Comprehensive Introduction to Expert Systems for Professional Accountants

By Charles Hoffman, CPA (Charles.Hoffman@me.com)

August 21, 2016 (DRAFT)

In the *Conceptual Overview of an XBRL-based, Structured Digital Financial Report*[1] I state that digital financial report creation software will be an expert system much like CAD/CAM software which is used to create blueprints.

What does that mean? Does it mean that professional accountants will be replaced by machines? If the value that you add is simply clerical, amounting to cutting and pasting information or rekeying information, then quite possibly some of the tasks you perform could be replaced by automated machine-based processes. As explained in the document *Comprehensive Introduction to Knowledge Engineering for Professional Accountants*[2], there are certain tasks machines can perform and other tasks which machines will never likely be able to perform.

Thousands of expert system tools of all prices and qualities are commercially available today for performing different tasks. Because XBRL-based financial reports are structured data, computer software programs, such as expert systems, offer new capabilities specifically for financial reporting[3]. But what new capabilities will be offered? What work might be automated? This document is intended to help professional accountants understand what an expert system is, how they work, and what capabilities they will bring to help them perform work. Understating or overstating these capabilities are both not helpful.

The point is that setting the right expectations helps one understand what is actually practical and useful. Thinking that technology will have no impact on how you perform your work will be a complete disaster for your career as an accountant and/or for your business.

The global consultancy firm Gartner classifies XBRL as a transformational technology[4]. Gartner defines transformational as something that "*enables new ways of doing business across industries that will result in major shifts in industry dynamics*". Major shifts means lots of change and some winners and some losers.

This document helps professional accountants sort through all the information and misinformation that they are hearing related to XBRL-based structured digital financial reporting.

---

[1] *Conceptual Overview of an XBRL-based, Structured Digital Financial Report*, http://xbrlsite.azurewebsites.net/2016/Library/ConceptualOverviewOfDigitalFinancialReporting.pdf

[2] *Comprehensive Introduction to Knowledge Engineering for Professional Accountants*, http://xbrlsite.azurewebsites.net/2016/Library/ComprehensiveIntroductionToKnowledgeEngineeringForProfessionalAccountants.pdf

[3] YouTube.com, *How XBRL Works*, https://www.youtube.com/watch?v=nATJBPOiTxM

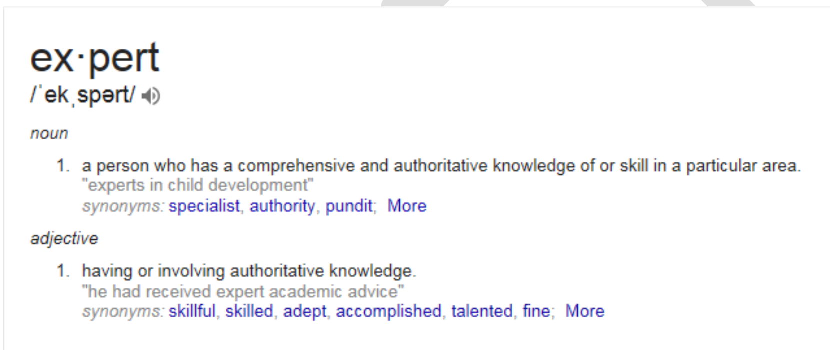
[4] Charles Hoffman and Liv Watson, *XBRL for Dummies*, page 145

1. Deconstructing the Notion of an Expert System

To understand expert systems one first needs to understand the notion of an expert. This section is dedicated to setting your perspective as to the notion of an expert and the fundamental notion of an expert system. The section provides specific definitions, deconstructing the pieces so that we can subsequently put the pieces back together.

1.1. Definition of an expert

Google defines expert[5] as "a person who has comprehensive and authoritative knowledge of skill in a particular area".


ex·pert
/ˈekˌspərt/

noun

1. a person who has a comprehensive and authoritative knowledge of or skill in a particular area.
"experts in child development"
synonyms: specialist, authority, pundit; More

adjective

1. having or involving authoritative knowledge.
"he had received expert academic advice"
synonyms: skillful, skilled, adept, accomplished, talented, fine; More

Professional accountants have a comprehensive and authoritative knowledge of and skills in the area of accounting and financial reporting. Accountants are experts at accounting and financial reporting.

1.2. Artificial intelligence

Artificial intelligence[6] is a branch of computer science. There are many good descriptions of artificial intelligence[7]. Here is one good definition:

> Artificial intelligence is the automation of activities that we associate with human thinking and activities such as decision making, problem solving, learning and so on.

Those trying to make artificial intelligence work over the past 40 or so years have had limited success[8]. But that is changing. People are putting the pieces together and the technology created from AI research are now available to experiment with. If expectations are not set too high, very useful functionality for limited, narrow problems can be

[5] Google search, *Expert definition*, retrieved August 21, 2016, https://www.google.com/search?q=expert+definition

[6] *Introduction to Artificial Intelligence Terminology*, http://xbrl.squarespace.com/journal/2016/7/21/introduction-to-artificial-intelligence-terminology.html

[7] AlanTuring.net, *What is Artificial Intelligence?*, http://www.alanturing.net/turing_archive/pages/reference%20articles/What%20is%20AI.html

[8] John F. Sowa, *Why Has AI Failed? And How Can it Succeed?*, http://www.jfsowa.com/pubs/micai.pdf

successfully created. Both under estimating or over estimating the capabilities the computer software will be able to achieve can have catastrophic consequences.

One good example of using artificial intelligence is driverless cars. Many people get confused as to what is truly achievable and practical when it comes to driverless cars. Driverless cars on the streets of Singapore today[9]. While still in prototype mode now to work out details, by 2018 these taxis are anticipated to be commercially available. Uber is testing autonomous cars in Pittsburg[10].

One of the best ways to understand the capabilities of artificial intelligence is to try it out. Go test drive a Tesla which has driver-assist features.

1.3. Expert systems

Expert systems[11] is a branch of artificial intelligence. The following is a definition of an expert system:

> Expert systems are computer programs that are built to mimic human behavior and knowledge. Expert systems are for reconstructing the expertise and reasoning capabilities of qualified experts within some limited, narrow domain of knowledge in machine-readable form. A model of the expertise of a domain of knowledge of the best practitioners or experts is formally represented in machine-readable form and the expert system reaches conclusions or takes actions based on that information when trying to solve some problem. The computer program performs tasks that would otherwise be performed by a human expert.

Key to understating the capabilities of expert systems is an understanding the fundamental capabilities of computers. To understand the capabilities of computers, please be sure to read the document *Comprehensive Introduction to Knowledge Engineering for Professional Accountants*[12].

1.4. Differentiating the mechanical aspects of a financial report and judgment

Computers are dumb beasts. Computers only follow instructions. Computers will never possess judgment. They may try and mimic judgment and could be successful to some degree.

A financial report itself is mechanical. While what must go into the financial report requires the judgement of a professional accountant, the mechanics of a financial report are objective. Balance sheets always balance[13]. Roll ups always roll up. Roll forwards always roll forward. Everything should always "tick and tie", "cross cast and foot". Accountants

---

[9] Reuters, *First driverless taxis hit the streets of Singapore*, http://www.reuters.com/article/us-autos-selfdriving-singapore-idUSKCN1100ZG

[10] The Atlantic, *Anybody Can Test a Driverless Car in Pennsylvania*, http://www.theatlantic.com/technology/archive/2016/09/anybody-can-test-a-self-driving-car-in-pennsylvania/499667/

[11] *Understanding the Components of an Expert System*, http://xbrl.squarespace.com/journal/2016/5/24/understanding-the-components-of-an-expert-system.html

[12] *Comprehensive Introduction to Knowledge Engineering for Professional Accountants*, http://xbrlsite.azurewebsites.net/2016/Library/ComprehensiveIntroductionToKnowledgeEngineeringForProfessionalAccountants.pdf

[13] Wikipedia, *Accounting Equation*, https://en.wikipedia.org/wiki/Accounting_equation

excel at performing these detailed tasks. But computers are completely capable of managing the mechanical details of a financial report. That frees professional accountants from having to worry about those mechanical details and focus on where they add the most value which is the aspects of creating a financial report that require judgment.

Not making the proper distinction between the mechanical aspects and the aspects that require judgement will cause someone to either overestimate the work a computer can perform or under estimate that work.

1.5. Business rules

Key to employing artificial intelligence and therefore making an expert system work is machine-readable business rules[14] of a domain of knowledge.

Business rules guide, control, suggest, or influence behavior. Business rules cause things to happen, prevent things from happing, or suggest that it might be a good idea if something did or did not happen. Business rules help shape judgment, help make decisions, help evaluate, help shape behavior, and help reach conclusions.

Business rules arise from the best practices of knowledgeable business professionals. A business rule is a rule that describes, defines, guides, controls, suggests, influences or otherwise constrains some aspect of knowledge or structure within some problem domain.

1.6. Objects as rules

Besides business rules, expert systems can use another form of knowledge representation. Objects, rather than rules, can play a leading role in knowledge representation. Although, object-based representations of knowledge can be regarded as rules organized in a different manner.

1.7. Human-readable and machine-readable business rules

Business professionals interact with business rules every day by may not even realize it. Most business rules are in human readable form. But business rules can be represented in both human-readable form and machine-readable form. With the move to digital, more and more business rules are being represented in both human readable form and more importantly machine-readable form. Formalized, standardized machine-readable business rules can help automate processes which have been manual in the past.

1.8. Intelligent software agents

As stated, artificial intelligence is the automation of activities that we associate with human thinking and activities such as decision making, problem solving, learning and so on. How do you create that sort of automation? The answer is using a style of computer programming called intelligent software agents.

An intelligent software agent[15] is software that assists people and acts on their behalf. Intelligent agents add value by allowing people to:

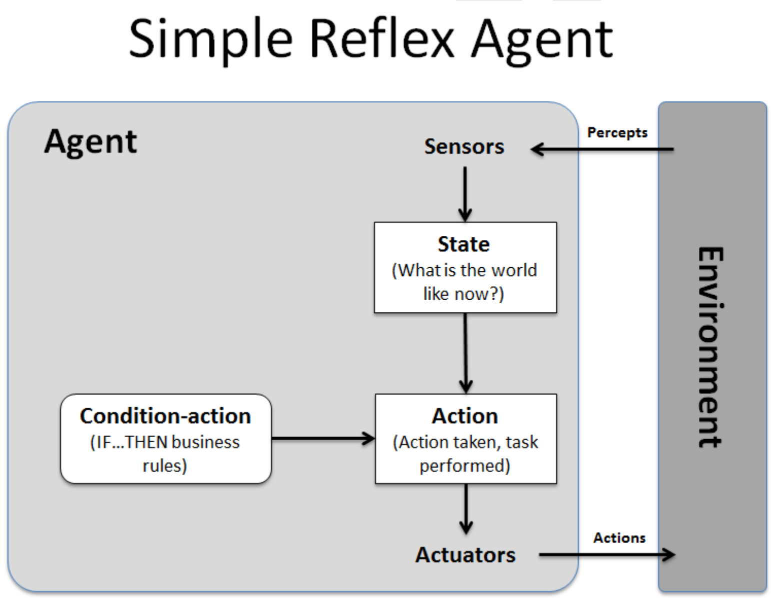
---

[14] *Comprehensive Introduction to Business Rules for Professional Accountants*, http://xbrl.squarespace.com/journal/2016/8/3/comprehensive-introduction-to-business-rules-for-professiona.html

[15] *Comprehensive Introduction to Intelligent Software Agents for Professional Accountants*, http://xbrlsite.azurewebsites.net/2016/Library/ComprehensiveIntroductionToIntelligentSoftwareAgentsForProfessionalAccountants.pdf

- delegate work that they could have done to the agent software.
- perform repetitive tasks,
- remember things you forgot,
- intelligently find, filter and summarize complex information,
- customize information to your preferences,
- learn from you and even make recommendations to you.

An intelligent agent is computer software capable of **sensing** the **state** of its **environment** and **acting** upon it based on a set of specified **rules**. An intelligent agent performs specific tasks on behalf of another. In the case of software, an agent is a software program. There are many different types of intelligent software agents[16].



The document *Comprehensive Introduction to Intelligent Software Agents for Professional Accountants*[17] goes into significantly more detail on the topic of intelligent software agents.

[16] Wikipedia*, Intelligent Agent,* Retrieved July 24, 2016; https://en.wikipedia.org/wiki/Intelligent_agent

[17] *Comprehensive Introduction to Intelligent Software Agents for Professional Accountants*, http://xbrlsite.azurewebsites.net/2016/Library/ComprehensiveIntroductionToIntelligentSoftwareAgentsForProfessionalAccountants.pdf

2. Digging deeper into expert systems

The previous section sets a foundation for understanding expert systems. In this section we go into additional important details that help round out your understanding of expert systems.

2.1. Types of expert systems

Frank Puppe explains in his book *Systematic Introduction to Expert Systems*[18] that there are three general categories of expert systems:

- **Classification or diagnosis type**: helps users of the system select from a set of given alternatives.
- **Construction type**: helps users of the system assemble something from given primitive components.
- **Simulation type**: helps users of the system understand how some model reacts to certain inputs.

The assembly of a financial report can be assisted by a construction-type expert system. Helping professional accountants understand what goes into that financial report can be assisted by a classification-type expert system. Creating forecasts and projections of future financial reports can be assisted by simulation-type expert systems.

2.2. Components of an expert system

A software based expert system has four primary components:

- **Database of facts**: A database of facts is a set of observations about some current situation or instance. The database of facts is "flexible" in that they apply to the current situation. The database of facts is machine-readable. An XBRL instance is a database of facts.
- **Knowledge base (rules)**: A knowledge base is a set of universally applicable rules created based on experience and knowledge of the practices of the best domain experts generally articulated in the form of IF…THEN statements or a form that can be converted to IF...THEN form. A knowledge base is "fixed" in that its rules are universally relevant to all situations covered by the knowledge base. Not all rules are relevant to every situation. But where a rule is applicable it is universally applicable. All knowledge base information is machine-readable. An XBRL taxonomy is a knowledge base. Business rules are declarative in order to maximize use of the rules and make it easy to maintain business rules.
- **Rules processor/inference engine**: A rules processor/inference engine takes existing information in the knowledge base and the database of facts and uses that information to reach conclusions or take actions. The inference engine derives new facts from existing facts using the rules of logic. The rules processor/inference engine is the machine that processes the information. An XBRL Formula processor is a rules processor and, if build correctly, can perform logical inference.

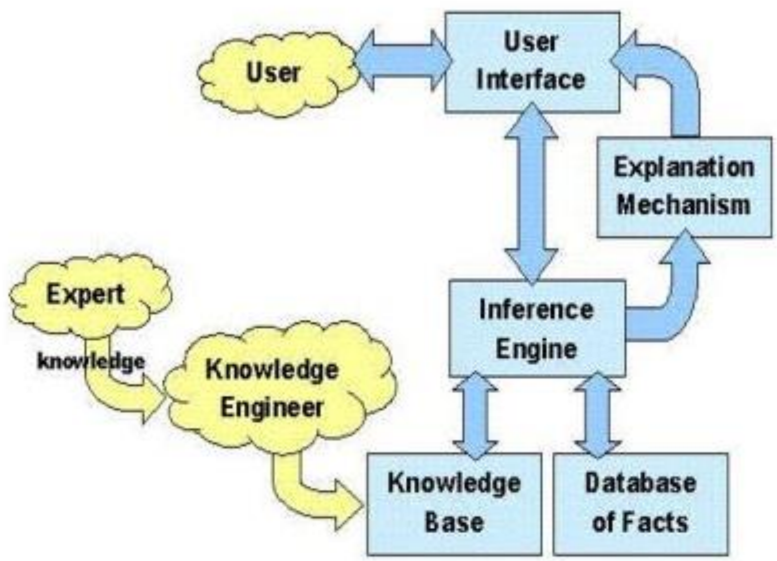
[18] Frank Puppe, *Systematic Introduction to Expert Systems, Knowledge Representations and Problem-Solving Methods*, page 11 (Note that you can read Parts I and II on Google Books here, https://books.google.com/books?id=_kKqCAAAQBAJ)

- **Explanation mechanism**: The explanation mechanism explains and justifies how a conclusion or conclusions are reached. It walks you through which facts and which rules were used to reach a conclusion. The explanation mechanism is the results of processing the information using the rules processor/inference engine and justifies why the conclusion was reached. The explanation mechanism provides both provenance and transparency to the user of the expert system.

These four pieces are exposed to the users of the expert system within software applications. One philosophical difference expert systems have from typical software systems which tend to be procedural is that expert systems separate domain logic and program control logic. This enables domain logic to be managed/maintained by domain experts and the domain logic is reusable by other programs.

Expert systems provide *transparency* to their users and can explain the solutions they provide by quoting the knowledge used to reach that solution. Single pieces of knowledge can be easily added, changed, or removed; providing *flexibility*. The users of expert systems should require no knowledge of programming languages by either the creator of the expert system or user of the expert system, providing *ease of use*. The boundaries of the probably solving capabilities should be clear so users of the system understand what the system provides and what needs to be provided using alternative processes.

The following graphic[19] shows how the components of an expert system interact:



2.3. Problem solving method

The objective of an expert system is to solve some problem. Conventional software applications work using algorithms (software programs) and data. Expert systems separate the "algorithms" into two parts; knowledge and the problem solving method. Another way

19 https://imscdrmba.wordpress.com/206-unit-iii/

to say this is that expert systems separate software programs into domain logic and control logic.

- Forward chaining
- Backward chaining
- Combination of forward and backward chaining:

Please see the document *Comprehensive Introduction to Business Rules for Professional Accountants* for a more complete introduction to forward and backward chaining.

2.4. Problem solving logic

One of the most complicated things to understand about expert systems is the problem solving logic used by the expert system to solve problems. The problem solving logic drives the extent of rules that can be created because the nature of the logic constrains what rules can be expressed.

A logic can be defined as any precise notation for expressing statements that can be judged to be either true or false[20]. Almost any declarative notation, graphic or linear, could be treated as a version of logic: just specify it precisely. A logic needs to define foundational terminology such as: *there exists*, *every*, *and*, *or*, *if and only if*, *if-then*, *not*, *true*, and *false*[21]. Really, it is that simple. What makes this complicated are all the different notations that are used to refer to those terms. Using natural language make logic more understandable to business professionals.

Determining the problem solving logic is a balancing act. The objective is to have the maximum amount of expressiveness but the minimum chance that software will break. The logic needs to be safely implementable by software.

For the past 30 or so years, many different technical solutions have been created to solve different business problems. Few of these technical solutions achieved the appropriate balance or equilibrium and tended to not maximize expressiveness or not be safely implementable in the form of software applications. Benjamin Grosof, Michael Kifer, and Mike Dean summarize this history in their presentation, *Semantic Web Rules: Fundamentals, Applications, and Standards*[22].

Please see the document *Comprehensive Introduction to Knowledge Engineering for Professional Accountants*[23] for a complete discussion of this topic.

The bottom line is that the best balance between expressiveness and safe implementation has been achieved by the ISO/IEC global standard Common Logic. **Common Logic**[24] is a framework for a family of logic languages, based on first-order logic, intended to facilitate the exchange and transmission of knowledge in computer-based systems. That safely expressive sweet spot is also used by the OMG standard **Semantics of Business**

---

[20] John F. Sowa, *Fads and Fallacies about Logic*, page 2, http://www.jfsowa.com/pubs/fflogic.pdf

[21] Wikipedia, *List of Logical Symbols*, https://en.wikipedia.org/wiki/List_of_logic_symbols

[22] Benjamin Grosof, Michael Kifer, and Mike Dean, *Semantic Web Rules: Fundamentals, Applications, and Standards*; http://coherentknowledge.com/wp-content/uploads/2013/05/talk-prelim-aaai13-rules-tutorial.pdf

[23] *Comprehensive Introduction to Knowledge Engineering for Professional Accountants*, http://xbrlsite.azurewebsites.net/2016/Library/ComprehensiveIntroductionToKnowledgeEngineeringForProfessionalAccountants.pdf

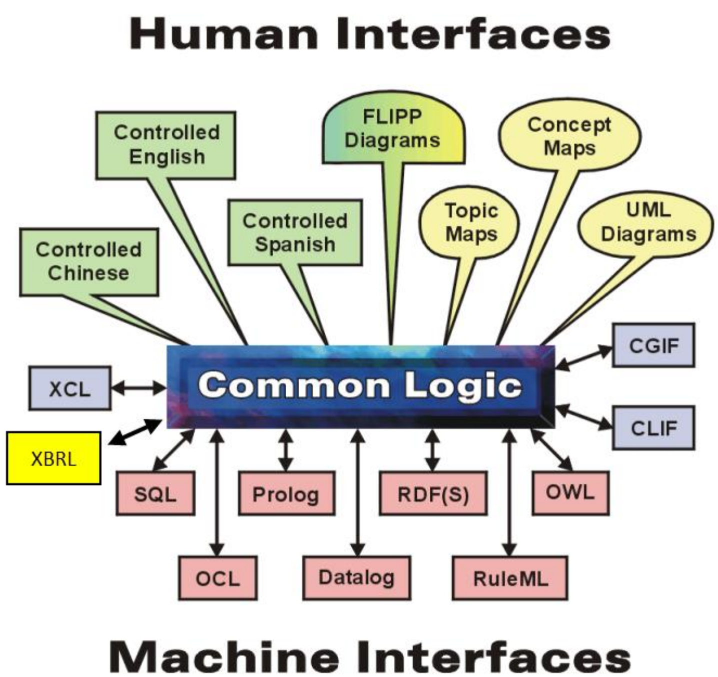
[24] *Understanding Common Logic*, http://xbrl.squarespace.com/journal/2016/6/23/understanding-common-logic.html

Vocabulary and Business Rules[25] which was consciously designed to be logically equivalent to ISO/IEC Common Logic.

Rulelog[26] is a logic for expressing knowledge that is consciously engineered to be consistent with ISO/IEC Common Logic and OMG Semantics of Business Vocabulary and Business Rules. Rulelog is a dialect of W3C's RIF[27]. RuleML[28] is a syntax for implementing rules. Other standard and proprietary syntaxes exist for implementing rules.

The most important thing to realize is that there is a good, safe target in terms of an expressive logic that is also safely implementable in software so catastrophic failures are avoided. Another very good thing is that business professionals don't need to understand the underlying technical details of these logic standards, nor will they every have to deal with them. Higher level languages that follow the foundations set by Common Logic, Semantics of Business Vocabulary and Business Rules, and Rulelog.

The following graphic shows the role Common Logic[29] plays, establishing a family of logical dialects shared between different software syntax implementations: (note that this graphic was modified, XBRL was added)



The language in which a problem is stated has no effect on complexity. Reducing the expressive power of a logic does not solve any problems faster; its only effect is to make

25 OMG, *Semantics of Business Vocabulary and Business Rules (SBVR)*, section 2.5 Conformance of an SBVR Processor, page 7, http://www.omg.org/spec/SBVR/1.0/
26 *Rulelog*, http://ruleml.org/rif/rulelog/spec/Rulelog.html
27 W3C, *RIF Overview (Second Addition)*, http://www.w3.org/TR/rif-overview/
28 *RuleML*, http://wiki.ruleml.org/index.php/RuleML_Home
29 John F. Sowa, *Common Logic: A Framework for a Family Of Logic-Based Languages*, page 5, http://www.jfsowa.com/ikl/SowaST08.pdf

some problems impossible to state[30].

2.5. Benefits of an expert system

Benefits from the use of expert systems include:

- **Automation**: elimination of routine, boring, repetitive, mundane, mechanical tasks that can be automate
- **Consistency**: computers are good at performing repetitive, mechanical tasks whereas humans are not; computers do not make mistakes and are good at repeating exactly the same thing each time
- **Diligence and tenacity**: computers excel at paying attention to detail; they never get bored or overwhelmed and they are always available and will keep doing their job until the task is complete with the same attention to detail
- **Reduced down-time**: computer based expert systems are tireless and do not get distracted
- **Availability**: computer based expert systems are always available simultaneously in multiple places at one time; you get quick response times and can replace absent or scarce experts
- **Training**: the best practices of the best practitioners can be available to those that are new to and learning about a domain of knowledge
- **Longevity and persistence**: computer based expert systems do not change jobs or retire so knowledge gathered by an organization can remain within that organization
- **Productivity**: computer based expert systems are cheaper that hiring experts and costs can be reduced a the same time that quality increases resulting in increased productivity
- **Multiple opinions**: Systems can integrate the view of multiple experts within a system and choose between the preferred view of multiple expert opinions in the same system
- **Objectivity**: computers apply the same inductive and deductive logic consistently; emotion and personal preferences can be eliminated where they should be eliminated

Financial report creation software of the future will be an expert system which operates similar to how CAD/CAM software for creating blueprints.

2.6. Contrasting universal tools and domain specific tools

In his book Systematic Introduction to Expert Systems[31] (an excellent book which I highly recommend), Frank Puppe provides the graphic below. The graphic basically points out that universal, general tools are less restrictive but cost more to create domain-specific tools. In

---

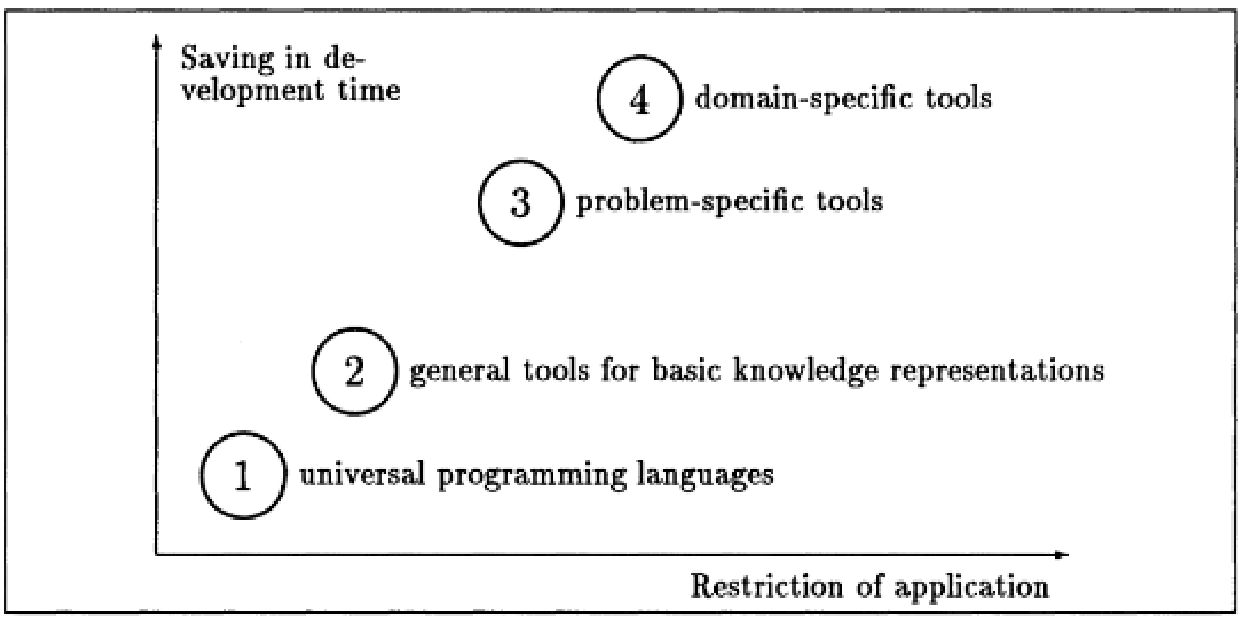
[30] John F. Sowa, *Fads and Fallacies about Logic*, page 5, http://www.jfsowa.com/pubs/fflogic.pdf

[31] Frank Puppe, *Systematic Introduction to Expert Systems*, page 11, https://books.google.com/books?id=_kKqCAAAQBAJ&printsec=frontcover&source=gbs_ge_summary_r&cad=0#v=onepage&q&f=false

addition to universal, general tools being more costly to create and more difficult to create; domain specific tools are easier to create and much, much easier for business professionals to use because of the restrictions.

So, a "restriction" is not a flaw. The restriction is what makes the tool easier to use and cost less and make easier to develop. You don't need the universe of all possible options; you only need to create what that specific domain needs. As long as you get these restrictions correct, they really are not "restrictions" of the domain, they are the "boundaries" of the domain. You don't need them. Technical people don't typically understand these domain boundaries. Many times to play it safe technical people add flexibility in order to make certain that business domain user needs are being met. But this flexibility comes at a cost. Additional costs are incurred to create the flexibility and software is harder to use because business professionals need to figure out which option they should use.

Business domain people do understand the boundaries if they think about them. Many business professionals cannot properly articulate the appropriate boundaries or restrictions. This communications problem tends to lead to software that costs more to create than is necessary and harder to use than necessary.



This is not an either-or choice. Sometimes universal tools are very appropriate. Other times domain-specific tools are appropriate. Being conscious of these dynamics will lead to the right software being created and the appropriate level of usability. Universal tools are not a panacea. Unconsciously constricting a domain-specific tool when it would have been better to create a more universally usable tool also can be a mistake one makes.

2.7. Examples of expert systems in other domains

Expert systems are available commercially at different price levels and with different capabilities. The following is a brief list of expert systems to give you an idea of the potential capabilities of expert systems:

- Chess game (for example, IBM’s Deep Blue beat the grand master at the time)
- Medical triage and diagnosis
- Robotic surgery
- Aircraft accident investigation
- Patriot missile guidance system
- Numerically controlled manufacturing machine

3. Using CLIPS to Understand Expert Systems

CLIPS[32] is a tool for building expert systems originally developed by NASA. Since it was first released in 1986 it has undergone significant enhancements and was put into the public domain by NASA in about 2002. CLIPS continues to be maintained as public domain software.

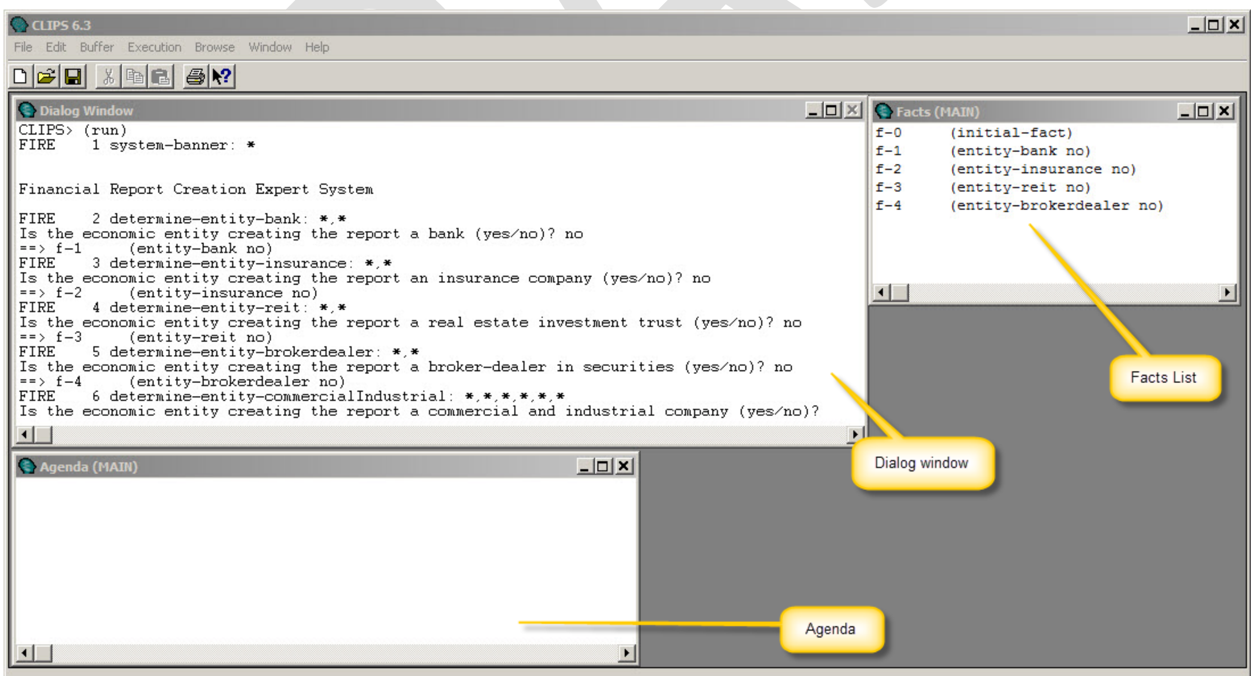
While CLIPS is not a tool the average professional accountant will use, CLIPS does offer a way to understand how expert systems work.

Recognize that CLIPS is a system for building any expert system that you want. As such, you have to understand how to put the pieces of an expert system together. Professional accountants will not have to do this for expert systems which create financial reports. Just as architects don't have to build computer aided design or computer aided manufacturing expert systems from scratch, neither will accountants. Software developers will create expert systems that professional accountants will use.

However, some professional accountants will build expert systems. Tools such as CLIPS allow someone who understand logic programming to create expert systems for other domains. For example, a small business might have a task they perform manually which they might want replaced by an automated expert system. Professional accountants will help small businesses create such micro expert systems. This could be a niche service offered by professional accountants.

3.1. Overview of CLIPS

Consider the software application interface of CLIPS below:



32 *Using CLIPS to Understand Expert Systems and Logic Programming*, http://xbrl.squarespace.com/journal/2016/9/15/using-clips-to-understand-expert-systems-and-logic-programmi.html

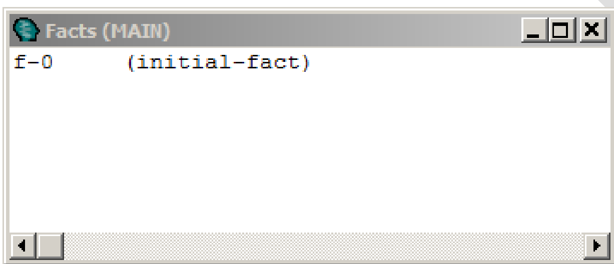
The interface is showing three parts:

- **Fact list**: In the upper right hand corner you see a facts list or "database of facts".
- **Agenda**: In the lower left side you see an agenda window. Currently the agenda window is empty which means the expert system has no more tasks to complete.
- **Dialog window**: In the upper left, the largest window is the dialog window. The dialog window is where the user interacts with the expert system software.

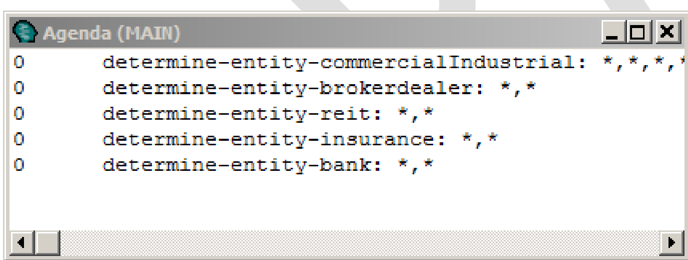
Remember that CLIPS is a universal programming language that is designed to enable anyone to build any sort of expert system. That is why the interface is general. So please don't get distracted by the nature of the interface. Focus on the logic of how an expert system works.

When the expert system starts, the facts list and the agenda look as follows:

Facts list:



Agenda:



The fact list is empty (except for a default fact) and then the agenda has five activations. Basically, no facts are known (in the case of this system) and there are five items on the agenda to determine the accounting activity of the economic entity creating the financial report.

This is the terminology used by the CLIPS system for building expert systems which will give you an idea of how you interact with CLIPS.

- **Goal**: The system will cease execution when no *activations* are on the *agenda*.
- **Strategy**: High-level plan for achieving a goal.
- **Fact**: A fact is the same as an XBRL definition of a fact.

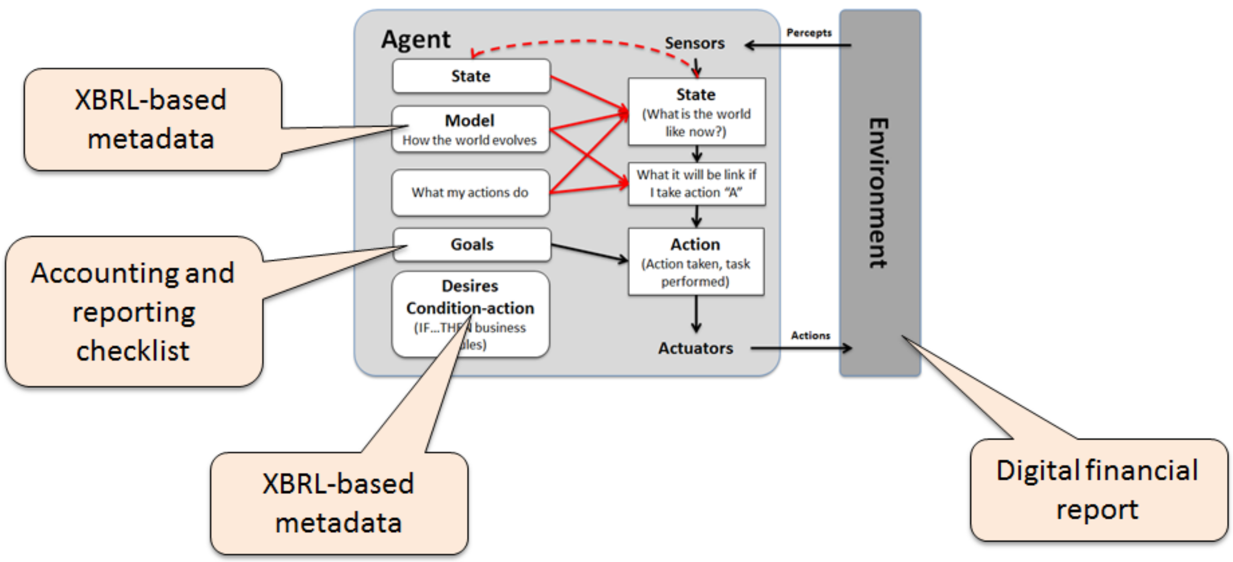
- **Fact list**: A fact list is the set of facts the system is currently working with to arrive at some goal.
- **Rule**: A rule is a relation between facts or a relation between a fact and the characteristics or traits of a fact.
- **Assertion**: Assert a new fact.
- **Retraction**: Retract an existing fact.
- **Activation**: An activation is a rule that is active because it matches the forward chaining strategy.
- **Agenda**: The agenda is a collection of activations which are those rules which match pattern entities. Zero or more activations may be on the agenda.
- **Salience**: When multiple activations are on the agenda, the system automatically determines which activation is appropriate to fire. The system orders the activations on the agenda in terms of increasing priority or salience.
- **Depth strategy**: In the depth strategy, new activations are placed on the agenda after activations with higher salience, but before activations with equal or lower salience. All this simply means is that the agenda is ordered from highest to lowest salience.
- **Conflict resolution**: The inference engine sorts the activations according to their salience. This sorting process is called conflict resolution because it eliminates the conflict of deciding which rule should fire next.
- **Refraction**: Refraction is the management of when rules fire so trivial loops are avoided. Without refraction, expert systems always would be caught in trivial loops. That is, as soon as a rule fired, it would keep firing on that same fact over and over again. In the real world, the stimulus that caused the firing eventually would disappear.
- **Clear**: Removes all facts and rules from working memory, basically resetting the system.

4. Opportunities for Using Expert Systems in Financial Reporting

What type of expert systems could be useful in the process of creating a financial report? Why would you even want to create an agent for financial reporting[33]?

4.1. Starting simple; example of one type of expert system

The following is the architecture of an agent that could be helpful in the process of creating a financial report[34]. I believe that such an expert system would be classified as a global standard rational, deliberative, non-learning, goal-based intelligent software agent.



4.2. Automating accounting and reporting checklists

Most accountants are familiar with the disclosure checklist. They use that human readable checklist as a memory jogger in the process of creating financial reports. What if you made that checklist also readable by machines and what if financial reports were structured? Automating the disclosure checklist will be one of the first uses of intelligent agent software[35]. This will not be a batch process that you run when a financial report is complete; rather it will be an expert system intelligent software agent watching over you as you create the financial report.

---

[33] *Benefits Offered by an Expert System,* Retrieved July 24, 2016; http://xbrl.squarespace.com/journal/2016/5/30/understanding-the-benefits-offered-by-expert-systems.html

[34] *Automating Accounting and Reporting Checklists,* Retrieved July 24, 2016; http://xbrl.squarespace.com/journal/2016/5/5/automating-accounting-and-reporting-checklists.html

[35] *Automating Accounting and Reporting Checklists,* Retrieved July 24, 2016; http://xbrl.squarespace.com/journal/2016/5/5/automating-accounting-and-reporting-checklists.html