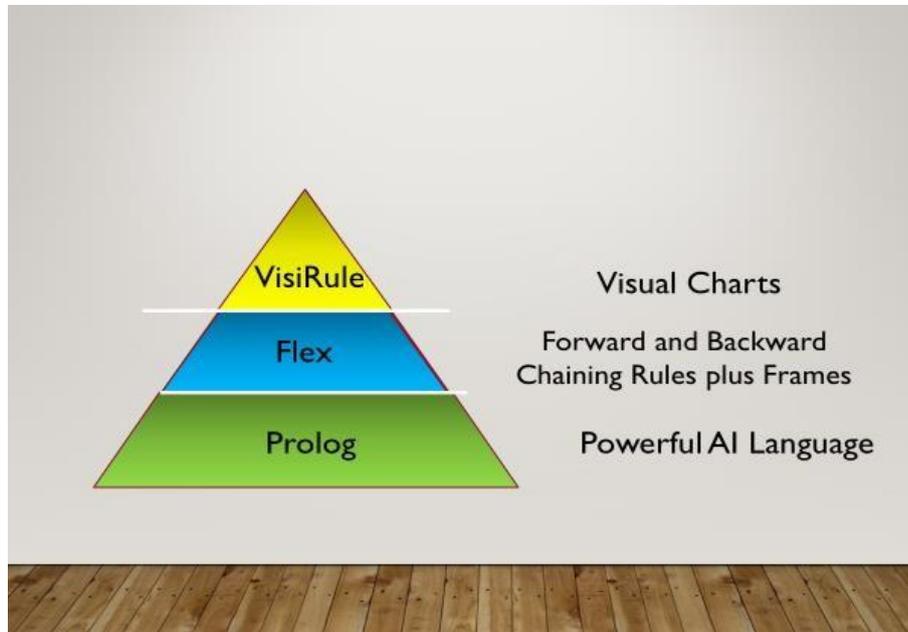


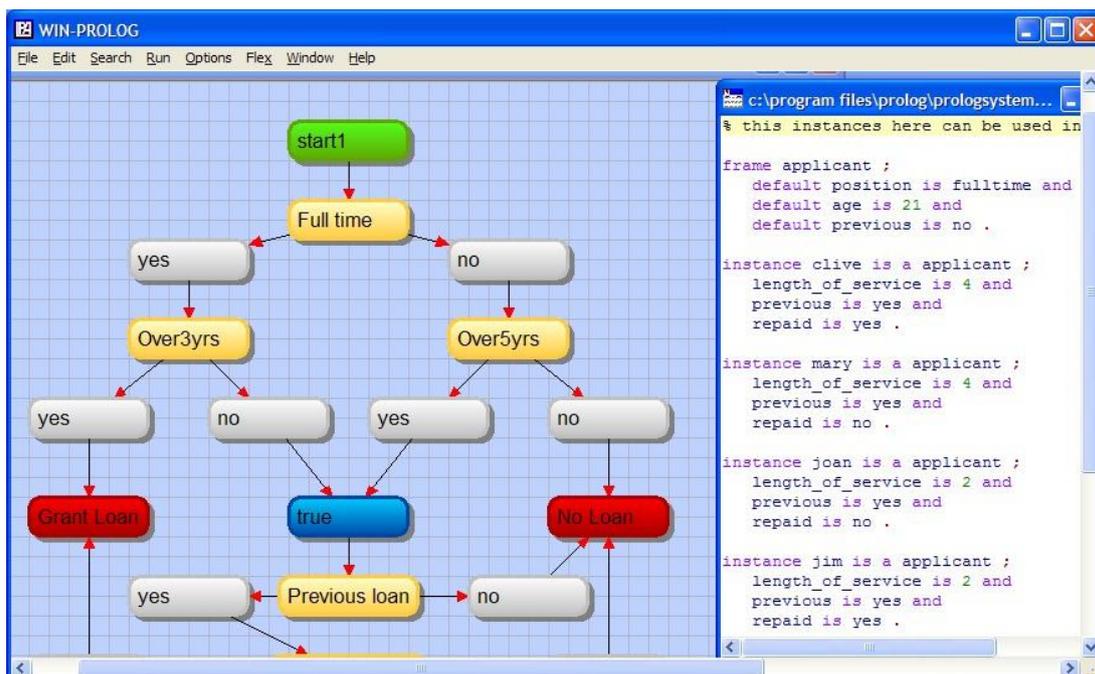
# LPA Software Tools

Summer 2020

This document presents LPA's suite of high-performance application development tools. This suite covers a wide range of Artificial Intelligence idioms such as logic programming, expert systems, case-based reasoning, data mining, fuzzy logic and intelligent agents - ideal for building smart, innovative solutions.



VisiRule is a No-Code, Low-Code graphical tool which enables authors to develop and deliver rule-based solutions simply by drawing their decision logic as a flow chart. VisiRule automatically generates executable code in the form of Flex. Flex is a hybrid expert system rules-based tool which has access to the underlying Prolog engine and also to other languages. This provides for a powerful multi-layered solution.

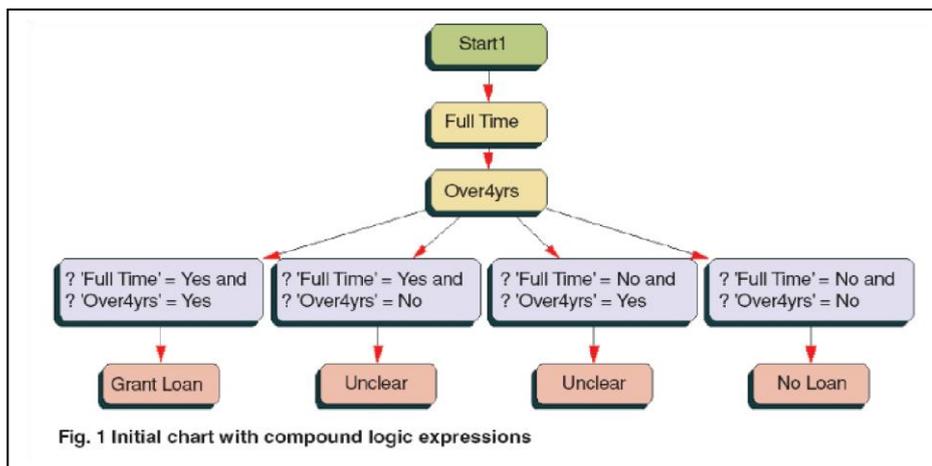


## VisiRule

VisiRule is a No-Code graphical tool for designing, developing and delivering business rule and decision support applications. Expert knowledge can be modeled using a visual chart and shared with colleagues. LPA provides a dedicated Windows desktop authoring tool to rapidly draw and test the correctness of charts.

- Executable - charts can be directly executed
- Explorable - charts and models can be explored and edited graphically
- Extendable - charts can be extended using a rules-based language
- Embeddable - the generated rules can be embedded within larger processes
- Exportable - charts can be exported to, and used by, other programs via XML/Json

*“VisiRule is a tool that allows experts to build decision models using a graphical paradigm, like MindMap, one that can be annotated using code and/or logic, executed and exported to other programs and processes.”*



## Logical Rules

Rule-based platforms take the business logic out of application code and place it in a dedicated managed environment. Taking the job of encapsulating executable rules away from programmers and returning them to the business user is a key goal. VisiRule is a graphical tool for non-programmers seated in logic-based programming – logic helps ensure a high-level of semantic integrity and provides the basis for subsequent auditing and verification in terms of completeness and consistency checks.

**Table 1. Simple set of rules.**

If Full time = yes	and Over4yrs = yes	then answer = Grant Loan
If Full time = yes	and Over4yrs = no	then answer = Unclear
If Full time = no	and Over4yrs = yes	then answer = Unclear
If Full time = no	and Over4yrs = no	then answer = No Loan

# VisiRule

## *Who is VisiRule for?*

Lawyers, advisors and business professionals who want to **design** and **deliver** expert systems  
Businesses who want to **capture** and **automate** decision-making processes  
Knowledge Engineers and Data Scientists who want to **build** intelligent solutions

## *Why Automate?*

**Availability:** automated decision systems are always available anytime, anywhere  
**Speed** and **Validation:** Answers are computed quickly and consistently along with audit trails  
**Integration** and **Re-use:** Answers and inputs can be stored for future processing

## *What is VisiRule used for?*

**Decision Support** and **Advisory** systems for Legal Advice, Finance and Compliancy  
Medical and machinery **diagnostic** systems  
Configuration, Planning, **Scheduling** and Resource Allocation

## *What does VisiRule provide?*

Easy-to-use, intuitive and highly interactive decision modeling tool  
Ability to test charts both interactively and using corporate data  
Automatic code generation for building both self-contained and embeddable applications  
Automated decision making which is both **explainable** and transparent

## *Aims of VisiRule*

Make AI technology available to non-programmers using a trusted and familiar metaphor Build question and answer-based systems which can be rapidly tested and deployed

## *Benefits of VisiRule*

Increased **productivity** through dedicated authoring environment  
Empower business users to **directly** model their own knowledge without programmers **Reduce** risk via consistent, compliant and automated solutions

## **Productivity Improvements due to Graphical Programming**

By helping automate and simplify the design and delivery of complex decision support systems, VisiRule saves both time and money. Not only can VisiRule be used by people with no programming skills, but it can also transform the experience of building decision support and business rules systems for practiced systems builders. VisiRule enhances productivity by reducing the production time while increasing the final quality of the solutions. Because, VisiRule is grounded in logic, charts can be analysed and verified in an intelligent manner. This results in fewer bugs and gives a higher degree of trust and confidence in the final outputs.

## **Shareable Know-How**

VisiRule captures the expertise and 'know-how' of experts in a way which is directly shareable with colleagues and collaborators. VisiRule charts can be exported using industry standard formats - no need to buy or install any additional software. All the relevant logic and business rules is visible in a simple and coherent manner. This helps people build a consensus and dramatically improves group productivity.

## Intelligent FlowChart Creation

VisiRule is an intelligent tool which guides the design process by monitoring what can and cannot be done based on the implied structure of the emerging program. This helps reduce the potential for drawing invalid or meaningless links and improves productivity by detecting errors early within the design process. VisiRule automatically constructs menus for questions using the expressions in the chart. This reduces the potential of simple but common errors caused by typographical errors and helps save time all-round.

## Immediate Testing and Delivery

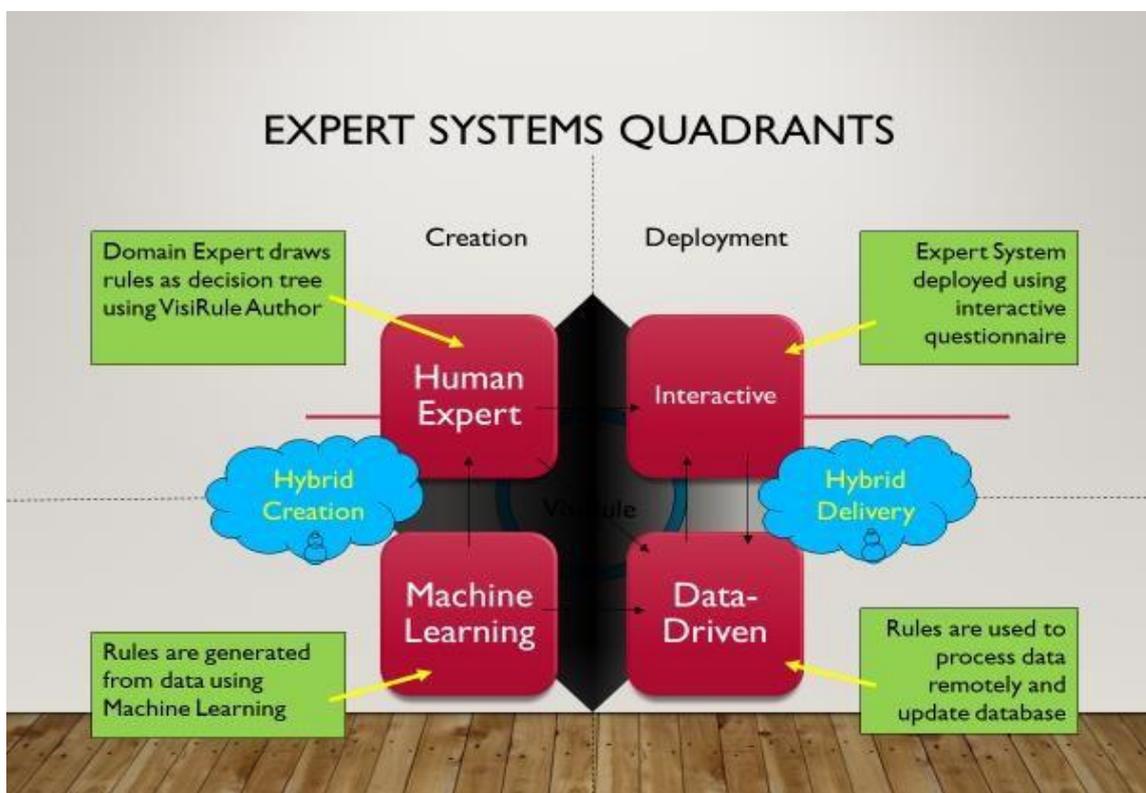
VisiRule charts can be immediately tested and executed within the Windows desktop authoring environment before uploading to the internet for general usage.

## Modularity

VisiRule supports multiple, linked charts. Complex problems can be split into smaller discrete problems where they are more easily tackled. Frequently re-occurring scenarios can be re-used in alternative situations.

## Publishing Knowledge

When a VisiRule chart is presented on the web, it can display an interactive map of the chart to act as a visual aid and help the user better understand their progress through the chart's logic, a bit like Google Maps.



## Exported Decision Tables

VisiRule charts can be run in both interactive and batch mode. The latter can compute against all possible inputs to produce complete Decision Tables. These can be used to verify and document the processes being modeled. In addition, you can export XML representation of your chart for editing or distribution purposes.

VisiRule Chart										
LEAVE										
Node	Type	Text			From	To	Size			
node1	expression	Name	>= 60		node16	node15	X	Y	Width	Height
		Prompt					36	72	48	24
		Explanation								
node2	expression	Name	>= 30		node14	node15	X	Y	Width	Height
		Prompt					84	216	48	24
		Explanation								
node3	expression	Name	? service < 15 and ? age < 45		node10	node13	X	Y	Width	Height
		Prompt					144	288	120	48
		Explanation								
node4	expression	Name	< 18		node16	node5	X	Y	Width	Height
		Prompt					372	72	48	24
		Explanation								
node5	end	Name	22 + 5		node4		X	Y	Width	Height
		Prompt					372	144	48	24
		Explanation								
node6	start	Name	Start			node16	X	Y	Width	Height
		Prompt					228	12	72	24
		Explanation								

## Automated Document Generation with VisiRule

Questions and Answers can be used to populate document templates to create document instances such as NDAs, Leases, Employment Contracts, Legal Agreements, Letters, reports and more.

VISIRule

[Restart](#)
[Logout](#)
[Review Answers](#)
[Go Back](#)
[Current Question](#)
[Reports](#)

Nda Theme: Dark

---

**Summary**

Generated Document

**1. Parties.**

This Non-Disclosure Agreement is between Peter, hitherto referred to as the 'Disclosing Party' and John hitherto referred to as the 'Receiving Party' for the purpose of protecting Confidential Information which shall be disclosed by the Disclosing Party to the Receiving Party .

**2. Definition of Confidential Information.**

For purposes of this Agreement, "Confidential Information" shall include all information or material that has or could have commercial value or other utility in the business in which Disclosing Party is engaged including specifically:

- i.cognitive models, ideas and any associated diagrams
- ii.intellectual artifacts such as ideas, constructs, concepts
- iii.specialized knowledge, understanding and informed insight
- iv.secret recipes, algorithms, implementation details that are not generally available
- v.intellectual property which may or may not be part of a patent application
- vi.business plans which may or may include marketing plans and marketing projections

As some of Confidential information is in written form, the Disclosing Party shall label or stamp the materials with the word "Confidential" or some similar warning.

## VisiRule for JavaScript

You can deliver many VisiRule charts client client using XML and a dedicated JavaScript evaluator.

### lawn\_advisor

Run

Does the lawn have moss or weeds?

no  
 yes

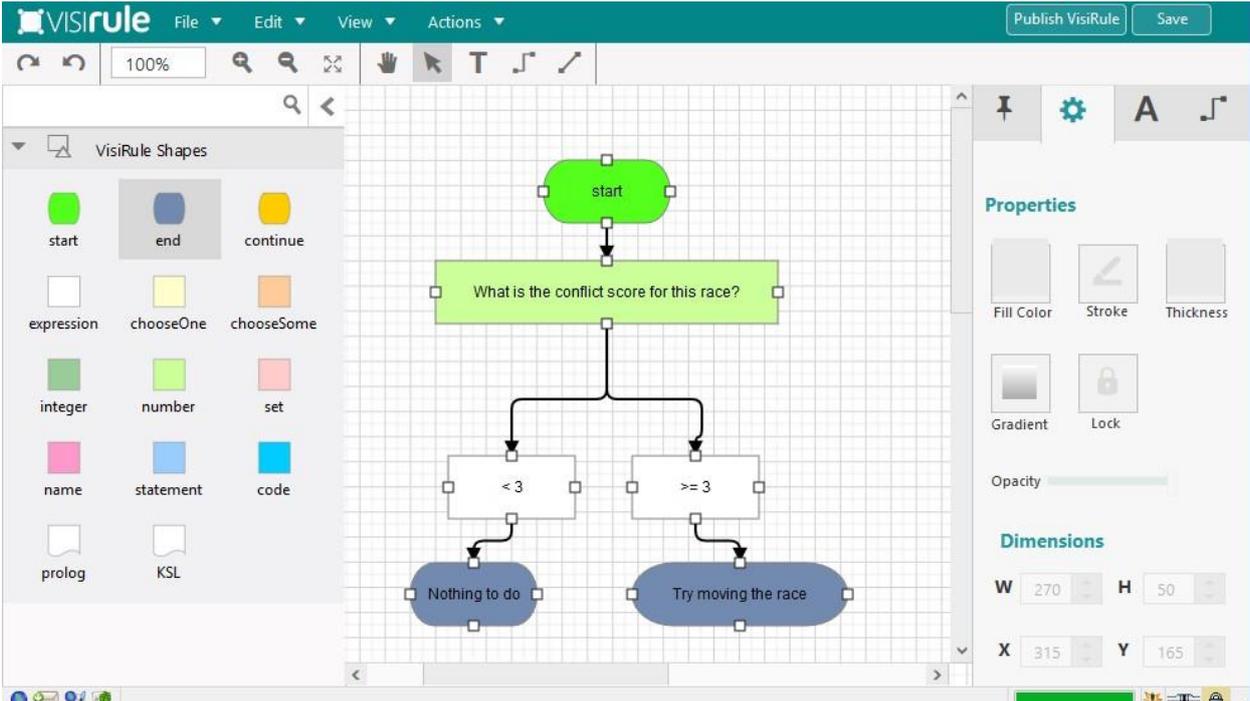
*What Is Moss?*

*Moss is a shallow-rooted plant that spreads by spores and root-like structures called rhizoids. It loves to grow during the gray, wet days of late fall and early spring. Moss won't overtake or kill your lawn, but instead will fill in the spaces where the lawn is thin. If your lawn is struggling and you have moss filling in the gaps, it's a good indication that you have a deeper problem, such as compacted soil, poor drainage, low soil pH, or not enough sunlight.*



## VisiRule365

VisiRule365 is a browser-based authoring tool which supports self-publishing of legal expert systems.



## Flex

Flex is a Low-Code expressive and powerful expert system toolkit which supports frame-based reasoning with inheritance, rule-based programming and data-driven procedures fully integrated within a logic programming environment.

### Knowledge Specification Language

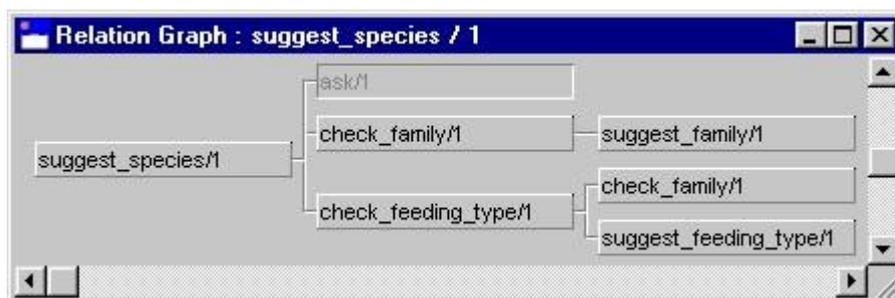
Flex has its own expressive English-like Knowledge Specification Language (KSL) for defining rules, frames and procedures. KSL enables developers to write simple and concise statements and produce a self-documenting knowledge-base. Mathematical, Boolean and conditional expressions and functions along with set abstractions are supported; and, the KSL is extendable through synonyms and templates. By supporting both logical and global variables, Flex avoids unnecessary duplication and so requires fewer rules.

### Rule-based inferencing

Flex includes support for different types of rule-based inferencing including forward-chaining production rules, ideal for data-driven reasoning and backward-chaining rules best suited for goal-based deduction. In addition, Flex is integrated with Flint which offers ways of handling inexact reasoning namely Fuzzy Logic, Bayesian Updating and Certainty Factors. This means you can describe rules and processes, even when you do not have a complete functional description.

### Development Tools

Flex contains an interactive development environment with program editor, frame browser, and debugger. Relationships and the connections between frames can be viewed graphically and inspected with object browsers.

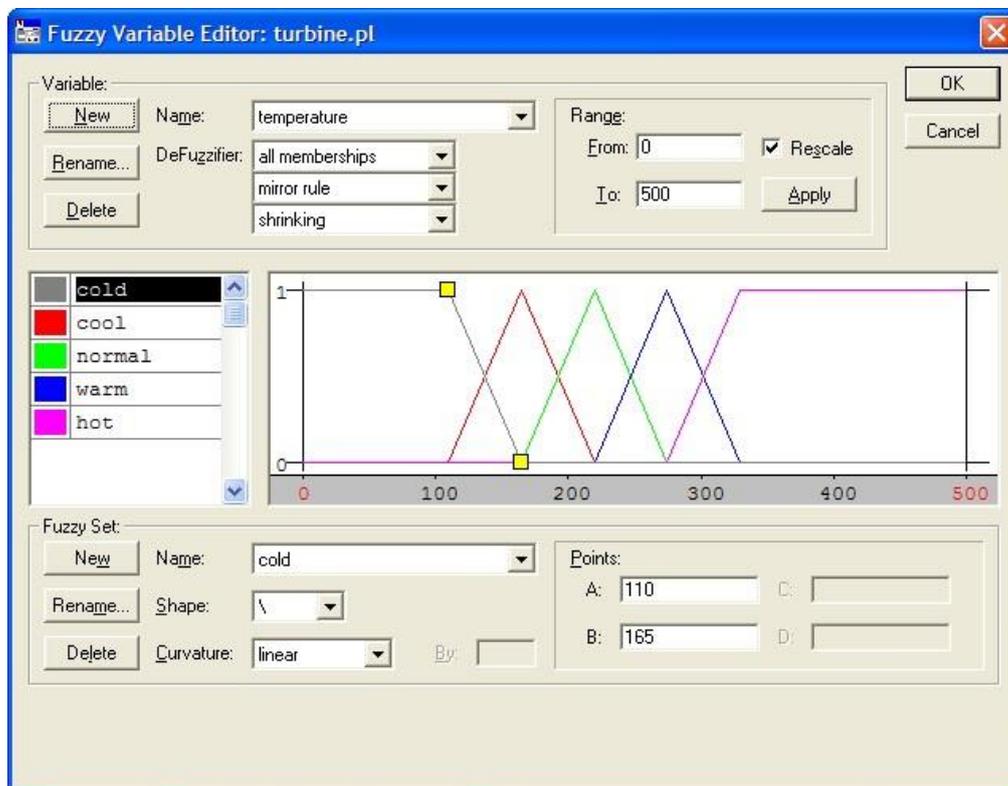


### Frame-based hierarchy

Frame hierarchies are similar to object-oriented hierarchies. They allow data to be stored in an abstract manner within a nested hierarchy with common properties automatically inherited through hierarchy. This avoids the unnecessary duplication of information, simplifies code and results in more readable and maintainable systems. Frames are complex data structures comparable to records in databases and can consist of any number of attributes or slots (comparable to fields).

## Uncertainty Handling and Fuzzy Logic

Flint provides support, in the way of rule engines and syntax, for various treatments of uncertainty including fuzzy logic, Bayesian updating and Certainty Factors; includes a graphical Fuzzy Editor for creating and editing fuzzy variables.



## Integration-friendly AI Toolkit

Flex goes beyond traditional expert system shells in that it allows you to access and modify its behavior through a layer of utility functions. The combination of Flex and Prolog, i.e. a hybrid expert system toolkit with a powerful general-purpose AI programming language, produces a functionally rich and versatile development environment. Add in VisiRule, and you have a unique multi-layered development framework.

## Expert Systems

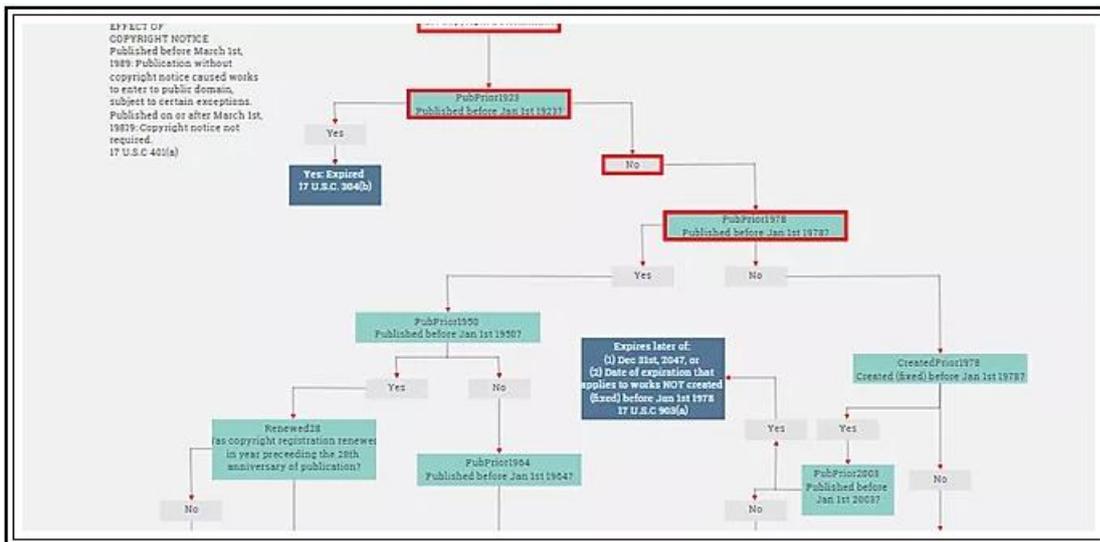
Expert systems allow the scarce and expensive knowledge of experts to be explicitly stored and made available to others who may be less experienced. They range from simple rule-based systems with flat data to large complex, developments taking many person-years. They typically have a set of if-then rules as the knowledge base, and a dedicated inference engine which provides the execution mechanism. This contrasts to conventional programs where domain knowledge and execution control are closely intertwined such that the knowledge is implicitly stored in the program. This explicit separation of the knowledge from the control mechanism makes it easier to examine knowledge, incorporate new knowledge, validate knowledge and modify existing knowledge.

## REST/JSON Support

JavaScript Object Notation (JSON) import and export is supported through a collection of special predicates, which allow JSON files or streams to be read and written as nested terms, to support REST and web-based technologies and connection to devices such as the iPhone.

## Delivery Options

Run-time delivery options include: compile to a self-contained Windows applications; embed within Java or C# or .NET using the LPA Intelligence Server, or publish directly on to the web using WebFlex. WebFlex supports multiple asynchronous clients running independent sessions thru a browser using IIS/CGI. HTML, CSS and JavaScript can be customized to control appearance. In addition, the run-time engine can be packaged within a container and embedded within Python or Java using Rest and Json.



**VISIRule** Insurance\_demo  
Restart Logout Review Answers Go Back Current Question Reports Theme Dark

**Question**  
What kind of contractor are you?  
 Heating AC  
 Plumber  
 General Contractor  
A further narrowing of the business classification for proper underwriting and pricing of the risk, as well as risk management recommendations. In some cases we do not have products to meet the needs of insureds. E.g. our products cater to trade contractors (e.g. heating and plumbing), rather than general contractors.  
Next

**Chart**  
 Display Chart

The chart shows a flow from "STARTS" to "Industry\_Group\_0n What Industry Group are you?". This node branches into "Contractor" and "Service". "Contractor" further branches into "Heating AC" and "Plumber". "Service" branches into "TBC".

## **LPA Prolog for Windows**

Prolog is a well-established AI language based on logic and offers a high-level and declarative way of specifying rules and data. LPA Prolog for Windows is a feature-rich compiler implementation with a multi-window editor dedicated to, and highly integrated with, all modern versions of Windows - both 32-bit and 64-bit.

### **Industrial Strength Engine**

LPA Prolog for Windows is a mature and robust industrial strength Prolog compiler and run-time system. It features a feature-rich integrated development environment, a wide variety of extra interface and data-oriented routines combined with an efficient and scaleable run-time delivery engine.

### **LPA Prolog: for Windows 10**

LPA Prolog is fully compatible with Windows 10, while still working well on earlier versions of Windows. The integrated development environment supports multiple program edit windows and various functions for displaying source code using a call graph, cross referencing code to discover undefined or redundant predicates. All of which helps make the maintenance of large, complex systems easier.

### **Debugging Tools**

The source-level debugger lets you scroll over program source code, inspect variable bindings, and view other dynamic information. A multi-level break facility allows you to break out to run supplementary queries. A traditional box model debugger, plus various special purpose debuggers are also provided.

### **Dedicated String Data Type**

To enable bulk text handling applications, LPA Prolog includes a dedicated text ``string`` data type. Strings can be used as input and output buffers. Strings provide a high bandwidth means for communication with low-level data types, such as files and screen buffers, as well as, the extensive library of GUI provided.

### **In-Depth Unicode Support**

WinProlog automatically handles Unicode, without penalizing ASCII-based applications. Unicode, ASCII and ISO/IEC 8859-1 text may be mixed freely, and used for predicates, variables and data.

### **Built-in XML Support**

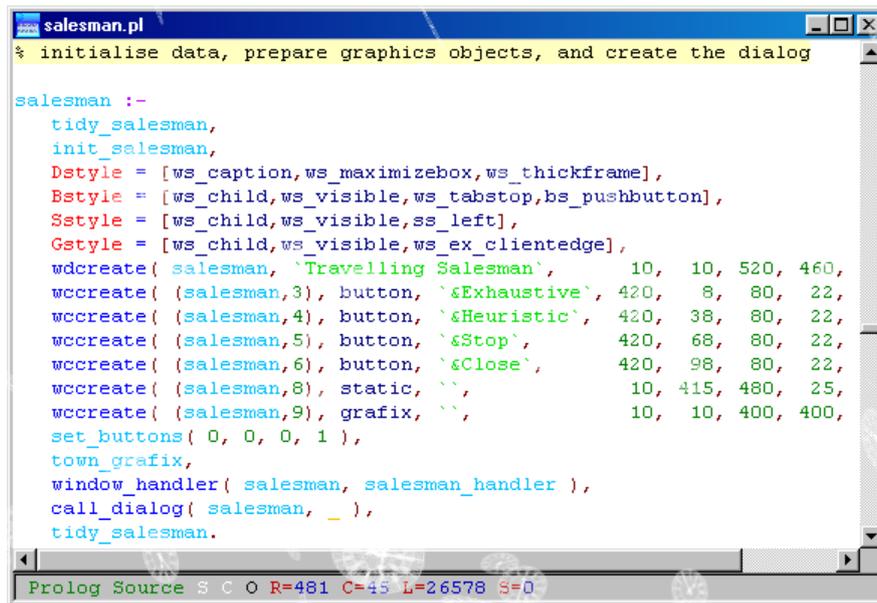
Extended Markup Language (XML) import and export is supported through a collection of special predicates, which allow XML files or streams to be read as tokens or nested terms, and written with or without indented formatting, to support web-centric and other data processing applications.

### **Socket & TCP/IP Support**

LPA Prolog has built-in Windows sockets support. Everything you need to write TCP/IP applications is built right into the Prolog kernel, and you can extend this functionality further with the Chimera Agent Toolkit.

## Dynamic Syntax Colouring

LPA Prolog makes programming easier. "Rich Syntax Colouring" helps you spot common program typos by identifying the text as variables, numbers, strings, predicates in real time, 'on-the-fly' as you edit your code.



```
salesman.pl
% initialise data, prepare graphics objects, and create the dialog

salesman :-
    tidy_salesman,
    init_salesman,
    Dstyle = [ws_caption,ws_maximizebox,ws_thickframe],
    Bstyle = [ws_child,ws_visible,ws_tabstop,bs_pushbutton],
    Sstyle = [ws_child,ws_visible,ss_left],
    Gstyle = [ws_child,ws_visible,ws_ex_clientedge],
    wcreate( salesman, `Travelling Salesman`, 10, 10, 520, 460,
    wcreate( (salesman,3), button, `&Exhaustive`, 420, 8, 80, 22,
    wcreate( (salesman,4), button, `&Heuristic`, 420, 38, 80, 22,
    wcreate( (salesman,5), button, `&Stop`, 420, 68, 80, 22,
    wcreate( (salesman,6), button, `&Close`, 420, 98, 80, 22,
    wcreate( (salesman,8), static, ``, 10, 415, 480, 25,
    wcreate( (salesman,9), grafix, ``, 10, 10, 400, 400,
    set_buttons( 0, 0, 0, 1 ),
    town_grafix,
    window_handler( salesman, salesman_handler ),
    call_dialog( salesman, _ ),
    tidy_salesman.

Prolog Source S C O R=481 C=45 L=26578 S=0
```

## Native x87 Floating Point Package

LPA Prolog includes a double-precision, floating-point arithmetic library for high-performance computations, with support for standard "calculator" functions, trigonometric and logarithmic functions, floating point to integer conversion and truncation functions, maximum and minimum functions, and pseudo random numbers.

## Optimized Performance

LPA Prolog uses an incremental compiler, and de-compiler, to support dynamic rule modeling. The Optimizing Compiler produces secure code. LPA Prolog also incorporates hashing technology to index very large in-memory databases and so provide efficient fast access to large corpora such as WordNet.

## Windows API

As well as providing high-level access to numerous functions, LPA Prolog lets you call external function directly, whether defined in the Windows API or an external DLL. The winapi/4 function provides a direct way to invoke external functions defined in DLLs, without having to write any parameter-translating ("glue") code in C/C++.

## DLL Prolog

DLL-PROLOG is a true Prolog compiler for Windows machines, available in both 32-bit and 64-bit versions, which shares its powerful Prolog engine with the all-bells-and-whistles product that is WIN-PROLOG, but without the latter's GUI features. Instead, DLL-PROLOG works directly through a simple but very powerful API that enables it to be embedded in other applications written in C, C++ and other development languages.

### **Case-Based Reasoning toolkit**

The CBR toolkit provides an API which allows you to retrieve closely matched records within commercial databases using SQL. You can define measures of similarity and relative importance for various dimensions or fields within the data. You can incorporate rules to help determine the relative similarity of records.

### **Data Mining toolkit**

The Data Mining toolkit provides routines to support the discovery of rules and knowledge within commercial databases using SQL and ODBC using Association Rule Mining discovery algorithms. Both the above toolkits include source code example applications.

### **ProData toolkit**

ProData provides seamless, tightly coupled, integration between the LPA Prolog and various commercial database systems such as SQL Server, Oracle, and MySQL thru ODBC. External data is treated as if it were asserted locally.

### **Chimera Agent toolkit**

Chimera is a powerful framework which supports the implementation of distributed multi-agent solutions. Using Chimera, you can build autonomous agents which can work on multiple connected machines across any TCP/IP network. Chimera allows you to use KQML, Fipa or almost any Agent Communication.

### **LPA Intelligence Server toolkit for Embeddability**

The LPA Intelligence Server toolkit allows you to embed Prolog-based components within many other applications written using VB.Net, C#, C, C++, Java, Delphi and Python/Flask and various other languages.

### **Windows Containers for Scalability**

You can deploy your VisiRule/Flex applications within Windows containers. Containers provide the tools to organize and build micro-services. They package up VisiRule/Flex applications within a complete file system that contains everything it needs to run: code, run-time, system tools and system libraries.

**LPA have been providing logic-based products and solutions since 1981 and have a proven track-record as a supplier of intelligent AI systems.**

**LPA products have been used in a wide-range of commercial and research applications from legal document assembly, environmental engineering, information modeling, disease diagnosis, fault diagnosis, to language analysis and hardware simulation.**

**LPA software is highly respected and used in teaching within many Universities and Research Institutes throughout the world.**

**LPA's goal is to provide innovative, integrated and intelligent solutions to business problems.**