CS 301 : Artificial Intelligence & Expert Systems

Expert Systems

- Designed to function similar to a human expert operating within a specific *problem domain*

- Used to:
  - Provide an answer to a certain problem, And / or
  - Clarify uncertainties where normally a human expert would be consulted

- Often created to operate with humans, working within the given problem domain, rather than as a replacement for them

Important Components

- Knowledge Base
  - Stores knowledge used by the system, usually represented in a formal logical manner
- Inference Engine
  - Defines how existing knowledge may be used to derive new knowledge

Knowledge Representation

- we use a simple If … Then … consequence relation using English semantics

- ie: If [it is raining] Then [I should wear a coat]
  - [it is raining] is the **antecedent** of the relation
  - [I should wear a coat] is the **consequent** of the relation

- Facts can be understood as consequence relations with an empty antecedent
  - ie: “If [] Then [it is raining]” is equivalent to the fact that [it is raining]

Inferring New Knowledge

k1: If [it is raining] Then [I should wear a coat]

k2: [it is raining]

result: [I should wear a coat]

- New knowledge can be constructed from existing knowledge using inference rules

- For instance, the inference rule modus ponens can be used to derive the consequent of a consequence relation, given that the antecedent is true

Goal Directed Reasoning

- I should wear a coat?

  This would read easier in English as “should I wear a coat”, but we want to use the same propositional symbol as is in our knowledge base

- Inference rules are applied to knowledge base in order to achieve a particular goal

- The goal in an expert system is formed as a question, or query, to which we want the answer

MYCIN

History

- Thesis Project by Shortliffe @ Stanford University
- Davis, Buchanan, van Melle, and others
  - Stanford Heuristic Programming Project
  - Infectious Disease Group
- Project Spans a Decade
  - Research started in 1972
  - Original implementation completed 1976
  - Research continues into the 80's

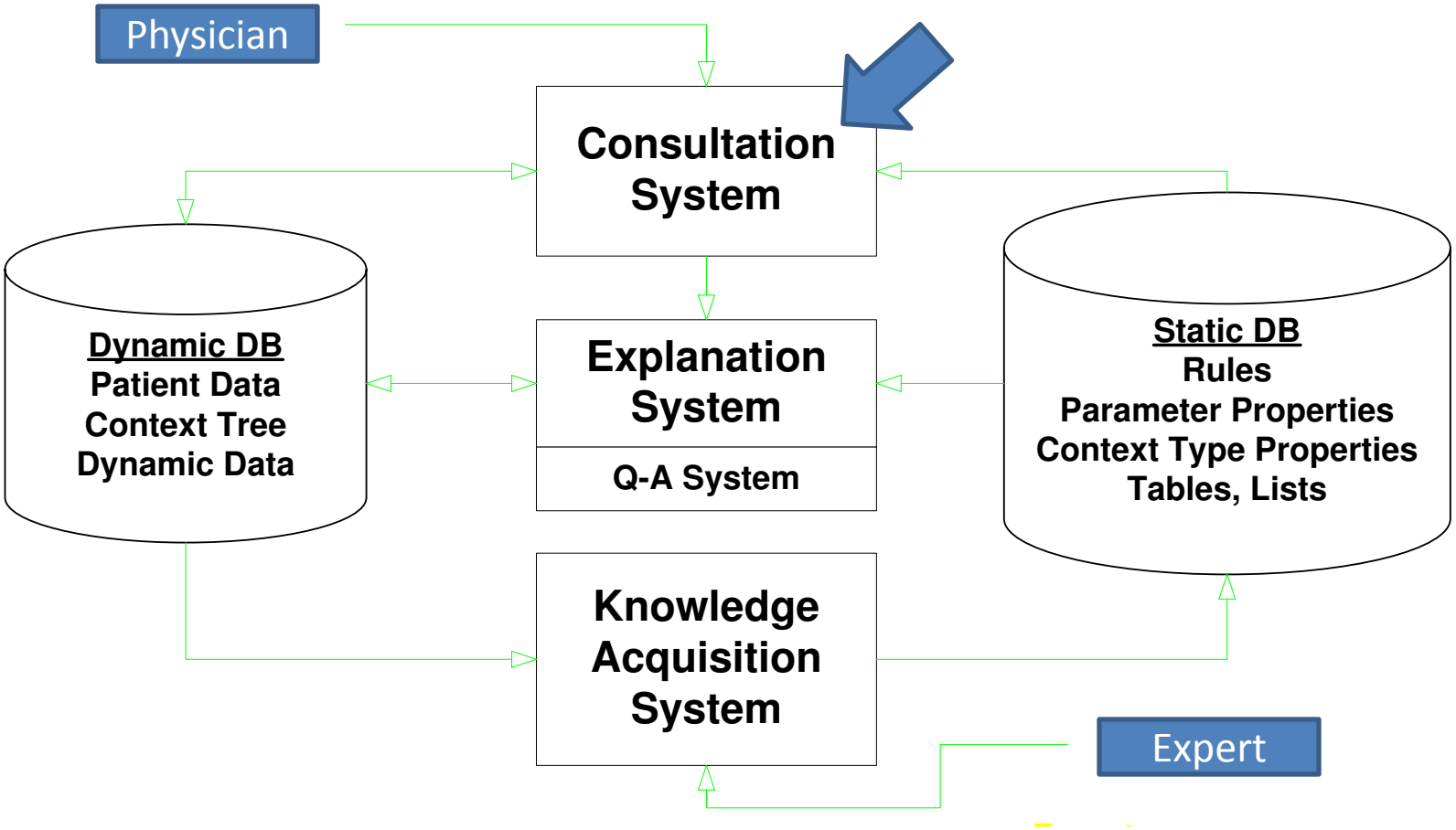
Tasks and Domain

- Advice for non-expert physicians with time considerations and **in**complete evidence on:
  - Bacterial infections of the blood
  - Expanded to meningitis and other ailments
- Disease DIAGNOSIS and Therapy SELECTION

System Goals

- Utility
- Flexibility
- Interactive Dialogue
- Fast and Easy

MYCIN Architecture

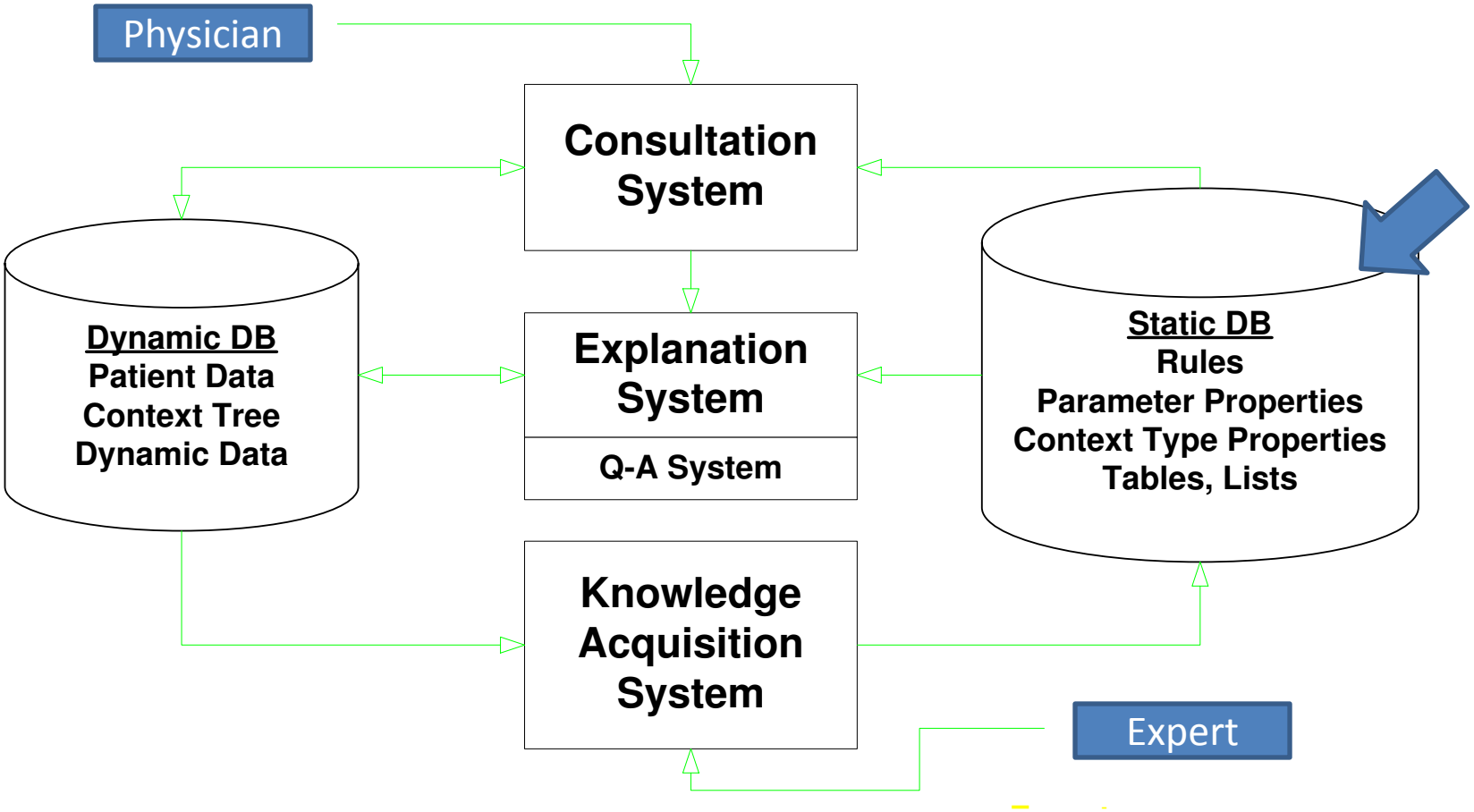


Consultation System

- Performs Diagnosis and Therapy Selection
- Control Structure reads Static DB (rules) and read/writes to Dynamic DB (patient, context)
- Linked to Explanations
- Terminal interface to Physician

Consultation: Control Structure

1. Determine if Patient has significant infection
2. Determine likely identity of significant organisms
3. Decide which drugs are potentially useful
4. Select best drug or coverage of drugs


Physician
Consultation System
Dynamic DB
Patient Data
Context Tree
Dynamic Data
Explanation System
Q-A System
Static DB
Rules
Parameter Properties
Context Type Properties
Tables, Lists
Knowledge Acquisition System
Expert

Static Database

- Rules
- Meta-Rules
- Templates
- Fed from Knowledge Acquisition System

Rules(Production)

- Represent Domain-specific Knowledge
- Above 450 rules in MYCIN
- Premise-Action (If-Then) Form:
- Each rule is completely modular, all relevant context is contained in the rule.

Meta-Rules

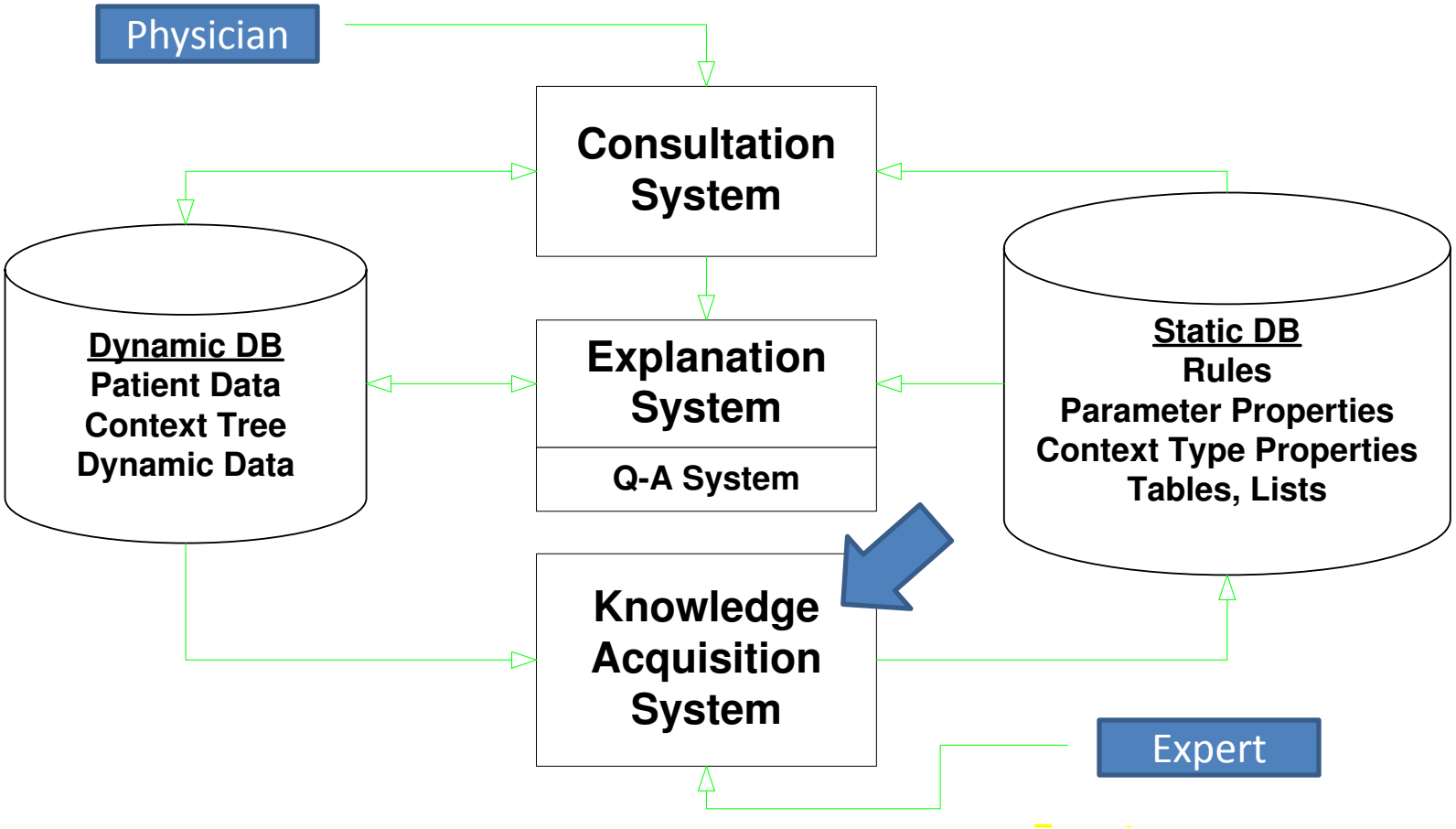
- Alternative to exhaustive invocation of all rules
- Strategy rules to suggest an approach for a given sub-goal
  - Ordering rules to try first, effectively pruning the search tree.

Meta-Rules (continued)

- High-order Meta-Rules (i.e. Meta-Rules for Meta-Rules)
  - Powerful, but used limitedly in practice
- Impact to the Explanation System:
  - (+) Encode Knowledge formerly in the Control Structure
  - (-) Sometimes create “murky” explanations

Templates

- The Production Rules are all based on Template structures
- This helps Knowledge-base expansion, because the system can “understand” its own representations
- Templates are updated by the system when a new rule is entered


Physician
Consultation System
Dynamic DB
Patient Data
Context Tree
Dynamic Data
Explanation System
Q-A System
Static DB
Rules
Parameter Properties
Context Type Properties
Tables, Lists
Knowledge Acquisition System
Expert

Knowledge Acquisition System

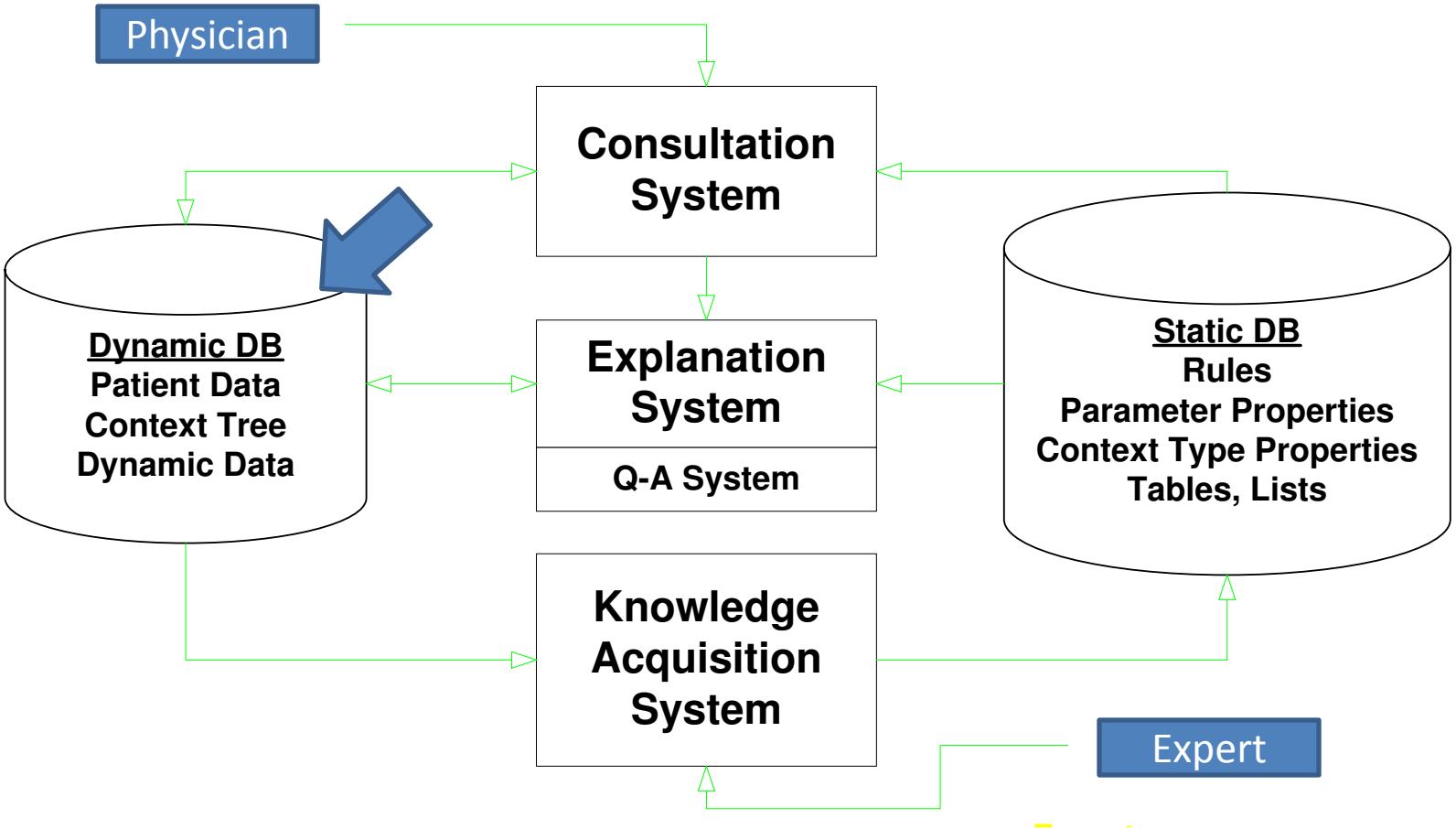
- Extends Static DB via Dialogue with Experts
- Dialogue Driven by System
- Requires minimal training for Experts

Knowledge Acquisition

- IF-THEN Symbolic logic was found to be easy for experts to learn, and required little training by the MYCIN team
- When faced with a rule, the expert must either except it or be forced to update it using the education process

Knowledge reasoning

- Inexact Reasoning with Certainty Factors (CF)
- (CF are not Probability!)
- Truth of a Hypothesis is measured by a sum of the CFs
  - Premises and Rules added together
  - Positive sum is confirming evidence
  - Negative sum is disconfirming evidence


Physician
Consultation System
Dynamic DB
Patient Data
Context Tree
Dynamic Data
Explanation System
Q-A System
Static DB
Rules
Parameter Properties
Context Type Properties
Tables, Lists
Knowledge Acquisition System
Expert

Dynamic Database

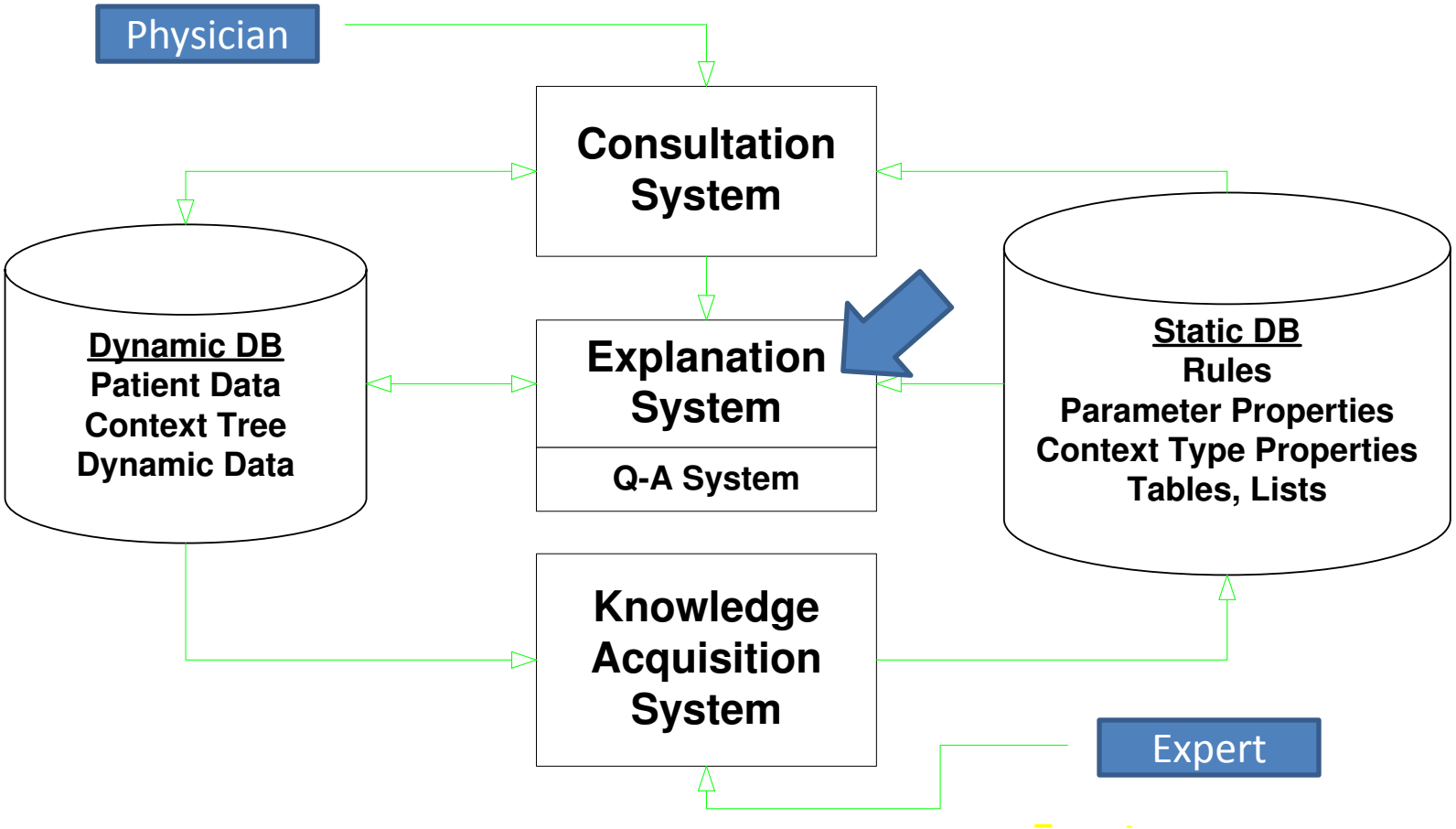
- Patient Data
- Laboratory Data
- Context Tree
- Built by Consultation System
- Used by Explanation System

Therapy Selection

- Plan-Generate-and-Test Process
- Therapy List Creation
  - Set of specific rules recommend treatments based on the probability (not CF)
  - Probabilities based on laboratory data
  - One therapy rule for every organism

Therapy Selection

- Final Selection based on:
  - Sensitivity
  - Contraindication Screening
  - Using the minimal number of drugs and maximizing the coverage of organisms
- Experts can ask for alternate treatments
  - Therapy selection is repeated with previously recommended drugs removed from the list


Physician
Consultation System
Dynamic DB
Patient Data
Context Tree
Dynamic Data
Explanation System
Q-A System
Static DB
Rules
Parameter Properties
Context Type Properties
Tables, Lists
Knowledge Acquisition System
Expert

Explanation System

- Provides reasoning why a conclusion has been made, or why a question is being asked
- Q-A Module
- Reasoning Status Checker

Preview Mechanism

- Interpreter reads rules before invoking them
- Avoids unnecessary deductive work if the sub-goal has already been tested/determined
- Ensures self-referencing sub-goals do not enter recursive infinite loops

Extension works

1. Bug is uncovered, usually by Explanation process
2. Add/Modify rules using *subset of English* by experts
3. Integrating new knowledge into KB
   - Found to be difficult in practice, requires detection of contradictions, and complex concepts become difficult to express

MYCIN -FACTS

- Never implemented for routine clinical use
- Shown to be competent by panels of experts, even in cases where experts themselves disagreed on conclusions
- Key Contributions:
  - Reuse of Production Rules (explanation, knowledge acquisition models)
  - Meta-Level Knowledge Use

References

- Jason Walonoski, Standsford University.
- Davis, Buchanan, Shortliffe. Production Rules as a Representation for a Knowledge-Based Consultation System. *Artificial Intelligence*, 1979.
- William van Melle. The Structure of the MYCIN System. *International Journal of Man-Machine Studies*, 1978.
- Shortliffe. Details of the Consultation System. *Computer-Based Medical Consultations: MYCIN*, 1976.
- Jadzia Cendrowska, Max Bramer. *Chapter 15?*
- *"Major Lessons From this Work"*
- William J. Clancey. *Details of the Revised Therapy Algorithm.* 1977