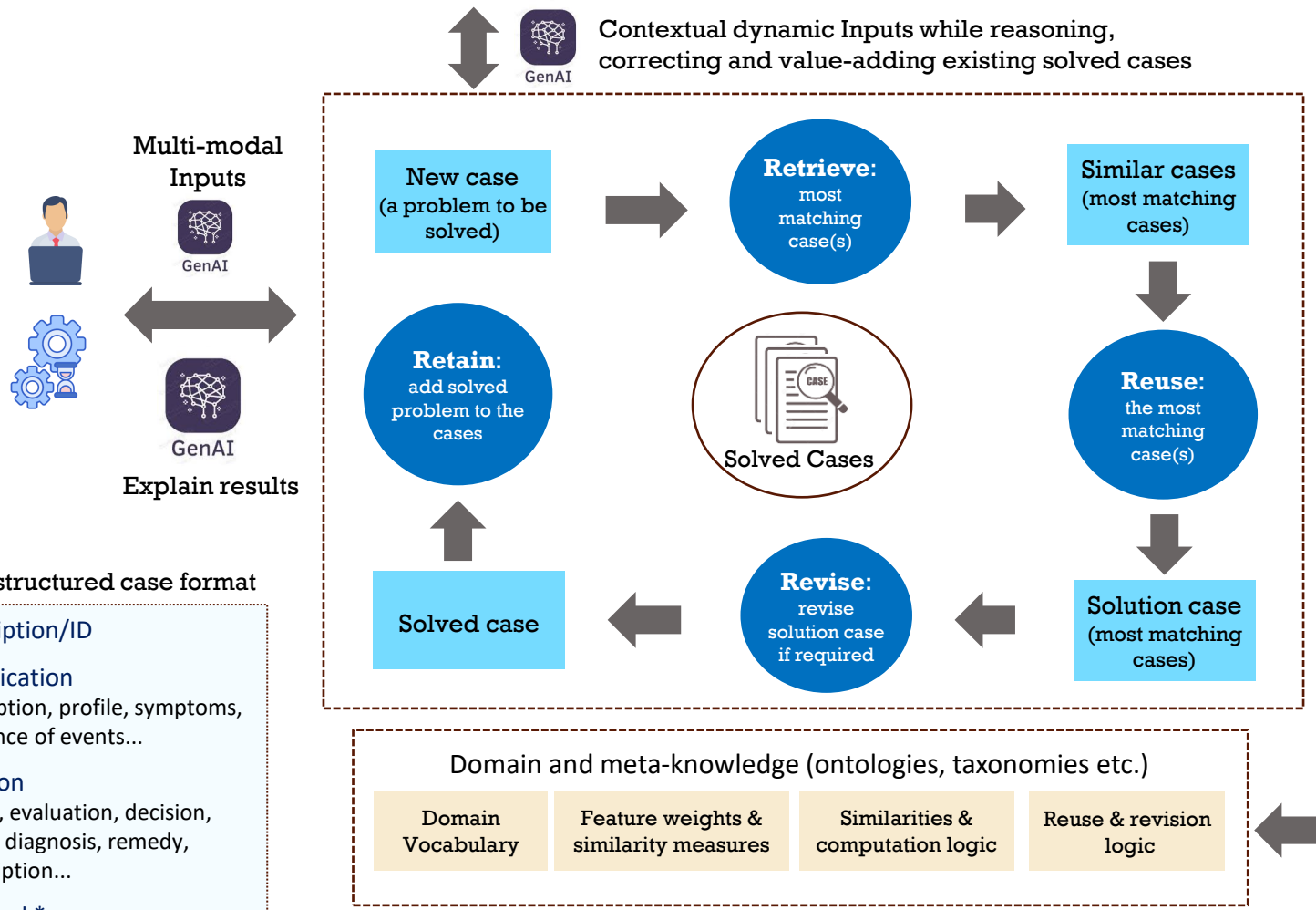
- ✓ They work like intelligent and smart agents by mimicking intelligence of human experts (no hallucinations and credible knowledge).
- ✓ Inference engine and knowledge base are separate making it easier to manage dynamic business logic.
- ✓ Ask only relevant inputs based on the context and heuristics
- ✓ Rules are easier to understand. Outcomes can be explained (explainability of reasoning), takes care of regulatory and compliance requirements.

Knowledge extraction using LLMs

- ✓ 80-90% of domain knowledge can be extracted from LLMs using rights prompts.
- ✓ Knowledge can be further validated, verified, corrected, augmented and parameterized by domain experts and knowledge engineers.
- ✓ This increases credibility of knowledge acquired without hallucinations.
- ✓ Can help in reducing dependence on LLMs subsequently.

* https://aicompetence.org/why-expert-systems-still-matter/

Making GenAI work for AI/ML Systems: Example – CBR (Case-Based Reasoning)



- ✓ Supports dynamism (lazy learning) combining knowledge (semantics) and data driven intelligence.
- ✓ Each feature can be modelled based on requirements.
- ✓ Models are easier to understand, customize and configure. Explainability.
- ✓ Can address large number of applications. Suitable for N=1 analysis and intelligence (e.g. hyper contextualization and personalization).

- × Domain knowledge acquisition, modelling and engineering
- × Initial efforts of creating case-base in case of unstructured cases, meta-data etc.
- × Run-time computation overhead as lazy learners

- ✓ LLMs can be used to acquire and model domain knowledge and logic which can be further validated, verified, corrected, augmented and parameterized by domain experts and knowledge engineers

Generic structured case format

- ❑ Description/ID
- ❑ Specification
  Description, profile, symptoms, sequence of events...
- ❑ Solution
  Action, evaluation, decision, repair, diagnosis, remedy, prescription...
- ❑ Feedback*

* Is not part of standard case and is very important to understand effectiveness of prescribed solution.

Domain and meta-knowledge (ontologies, taxonomies etc.)

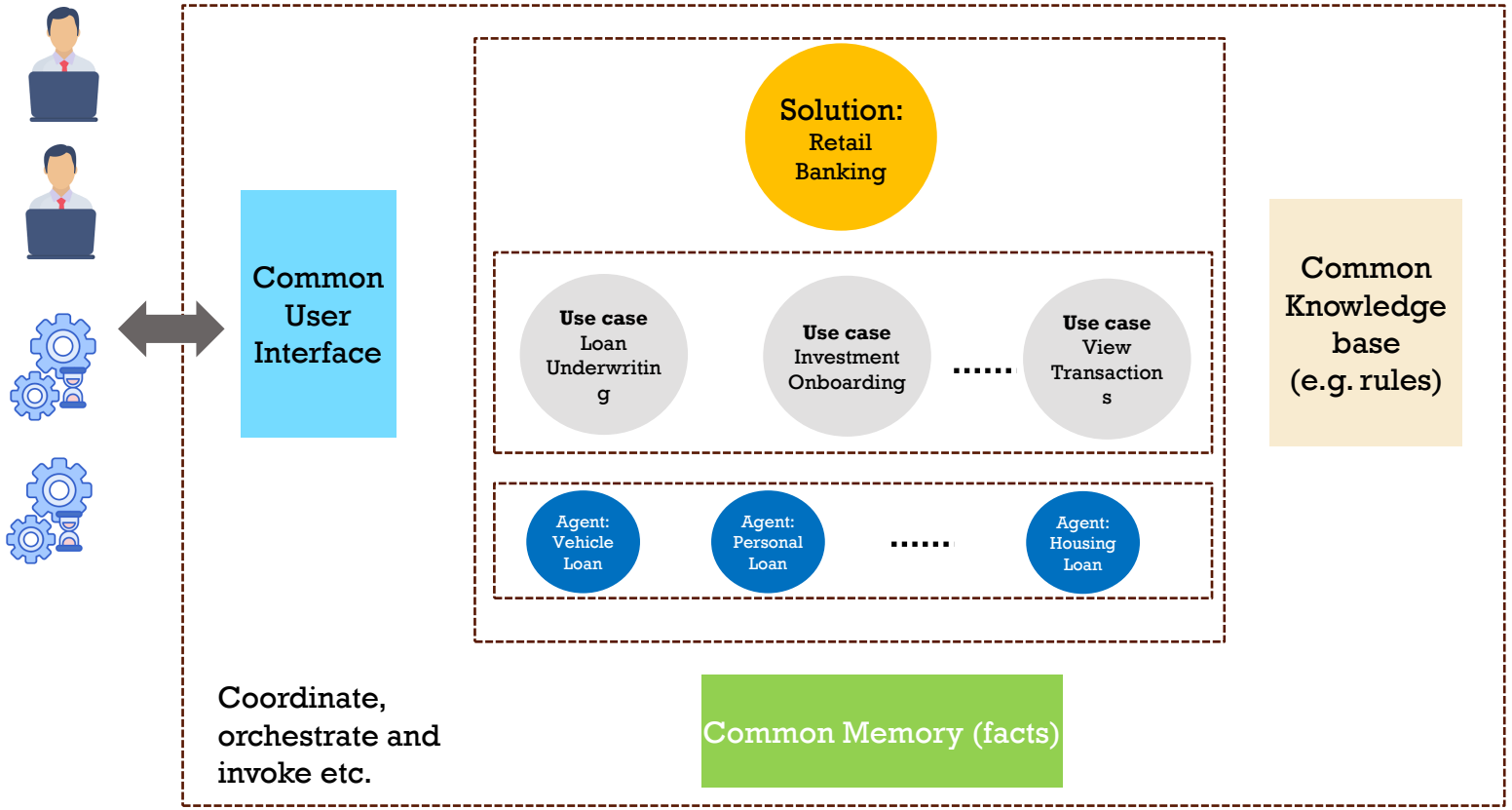
Domain Vocabulary	Feature weights & similarity measures	Similarities & computation logic	Reuse & revision logic

GenAI — Create and manage domain and meta-knowledge

Sample case formats

Credit status	Purchase behaviour	Churn analysis	Transaction monitoring	Product life	Monitoring/diagnostics
❑ Customer ID	❑ Customer ID	❑ Customer ID	❑ Transaction ID	❑ Shelf product ID	❑ Failure description
❑ Customer profile: Age, gender, income, education, ...	❑ Customer profile: Age, gender, income, education,...	❑ Transaction details: Last recharge, recharge amount, friends status, change call usage,..	❑ Purchase details: Place, merchant, time and date, items purchased, amount, payment type,..	❑ Product profile: name, category, price, store, location, discount offered, features,...	❑ Symptoms: S1(Q1-A1), S2(Q2-A2),...
❑ Repayment history: Good/Average...	❑ Items purchased: Item_1, Item_2, ...	❑ Churn status: Churned out	❑ Payment status: OK/Delayed/Fraud	❑ Shelf life: Days in shelf	❑ Fault/remedy: F1, R1 ...

Framework for AI Agents/Agentic AI



- Can mimic goal driven intelligence of human agents leveraging ML | GenAI capabilities
- Easier to manage dynamic business logic.
- Multiple agents can coordinate to each other to accomplish a goal (task)

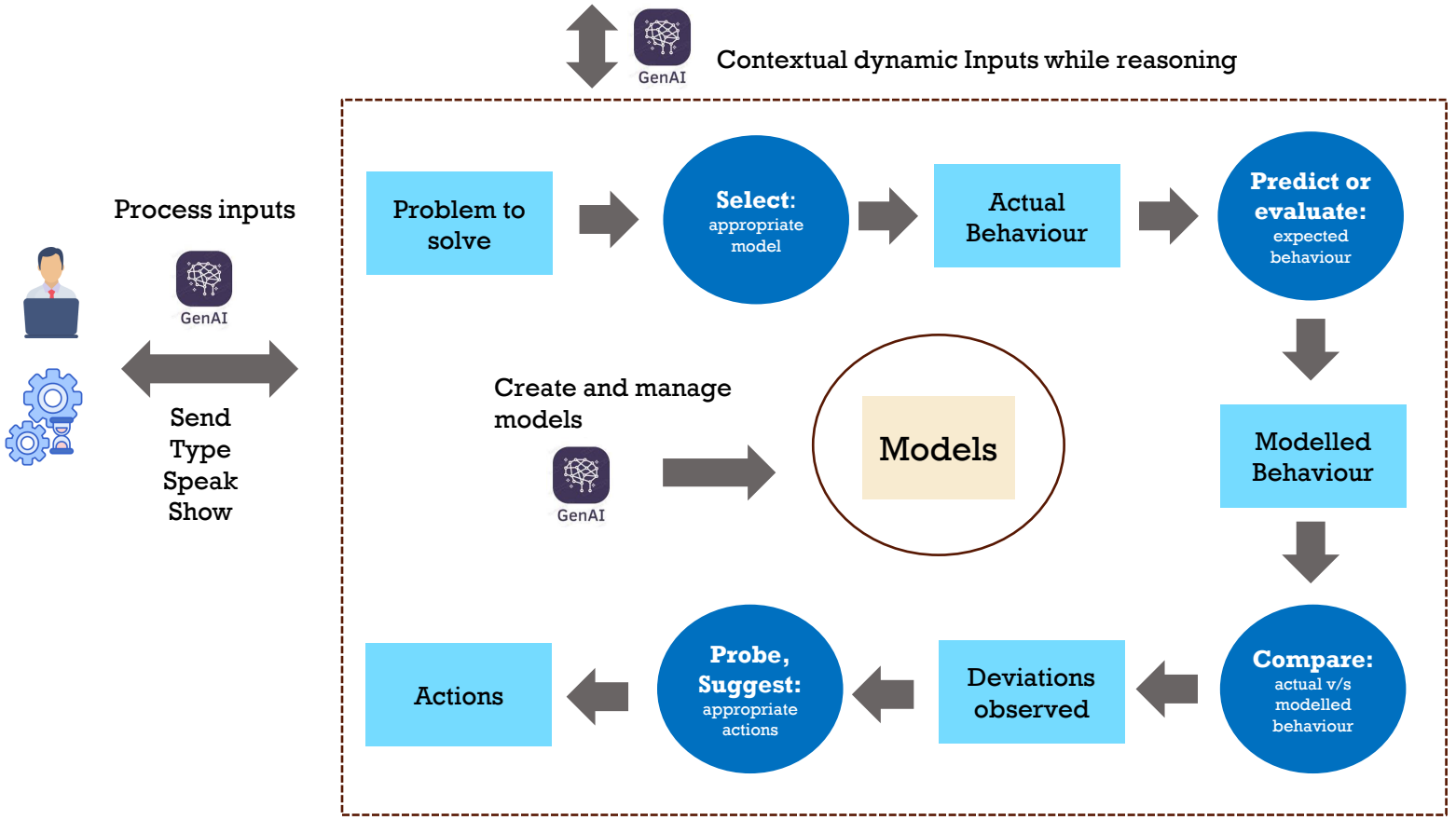
Backed by the state-of-the-art 4GL rule engine/framework supporting integration (and hybrids) of ML tasks, intelligent systems, APIs, tools, databases etc.

GenAI powered Solution Expert System. It can invoke use-cases or agents based on the context and work-flows. For example, if customer is looking for loan, then loan use case can be invoked which in turn can understand whether customer is looking for personal loan, personal loan agent can be invoked.

GenAI powered Use-case Expert System. It can invoke agents based on the requirements.

GenAI powered Agent Expert System. An agent can implement complex functionality and change the flow of use interactions based on current facts and context

Making GenAI work for AI/ML Systems: Example – MBR (Model-Based Reasoning)



- ✓ Models are widely used to represent real world systems (qualitative as well as quantitative)
- ✓ Models can be static or dynamic models. Dynamic models can be built for specific entities (e.g. customer spending model)
- ✓ Building model-based reasoning system is relatively easier if model is already available. Models can be built at N=1 level to understand behaviours of individual entities like customer risk or opportunity
- ✓ Can address applications like: diagnostics, early warning signals, fault analysis, auditing, repair, troubleshooting

- × Model building of real systems is not always easy and it is time consuming and needs explicit knowledge about real world system.
- × Qualitative models may need domain knowledge to build

Images from different sources are acknowledged