exsys Corvid

Exsys Corvid
Expert System
Development &
Deployment Tools

Technology Overview

Expert Systems - Bringing Expertise to Those that Need It

Expert systems are computer programs that provide problem solving knowledge and situation-specific advice comparable to talking to a human expert. With Exsys Corvid® Knowledge Automation Expert Systems, this is done by an interactive session where the expert system asks the user questions. Their input is combined with the expert's decision making logic, which is contained in the system, to reach specific conclusions and advice that is then provided to the user.

Being able to automate complex decision-making tasks, diagnostics, and support is highly beneficial. Most companies and organizations rely on the Web as their primary means of reaching new and existing customers and employees, answering questions and providing support internally and externally. Web sites can easily provide plenty of data, but most real-world questions require more than data - they require knowledge and customized answers. Site visitors are often overloaded with too much information that they must read, understand and evaluate to make informed decisions. Customers and employees would prefer to talk to human experts capable of answering their questions, but that is difficult, costly and impractical to provide in most cases. Exsys Corvid online expert systems provide an effective and high R.O.I. alternative solution.

Exsys Corvid provides ways to capture expert knowledge and decision-making processes that allow computers to emulate the interaction people have with human experts to solve problems. Corvid software makes it very desirable and practical for firms to deliver expertise in interactive online systems.

With Corvid systems, potential customers, support staff, clients and employees are able to obtain problem-solving recommendations, as if they were consulting with a firms' best experts – and they're available 24 hours a day.

Corvid systems are portable across platforms for enterprise-wide integration and run from Web sites, intranets and client-side. They capture expertise and disseminate knowledge to handle mission-critical projects, automate routine tasks, assist in selection processes, analyze data streams, enhance customer/product support, and ensure policy and compliance - any area where consistent, logical decisions are needed.

Exsys Corvid Knowledge Automation and Inference Engines

"Knowledge Automation" is an approach to online interactive systems that rapidly and automatically provide each site visitor with tailored, reasoned answers and advice, based on expert knowledge.   Knowledge Automation makes it possible to provide expert consultations in a way that is automated, consistent and reliable 24/7. Knowledge Automation systems built with Exsys Corvid follow the same problem-solving logic and process as human experts; reaching the same conclusions and making consistent recommendations. Exsys Corvid makes it practical to build and field these systems on a Web site to provide assistance in any decision-making task, including diagnostics, customer support, regulatory compliance, pre-sales advice, HR questions and many other areas.

A human expert makes a complex decision by considering many things. Through experience and "know-how" an expert learns which factors are potentially relevant, the implications of certain elements or circumstances, and how they should be combined and weighted to reach a recommendation. Knowledge Automation Expert Systems capture this decision-making logic as rules that are processed by an Inference Engine.

The rules are "Heuristics" - If/Then rules of thumb that are the individual steps or factors that makeup the overall decision. In Exsys Corvid, rules are written in English (or whatever language is preferred) and Algebra. They are easy to read and very similar to what the expert would tell you if you asked, "How did you make that decision?" Exsys Corvid is designed to help developers (domain experts) build rules that are easy to read, understand and maintain.

The Corvid Inference Engine is the key to building and fielding knowledge automation expert systems. It is the bridge between rules that people can read and understand, and rules that the computer can use effectively.   It is able to process the rules to determine:

- What rules are relevant to a particular part of the decision-making problem
- How those rules can and should be used
- What data is needed to test the rules
- How that data can be obtained
- When a conclusion has been reached

In addition, the Inference Engine handles the end user interface, connections to other external IT resources such as databases and the online delivery of the system.

With Exsys Corvid all the developer has to do is write easily read rules describing the logic, and the Inference Engine does the rest. Without the inference Engine, the entire logical process would have to be converted and coded in a computer language, including the user interface, all internal interaction between logical elements and all external connections. This is possible for very simple logic, but extremely difficult for complex problems and always a maintenance nightmare.

Corvid's knowledge automation rules and Inference Engine remain the most effective way to capture and deliver expert decision-making knowledge.

Exsys Corvid - The Industry Standard Tool for Success

Exsys Corvid is a powerful and extensively proven software tool for building and fielding knowledge automation expert system applications. It is designed to be easy to learn and aimed at non-programmers. It enables the decision-making logic and process of the domain expert to be converted into a structured form that can be used by the Exsys Inference Engine to drive interactive online sessions that provide recommendations and advice to end users comparable to what they would get talking to the expert. The Java based Exsys Inference Engine makes it easy to deploy systems on a wide range of platforms and to integrate with a wide range of external programs.

One of the pioneers in the expert system field, Exsys Inc has been working with companies and organizations since 1983 to create development tools focused on what is needed to build practical real-world systems. Exsys Corvid is aimed at individuals and development teams responsible for building knowledge automation expert systems to distribute accessible top-level advice. It takes a very pragmatic approach to system development. It was created with only one goal in mind - make it easy for developers to successfully get systems built and fielded as quickly as possible.

Expert system development has 3 main parts:

- Fully capturing the decision-making logic and process of the domain expert
- Wrapping the system in a user interface with the desired look-and-feel for online deployment
- Integrating with other IT resources

Exsys Corvid has a well proven approach and methodology for each of these.

Capturing the Decision Making Logic

Fully capturing the domain expert's decision making logic and process is essential to building a system that can provide advice comparable to the expert. The rules in a system must be able to completely describe how the expert makes the decision. Many problems can seem simple at first, but when examined in detail have aspects that are subtle, complicated or probabilistic. Exsys Corvid is designed to handle even very complex logic, and the many systems built with it illustrate its flexibility and power.

The key to fully capturing the expert's decision-making logic is to describe it in the same way the expert thinks about it. Rules that are written this way are far more likely to be accurate and complete compared to rules that have been converted into a complex syntax or computer language. If a decision-making process is already documented in diagrams, SOPs or written regulations, it is best to have the rules match the terminology and structure of the existing documentation. When rules in a system are hard-coded in a computer language, it is very likely that errors will creep in and some nuances will be lost.

The approach used in Exsys Corvid is to write rules that are easily read, and that closely match the way the domain expert would explain their logic to another person. If there is existing documentation, the Corvid system can be built to incorporate it. Keeping the rules easy to read enables the domain expert to build, or directly participate in building the system and makes it much easier to fully capture the expert's logic. It also has major benefits in maintenance, where the meaning of the rules is clear and changes can be quickly and reliably implemented.

Corvid provides an intuitive development environment allowing domain experts to "describe" their decision making steps in a logical manner, much as they would to another person. There is no complex syntax to learn. The rules are described simply in English and Algebra. Tree-structured logic diagrams are used to describe individual sections of the process. Corvid allows the problem to be broken into small discrete parts for faster structured development. It combines a very flexible way to describe rules in its unique "Logic Block" structures and "Command Blocks" that provide procedural control on system execution through Corvid's Inference Engine using both Backward and Forward chaining as appropriate.

Heuristics and Rules

In expert system terminology, each of the expert's "rules of thumb" is a heuristic. That is a specific small fact that tells how to make a part of a decision. The combination of all the heuristics allows the overall decision-making problem to be solved. In our brain, people combine these individual heuristics intuitively and systematically. You don't have to stop and say, "Now I need to know this to help make that decision…", our brain just does it.

With Exsys Corvid building an expert system is simply identifying the individual decision steps and converting them into a form that a computer can use, but that is easy for people to read and understand.

There are many ways of describing the heuristics for a decision-making process, but the one that is usually most effective and efficient is the IF/THEN rule. This is a rule where there is an IF part that can be tested to be true or false based on the data for a specific case or situation. When the IF part is true, the statements in the THEN part are also considered true. For example, a basic rule might be:

```
IF
        It is raining
THEN
        You should wear a raincoat
```

With Exsys Corvid, the rules are very similar to the form that you would use to explain the heuristic to another person. For example, "If the investment customer has a high risk tolerance and requires rapid growth to reach their objectives, Mutual Fund X would be a good choice."

In a Corvid rule this would become:

```
IF
        The customer has high-risk tolerance
  AND   Meeting objectives requires rapid growth

THEN
        Mutual Fund X is a good choice
```

This rule includes a small amount of syntax, but it is still very easy to read and understand what it means. If you built similar rules for each of the heuristics in the decision-making process, you would have the logic for the expert system.

A rule may have one or more IF conditions, and one or more THEN conditions. The IF conditions are always Boolean tests that will evaluate to TRUE or FALSE. This can be done with Algebraic expressions, special Corvid functions to test values or simple tests to see if a particular value was selected. When there are multiple IF conditions, they are combined with AND and must all be true for the overall rule to be TRUE. (There are also ways to build conditions with OR and other logical operators when needed)

The THEN statements assign a value to a variable. This may be simply setting a value, adding content to a report, or adjusting a confidence (probability) value for a particular fact. When the IF conditions for a rule are TRUE, the assignments in the THEN statements are made, adding to the information the system has, or setting values that will be part of the results and system advice.

Each rule is a small part of the overall decision-making logic. The Corvid Inference Engine processes the rules to use them in the most effective way to solve a particular problem. In general, rules can be in any order and structured in any way - allowing the organization of the rules to match the way the domain expert approaches a problem, or existing documentation of the process.

Backward Chaining

Backward chaining is one of the most powerful features of the Corvid Inference Engine and is the main reason it is MUCH easier to solve complex problems with IF/THEN rules in Corvid than using IF/THEN statements in a programming language.

Backward chaining is conceptually quite simple. If the Corvid Inference Engine needs a particular value for what is is currently doing, it will check all the rules to see if there are any rules that could tell it that value. If it finds a rule, it will suspend what it is currently doing and try to evaluate the new rule. Once it has gotten the value it needs, it will return to what it was originally doing. This happens recursively, so if the new rule required a value that could be obtained from other rules, those would be tested, etc.

People do it all the time when making decisions. If you are diagnosing a machine that is not working, you might think: "Maybe it is the power supply", but if you saw the lights were still on, you would know it was getting power. You don't consciously think, "If the lights are on, then the machine is getting power", but if asked "How do you know the machine is getting power?", the answer would be the "lights". In a Corvid system there could be a rule:

```
IF
        The machine's lights are on
THEN
        The machine is getting power
```

This is a typical "heuristic" rule that describes a specific fact that might be used in making the decision.

Backward chaining is often referred to as "Goal Driven" and it can be thought of as a "To Do" list. A system will start with the top To Do list item being something like "Generate a report" or "Find the most likely diagnosis". The Corvid Inference Engine always tries to do whatever is at the top of the list. If the top item on the list requires some value that is not known, getting that value becomes the new top item on the list, pushing down the previous goal. The Inference Engine now focuses on that item, but it too may require other values that become the top values. As values are obtained from the rules, the items are removed from the To Do list, and the item below again becomes the top item. This continues with items being added and removed until all the items are removed and the system is done.

This may sound complicated, but it is all done automatically by the Inference Engine. All the developer has to do is provide the rules that will tell the system how to derive the values it needs. The order of the rules is not particularly important, since the Inference Engine will find whatever rules are appropriate when it needs them. This makes for a very "free-form" approach to writing the rules, which is very different from the highly order dependent way the logic has to be coded in a computer language.

Since rules will be used dynamically as needed, modules of rules can be created to solve parts of a problem and they will be used when (and if) they are needed. Interdependencies between values or rules do not have to be "hard coded" and happen automatically in Backward Chaining. This greatly simplifies development of complex systems.

Corvid also supports "Forward Chaining" which is a more procedural, order dependent execution of the rules. This can be combined with Backward Chaining to run top level rules in a particular order, this will allow backward chaining to be used to derive needed values from other rule modules. It provides the best of both approaches.

Variables

Corvid rules are built using variables. Corvid supports 7 types of variables, so many types of logical statements can be made.

Corvid Variable Types

Static List - Multiple-Choice list. (e.g. day of the week, yes/no, high/medium/low) Makes it easy to build structured logic diagrams and effective user interfaces.

Dynamic List - Multiple-choice list with the values defined dynamically during system runtime. The values may come from external sources such as databases or be set by the logic of the system.

Numeric List - A numeric value that can be used in formulas or Algebraic test expressions.

String - A string value that can hold any text, either input by the user or determined by the system. String functions allow parsing, testing and building string values.

Date - A date value that can be used in comparison (future/past, etc.) tests.

Collection - A list of strings values built-up during a system run. Various methods allow adding, removing and testing items in the list. Very useful for building dynamic reports such as text, HTML, RTF documents or XML.

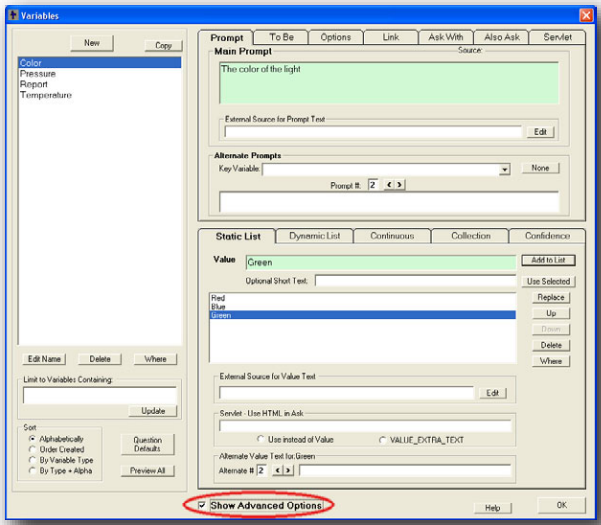
Confidence - A Variable that can be assigned a confidence value that reflects a degree of certainty. Various formulas can be used to combine the values assigned for an overall confidence for the Variable.

Corvid variables are indicated by the variable's name in square brackets, such as [Color], [Price], [Temperature_of_the_fluid]. Each variable type has its own methods and properties, and a wide range of functions available to use them in expressions and assignments.

The Corvid Variables window is used to add and edit variables. The various properties for the variable can be set to control how the variable will be used and displayed.

It allows setting:

- The text to associate with the variable
- Multiple choice variables, the value list
- How the variable uses backward chaining
- Default values
- Controls and "Look-and-Feel" for the user interface when asking for a value



Color
Pressure
Report
Temperature

Main Prompt: The color of the light

Prompt #: 2

Value: Green

Optional Short Text:

Red
Blue
Green

Alternate Value Text for:Green
Alternate #: 2

Sort: [✓ Alphabetically] Order Created, By Variable Type, By Type + Alpha

[✓ Show Advanced Options]

In addition, there are "Advance Options" for:

- Building systems that run in multiple languages
- Getting variable text from a database, XML file or other data sources
- Using the Corvid Servlet Runtime for more complex user interfaces

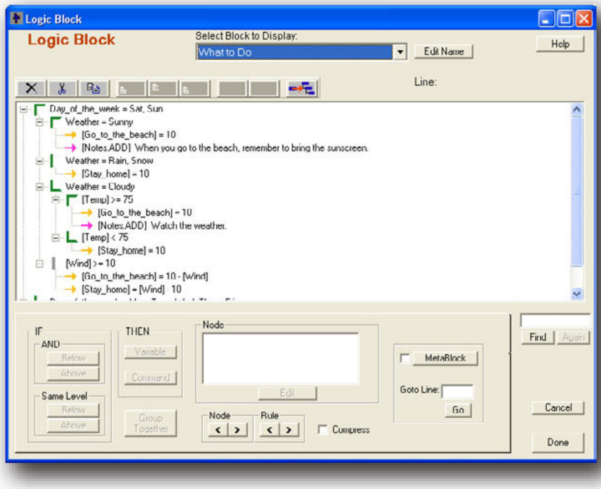
Confidence Variables

A very powerful feature of Exsys Corvid is that the rules can include a "confidence factor" for a particular answer. This enables expert systems to provide multiple recommendations with differing degrees of confidence. While in some cases it is possible to give a specific recommendation with absolute precision, the real world is not often so clear-cut. Often multiple recommendations are simultaneously possible and the system ranks them based on relative merit. Corvid provides many ways to combine and work with confidence and probability. Sometimes a system may have statistical data that can be handled using the exact laws of probability. However, often the domain expert will have more generalized knowledge of the likelihood of various situations based on their experience. Corvid has various "Confidence modes" specifically for capturing and using this less statistical, but highly valuable expert knowledge.

The ability to handle confidence factors in Corvid expert systems provides a much more effective way to build systems that emulate the real world and give the type of recommendations that human experts would.

Logic Blocks

Exsys Corvid has a unique way to define, organize and structure rules into logically related modules. A Logic Block is made up of one or more structured logic diagrams. The logic may be a complex branching tree that systematically covers all possible input cases or a simple structure that corresponds more to a single rule. The rules in the Logic Block usually all relate to one aspect of the decision, and can be referenced in the system by simply calling that block.

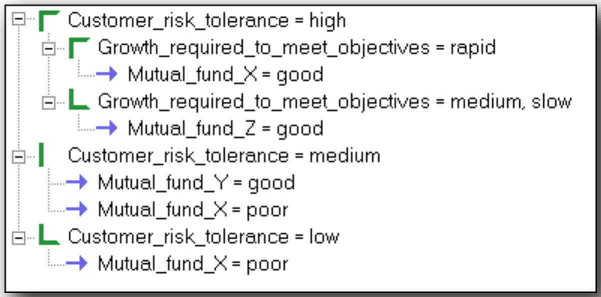


```
Day_of_the_week = Sat, Sun
  Weather = Sunny
    [Go_to_the_beach] = 10
    [Notes.ADD] When you go to the beach, remember to bring the sunscreen.
  Weather = Rain, Snow
    [Stay_home] = 10
  Weather = Cloudy
    [Temp] >= 75
      [Go_to_the_beach] = 10
      [Notes.ADD] Watch the weather.
    [Temp] < 75
      [Stay_home] = 10
  [Wind] >= 10
    [Go_to_the_beach] = 10 - [Wind]
    [Stay_home] = [Wind]  10
```

For example, suppose the logic to derive a particular fact calls for 3 multi-branched trees and several other individual rules to cover special cases. A single Logic Block can be built with all the required trees and rules. When the Variable's value is needed it will be called.

Also, since this block is a module for the derivation of the value of a variable(s), it will automatically be used by any other Logic Blocks through backward chaining. If the details of the derivation logic change, all that is needed is to modify the rules in the single Logic Block, and the change will automatically propagate to all the rules that use the value.

The rules in the Logic Blocks typically cover a group of related heuristics, and should explain how to resolve each potential decision point in a system. Logic Blocks are typically made up of tree diagrams that organize a group of related rules.



```
Customer_risk_tolerance = high
  Growth_required_to_meet_objectives = rapid
    Mutual_fund_X = good
  Growth_required_to_meet_objectives = medium, slow
    Mutual_fund_Z = good
Customer_risk_tolerance = medium
  Mutual_fund_Y = good
  Mutual_fund_X = poor
Customer_risk_tolerance = low
  Mutual_fund_X = poor
```

Logic Blocks diagram the tree structure by using brackets that indicate related groups of IF conditions. Indentation indicates that the IF condition is ANDed with its "parent" node, allowing the tree structure to be built out. Arrows indicate THEN assignments.

The Logic Block displayed here is equivalent to the following 4 rules:

IF
The customer has high-risk tolerance
AND Meeting objectives requires rapid growth

THEN
Mutual Fund X is a good choice

IF
The customer has high-risk tolerance
AND Meeting objectives requires only medium or slow growth

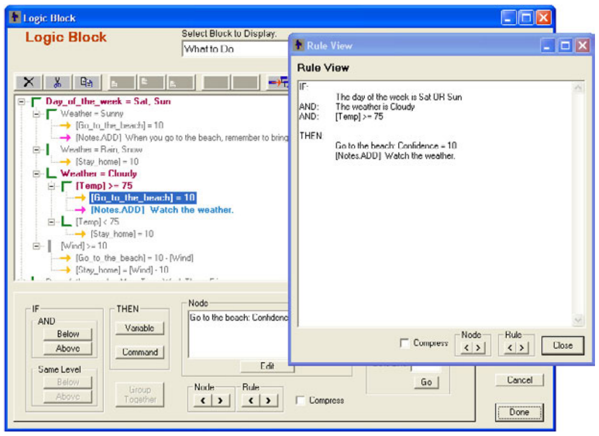
THEN
Mutual Fund Z is a good choice

IF
    The customer has medium risk tolerance
THEN
    Mutual Fund Y is a good choice
AND Mutual Fund X is a poor choice

IF
    The customer has low risk tolerance
THEN
    Mutual Fund X is a poor choice

This makes a very compact representation. Related IF conditions can be added as a group, typically various possible values for a list variable or a group of related Algebraic expressions that cover various possible situations. Adding IF conditions as related groups quickly builds out the tree and color highlighting is used to show the branches that have not yet been completed. When the Logic Block is complete, all the brackets should be green and it will be able to handle any combination of user input.

To make it easier to see the individual rules built by a Logic Block, just click on a node and Corvid both highlights the "parent" conditions that makeup the rule and displays the full text of the individual rule in the "Rule View" window. The rules in the system can be examined either way.



```
Day_of_the_week = Sat, Sun
  Weather = Sunny
    [Go_to_the_beach] = 10
    [Notes.ADD] When you go to the beach, remember to bring
  Weather = Rain, Snow
    [Stay_home] = 10
  Weather = Cloudy
    [Temp] >= 75
      [Go_to_the_beach] = 10
      [Notes.ADD] Watch the weather.
    [Temp] < 75
      [Stay_home] = 10
  [Wind] >= 10
    [Go_to_the_beach] = 10 - [Wind]
    [Stay_home] = [Wind] - 10
```

```
IF:
        The day of the week is Sat OR Sun
AND:    The weather is Cloudy
AND:    [Temp] >= 75

THEN
        Go to the beach: Confidence = 10
        [Notes.ADD] Watch the weather.
```

Corvid's Logic Block allow modules of logic to be built quickly that are automatically combined through backward chaining to solve larger, complex problems.

Well written Logic Blocks are easy to read and maintain. Building them simplifies creating logic that will cover all possible situations and helps the domain expert to fully describe their decision making process.

MetaBlocks

Corvid Logic Blocks also support MetaBlocks, a unique capability that provides a way to apply high-level generic logic to data on multiple items stored in a spreadsheet or XML file. MetaBlocks provide a way to efficiently rank and select among a group of items based on user criteria.

This is particularly effective for building pre-sales product selection expert systems, and works for some types of probabilistic diagnostic systems. The Logic Block contains "generic" logic on how to decide among products based on users requirements. The data on the individual products or items can be stored in a spreadsheet or XML file. This logic is written in a way that uses product parameters, rather than specific data for an individual product.

For example, a MetaBlock rule might be: **If the customer's budget is less than half the price of the product, the product should not be recommended**:

IF
    The customer budget is less than 50% of the product price
THEN
    The product should not be recommended

This is a general rule and does not "hard code" any price information on a specific product. In MetaBlock form this would be:

IF
    [Customer_Budget] < .5 * {Price}
THEN
    [Product_ranking] = -100

The "Price" parameter in { }, will be replaced by the value from the "Price" column from the spreadsheet or XML data for each specific product. The -100 value for the confidence variable [Product_ranking] would greatly lower the "probability" of the specific product being recommended. A real system would have many rules on various aspects of the product and how well they match the customer's requirements.

The customer will be asked for their desired product specifications. Then the MetaBlock rules are applied sequentially to data on each products stored in a spreadsheet or XML file. As each row (product) in the spreadsheet is analyzed by the MetaBlock it is ranked as to how well it matches the customer's requirements. In the end, the products are sorted according to their ranking to find the best matches to the user's requirements.

Unlike database approaches, the MetaBlock systems will always find the "Best" matches, even though no individual product may be a perfect match. Along with recommending products, the system can easily display a report on each product describing how well it matches the customer's requests. If a product is a good match for desired functions, but 20% over their budget, this could be explained to the user. This is a much better emulation of the interaction a customer would have with a good salesperson who would always try to find the best overall product for the customer.

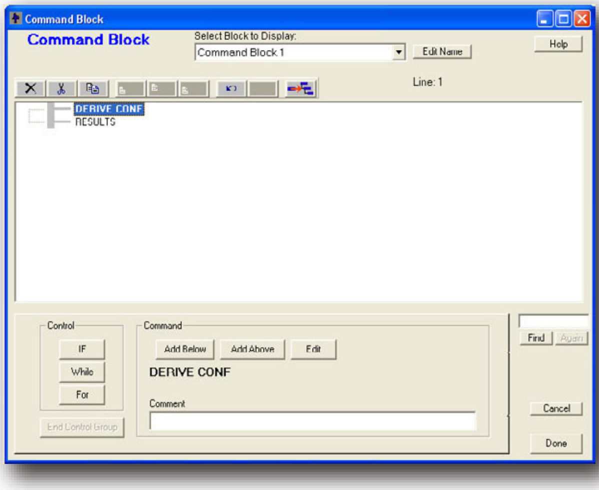
Product selection systems using only a database approach often concludes "No perfect match - there is nothing to sell you". A Corvid system will always find the best match and explain how and why it is best, helping inform a customer and making them far more likely to buy. Also the Corvid system allows the customer to specify more criteria to define exactly what they would most like. A MetaBlock system can incorporate the expert knowledge of top salespeople on what factors are the most important in making the decision, or allow the customer to provide their own weighting factors.

As new products need to be added, or product features change, all that has to be done is updating the spreadsheet or XML file. As long as the logic for selecting a product does not change, the MetaBlock rules stay the same. This makes it easy to maintain systems even when the products are rapidly changing. A system might even incorporate inventory, configuration, delivery time and other dynamic real-time factors in making a decision. Also, since a URL references the "spreadsheet" or XML file, it can be dynamically created by external programs.

If your web site is how you reach your customers, and you provide a range of potentially confusing products, Corvid MetaBlocks are an ideal way to deliver the precise advice and analysis customers need to make purchasing decisions.

Command Blocks

Command Blocks control the procedural flow of the system. The Logic Blocks in a system provide the rules of <u>how</u> to make a decision. Command Blocks tell the system what to do and how the rules should be used. They describe precisely how the system will operate independent of the actual rules in the system. Separating the Logic and Command Blocks greatly simplifies system development. This is especially true if a system has complex procedural requirements for integration, interfaces, monitoring or looping.



```
DERIVE CONF
RESULTS
```

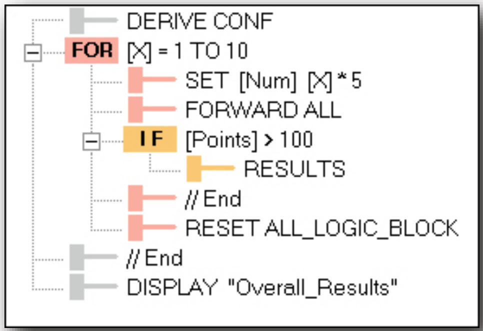
DERIVE CONF

Command Blocks control the procedural flow of the system including setting what variables are the "goals" for backward chaining, overriding the default operation of the Inference Engine, running Logic Blocks in a particular order, controlling how results will be displayed and external interface calls.

Command Blocks can be a single command such as backward chaining on all Confidence Variables, up to much more complex systems that use While and For loops, conditional branching, displaying intermediate results, etc.

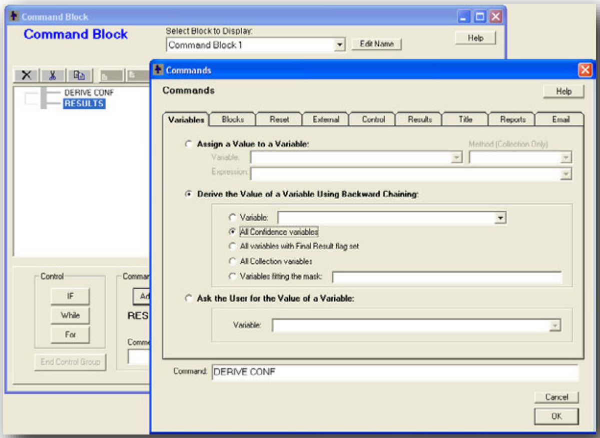
The Command Block provides a graphical development interface to describe the procedural operations, no matter how complex they get. Command Blocks are built and edited in the Corvid Command Block window. This window displays the command structure in a visual interface.

Conditional branches and loops are color coded and easy to see.



```
DERIVE CONF
FOR [X] = 1 TO 10
    SET [Num] [X] * 5
    FORWARD ALL
    I F [Points] > 100
        RESULTS
    // End
    RESET ALL_LOGIC_BLOCK
// End
DISPLAY "Overall_Results"
```

The Command Builder window enables all Corvid commands to be easily built with a few mouse clicks – no complex syntax to learn, or remember. Using this window helps ensure that the commands are syntactically correct. Each command also can have an optional note, making the Command Block even easier to understand. Simply work through the tabs:



Command Block

Select Block to Display: Command Block 1

DERIVE CONF
RESULTS

Commands

Variables | Blocks | Reset | External | Control | Results | Title | Reports | Email

( ) Assign a Value to a Variable:
Variable:
Expression:
Method (Collection Only)

(•) Derive the Value of a Variable Using Backward Chaining:
( ) Variable:
(•) All Confidence variables
( ) All variables with Final Result flag set
( ) All Collection variables
( ) Variables fitting the mask:

( ) Ask the User for the Value of a Variable:
Variable:

Command: DERIVE CONF

Variables Tab - Builds commands that set or drive the value for a Variable, or force the Variable to be asked of the user.

Blocks Tab - Builds commands that run a Logic Block in forward chaining mode or execute other Command Blocks.

Reset Tab - Allows data or blocks to be cleared for reuse when looping or monitoring.

External Tab - Allows commands to be added that call databases, XML files, other applets, and external programs.

Control Tab - Provides ways to override the default options for backward chaining.

Results Tab - Provides ways to display the results of a system.

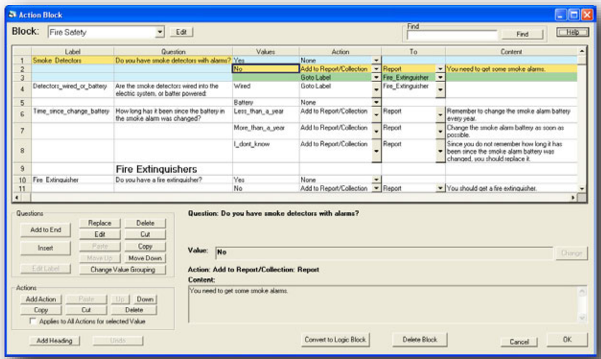
Title Tab - Allows the creation and display of a title page at the start of a run.

Reports Tab - Commands for creating and displaying reports in text, HTML, RTF and PDF formats.

Email Tab - Commands for automatically creating and sending emails.

Action Blocks and Smart Questionnaires

In addition to Logic Blocks, Corvid provides another way to describe logic, aimed primarily for Smart Questionnaire systems. Action Blocks use a spreadsheet approach to listing the questions to ask and the actions to take based on the various answers. Actions can include adding content to a report, asking additional or more detailed questions, skipping over unnecessary questions, setting values or counters, etc.



Action Block

Block: Fire Safety

	Label	Question	Values	Action	To	Content
1	Smoke_Detectors	Do you have smoke detectors with alarms?	Yes	None		
2			No	Add to Report/Collection	Report	You need to get some smoke alarms.
3				Goto Label	Fire_Extinguisher	
4	Detectors_wired_or_battery	Are the smoke detectors wired into the electric system, or batter powered:	Wired	Goto Label	Fire_Extinguisher	
5			Battery	None		
6	Time_since_change_battery	How long has it been since the battery in the smoke alarm was changed?	Less_than_a_year	Add to Report/Collection	Report	Remember to change the smoke alarm battery every year.
7			More_than_a_year	Add to Report/Collection	Report	Change the smoke alarm battery as soon as possible.
8			I_dont_know	Add to Report/Collection	Report	Since you do not remember how long it has been since the smoke alarm battery was changed, you should replace it.
9		Fire Extinguishers				
10	Fire_Extinguisher	Do you have a fire extinguisher?	Yes	None		
11			No	Add to Report/Collection	Report	You should get a fire extinguisher.

Question: Do you have smoke detectors with alarms?

Value: No

Action: Add to Report/Collection: Report

Content: You need to get some smoke alarms.

Action Blocks are designed to be run with forward chaining, but can use backward chaining to derive needed values. They make it very easy to convert questionnaires or interview procedures into online systems that ask questions “intelligently”, build reports and integrate with other IT resources.

Controlling The End User Interface

The way an expert system interacts with the user, and integrates into a website or IT infrastructure is critical to system success. Corvid provides a wide range of ways to customize the look and feel of a system to fit into existing web sites or to achieve the look designers want.

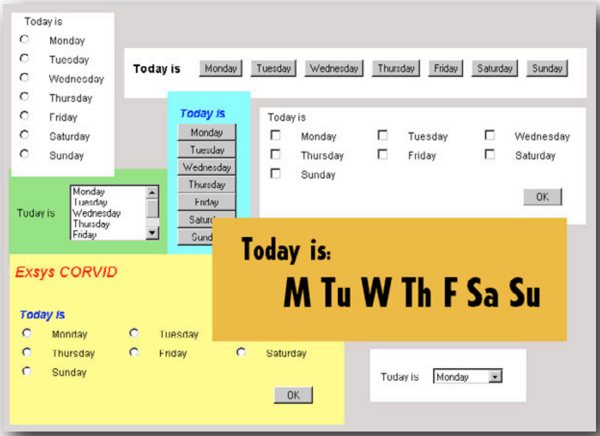
A Corvid system typically interacts with the end user by displaying information screens, asking questions, displaying messages or results, and writing reports. Corvid provides 5 main options for system delivery.

- Running as a Java Applet in a web page
- Running as a Java Servlet using HTML
- Running as a Java Servlet using Adobe Flash
- Running standalone (off-line) as a Java executable
- Embedded under another program that provides the end user interface

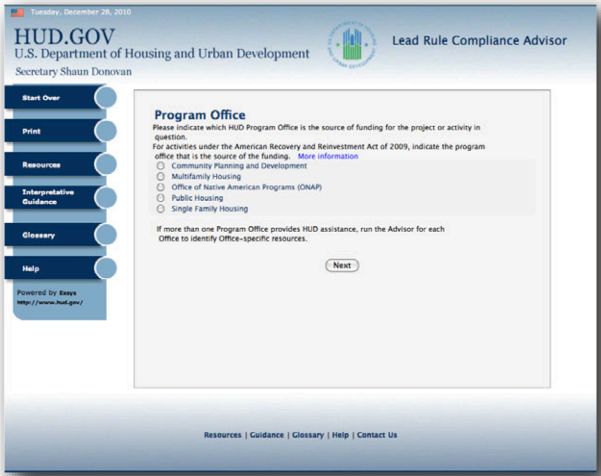
Running as a Java Applet in a Web Page

Running a system with the Corvid Applet Runtime is the default and easiest option. When a system is built, just click the "Run" icon and Corvid automatically builds everything needed to field the system as an applet on a web server. Corvid will build an HTML page that contains a Java Applet where the system runs.

The system content in the applet window can be controlled by setting text fonts, colors, positions, adding images, etc. - very much like setting the look of a word processing document. No knowledge of HTML is required.



Various ways to ask the same question



Program Office

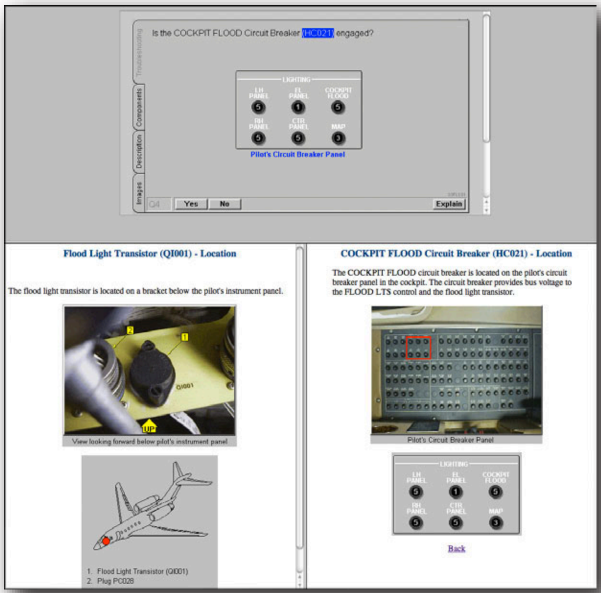
Please indicate which HUD Program Office is the source of funding for the project or activity in question.
For activities under the American Recovery and Reinvestment Act of 2009, indicate the program office that is the source of the funding. More information

- ○ Community Planning and Development
- ○ Multifamily Housing
- ○ Office of Native American Programs (ONAP)
- ○ Public Housing
- ○ Single Family Housing

If more than one Program Office provides HUD assistance, run the Advisor for each Office to identify Office-specific resources.

Applet Compliance System

In addition, the outer HTML page surrounding the applet can be edited to match the look of a site. This can be done independent of the Corvid system and can incorporate any HTML, CSS, JavaScript, etc supported by the browser.


Is the COCKPIT FLOOD Circuit Breaker (HC021) engaged?

Flood Light Transistor (QI001) - Location

The flood light transistor is located on a bracket below the pilot's instrument panel.

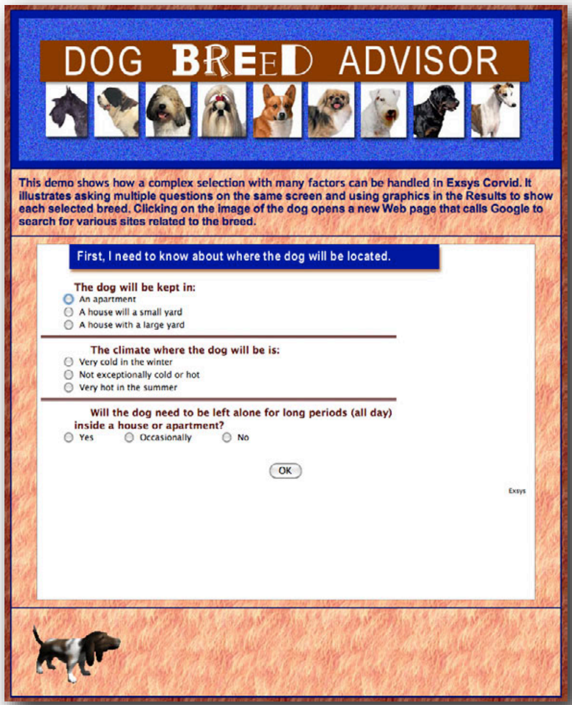
2
1
QI001
UP
View looking forward below pilot's instrument panel
1. Flood Light Transistor (QI001)
2. Plug PC028

COCKPIT FLOOD Circuit Breaker (HC021) - Location

The COCKPIT FLOOD circuit breaker is located on the pilot's circuit breaker panel in the cockpit. The circuit breaker provides bus voltage to the FLOOD LTS control and the flood light transistor.

Cesura diagnostic system using frames and diagrams

The Corvid Applet Runtime runs on the client machine making it very scalable since all the server has to do is serve the content. It can be distributed on any server capable of serving HTML pages, without any special server support for Java.



This demo shows how a complex selection with many factors can be handled in Exsys Corvid. It illustrates asking multiple questions on the same screen and using graphics in the Results to show each selected breed. Clicking on the image of the dog opens a new Web page that calls Google to search for various sites related to the breed.

First, I need to know about where the dog will be located.

The dog will be kept in:
- ● An apartment
- ○ A house will a small yard
- ○ A house with a large yard

The climate where the dog will be is:
- ○ Very cold in the winter
- ○ Not exceptionally cold or hot
- ○ Very hot in the summer

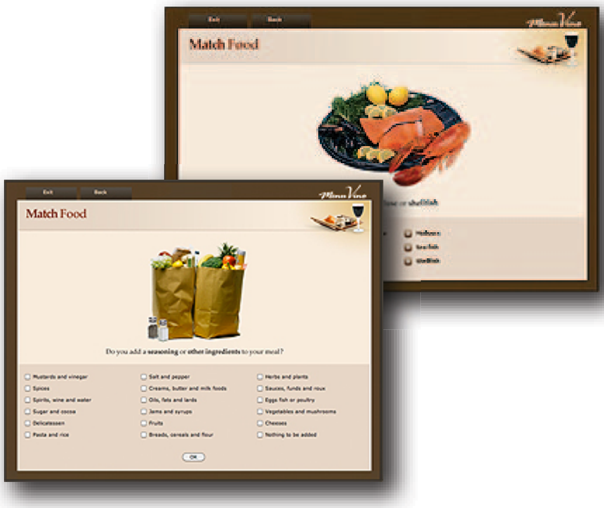
Will the dog need to be left alone for long periods (all day) inside a house or apartment?
○ Yes ○ Occasionally ○ No

OK

Exsys

Selection system with HTML
design elements and animation

Using a combination of controlling text styles, placement and images with HTML for the rest of the page, quite complex designs can be fielded. For more complex looks, Corvid supports image maps that can be used to ask questions graphically and can use frames to have various parts of the page displaying supporting content for the system that automatically synchronizes with the Corvid system. Both the system content and HTML page can include links to other pages or explanatory content.



Do you add a **seasoning** or **other ingredients** to your meal?

☐ Mustards and vinegar	☐ Salt and pepper	☐ Herbs and plants
☐ Spices	☐ Creams, butter and milk foods	☐ Sauces, funds and roux
☐ Spirits, wine and water	☐ Oils, fats and lards	☐ Eggs fish or poultry
☐ Sugar and cocoa	☐ Jams and syrups	☐ Vegetables and mushrooms
☐ Delicatessen	☐ Fruits	☐ Cheeses
☐ Pasta and rice	☐ Breads, cereals and flour	☐ Nothing to be added

OK

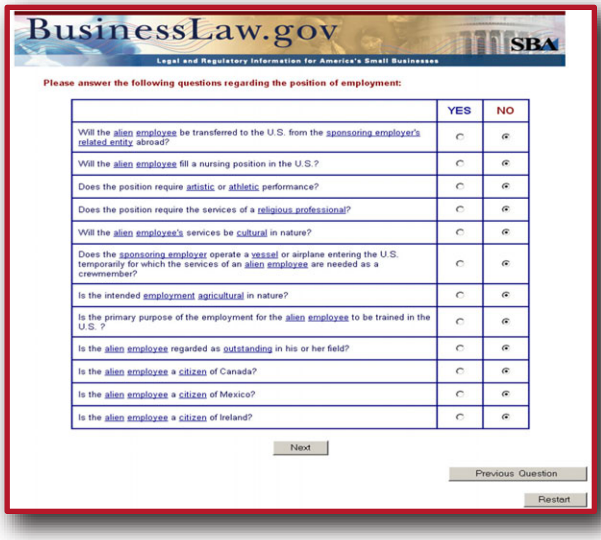
Commercial wine selection system
utilizing the Corvid Servlet Runtime

Running as a Java Servlet Using HTML

The Exsys Corvid Servlet Runtime provides another way to run Corvid systems. The Corvid Inference Engine runs in a Java Servlet on the server. The user interface is designed using HTML "Template" screens which create HTML forms that are sent to the end user. Templates are just standard HTML, but include a few special Corvid commands. The Corvid commands are added as HTML comments so templates can be edited with standard HTML editors. Sample templates with various designs are included with the Corvid Servlet Runtime. Templates can use CSS, JavaScript, AJAX and anything else the browser supports.

The servlet provides far more design options than running as an applet. It also provides more system security and portability as only HTML forms are sent to the end user. Since the system is run on the server, connections with other server resources such as databases can be made more directly and securely. There is no requirement for the end user machine to support Java or Applets. Servlet Runtime based systems can be run on iPhones, iPads and other potable devices that do not have Java support.

The Servlet Runtime allows very intricate designs to be implemented, especially those calling for tables or CSS positioning. JavaScript can be used in the pages to create more interactive interfaces or ones which validate user input before sending it to the server.


BusinessLaw.gov
SBA
Legal and Regulatory Information for America's Small Businesses

Please answer the following questions regarding the position of employment:

	YES	NO
Will the alien employee be transferred to the U.S. from the sponsoring employer's related entity abroad?	○	●
Will the alien employee fill a nursing position in the U.S.?	○	●
Does the position require artistic or athletic performance?	○	●
Does the position require the services of a religious professional?	○	●
Will the alien employee's services be cultural in nature?	○	●
Does the sponsoring employer operate a vessel or airplane entering the U.S. temporarily for which the services of an alien employee are needed as a crewmember?	○	●
Is the intended employment agricultural in nature?	○	●
Is the primary purpose of the employment for the alien employee to be trained in the U.S. ?	○	●
Is the alien employee regarded as outstanding in his or her field?	○	●
Is the alien employee a citizen of Canada?	○	●
Is the alien employee a citizen of Mexico?	○	●
Is the alien employee a citizen of Ireland?	○	●

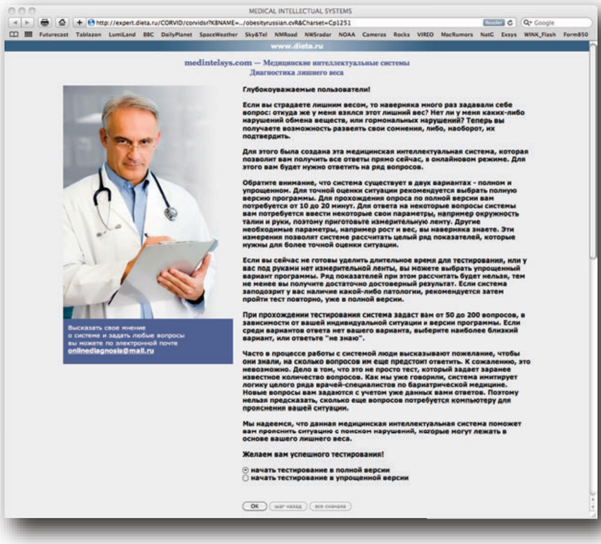
Next

Previous Question

Restart

SBA system using Corvid Servlet Runtime,
with multiple questions being asked
on a single page

The Corvid Servlet Runtime works by sending an HTML form to the end user machine. When the user answers the questions or indicates they are ready for the next screen, the form sends data back to the Corvid Servlet along with a session identifier for the user. Corvid takes the stored session data, and new information that was provided and continues processing that session. This may lead to needing to ask additional questions which are then asked of the end user by sending another form. While the user is reading and responding to the questions, that session is inactive on the server, so the process is highly scalable using the underlying Java Servlet technology. When the system has finished processing, conclusions or a report will be displayed to the end user. Reports can be in HTML, RTF or PDF formats. RTF and PDF reports are automatically opened by the end users browser in the associated programs (typically MS Word and Adobe Acrobat).


MEDICAL INTELLECTUAL SYSTEMS
http://expert.dieta.ru/CORVID/corvidsr?KBNAME=../obesityrussian.cvR&Charset=Cp1251
Futurecast Tablazon LumiLand BBC DailyPlanet SpaceWeather Sky&Tel NMRoad NWSradar NOAA Cameras Rocks VIREO MacRumors NatC Exsys WINK_Flash Form850
www.dieta.ru

medintelsys.com — Медицинские интеллектуальные системы
Диагностика лишнего веса

Высказать свое мнение
о системе и задать любые вопросы
вы можете по электронной почте
onlinediagnosis@mail.ru

Глубокоуважаемые пользователи!

Если вы страдаете лишним весом, то наверняка много раз задавали себе вопрос: откуда же у меня взялся этот лишний вес? Нет ли у меня каких-либо нарушений обмена веществ, или гормональных нарушений? Теперь вы получаете возможность развеять свои сомнения, либо, наоборот, их подтвердить.

Для этого была создана эта медицинская интеллектуальная система, которая позволит вам получить все ответы прямо сейчас, в онлайновом режиме. Для этого вам будет нужно ответить на ряд вопросов.

Обратите внимание, что система существует в двух вариантах - полном и упрощенном. Для точной оценки ситуации рекомендуется выбрать полную версию программы. Для прохождения опроса по полной версии вам потребуется от 10 до 20 минут. Для ответа на некоторые вопросы системы вам потребуется ввести некоторые свои параметры, например окружность талии и руки, поэтому приготовьте измерительную ленту. Другие необходимые параметры, например рост и вес, вы наверняка знаете. Эти измерения позволят системе рассчитать целый ряд показателей, которые нужны для более точной оценки ситуации.

Если вы сейчас не готовы уделить длительное время для тестирования, или у вас под руками нет измерительной ленты, вы можете выбрать упрощенный вариант программы. Ряд показателей при этом рассчитать будет нельзя, тем не менее вы получите достаточно достоверный результат. Если система заподозрит у вас наличие какой-либо патологии, рекомендуется затем пройти тест повторно, уже в полной версии.

При прохождении тестирования система задаст вам от 50 до 200 вопросов, в зависимости от вашей индивидуальной ситуации и версии программы. Если среди вариантов ответа нет вашего варианта, выберите наиболее близкий вариант, или ответьте "не знаю".

Часто в процессе работы с системой люди высказывают пожелание, чтобы они знали, на сколько вопросов им еще предстоит ответить. К сожалению, это невозможно. Дело в том, что это не просто тест, который задает заранее известное количество вопросов. Как мы уже говорили, система имитирует логику целого ряда врачей-специалистов по бариатрической медицине. Новые вопросы вам задаются с учетом уже данных вами ответов. Поэтому нельзя предсказать, сколько еще вопросов потребуется компьютеру для прояснения вашей ситуации.

Мы надеемся, что данная медицинская интеллектуальная система поможет вам прояснить ситуацию с поиском нарушений, которые могут лежать в основе вашего лишнего веса.

Желаем вам успешного тестирования!

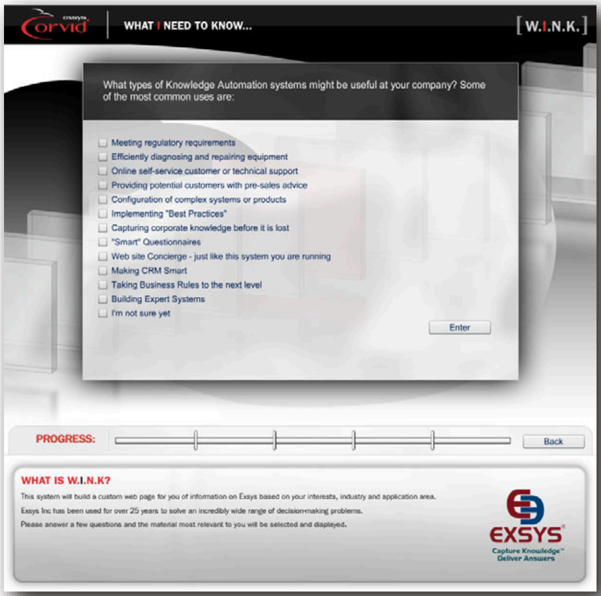
[✓ начать тестирование в полной версии]
начать тестирование в упрощенной версии

Systems can run in any language such as this Russian healthcare system

The Corvid Servlet Runtime requires support on the server for Java Servlets, such as Tomcat, Glassfish or IBM WebSphere.

Running as a Java Servlet Using Adobe® Flash®

The Exsys Corvid Servlet Runtime can also be used to field systems that use Adobe Flash for the user interface. This allows a more elaborate end user interaction with animation, complex graphics and interactions using the extensive options of Adobe Flash.



What types of Knowledge Automation systems might be useful at your company? Some of the most common uses are:

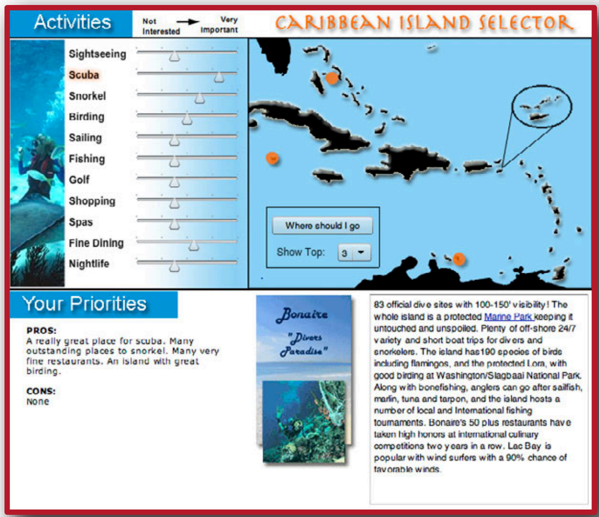
- ☐ Meeting regulatory requirements
- ☐ Efficiently diagnosing and repairing equipment
- ☐ Online self-service customer or technical support
- ☐ Providing potential customers with pre-sales advice
- ☐ Configuration of complex systems or products
- ☐ Implementing "Best Practices"
- ☐ Capturing corporate knowledge before it is lost
- ☐ "Smart" Questionnaires
- ☐ Web site Concierge - just like this system you are running
- ☐ Making CRM Smart
- ☐ Taking Business Rules to the next level
- ☐ Building Expert Systems
- ☐ I'm not sure yet

WHAT IS W.I.N.K?
This system will build a custom web page for you of information on Exsys based on your interests, industry and application area.
Exsys Inc has been used for over 25 years to solve an incredibly wide range of decision-making problems.
Please answer a few questions and the material most relevant to you will be selected and displayed.

Web site concierge using Flash
W.I.N.K (What I Need to Know)® system

This works by having the Flash SWF file send data to the Corvid Servlet Runtime using POST. The Corvid Inference Engine processes the data and sends back XML data to the SWF file. The XML data may contain additional questions to ask, conclusions to display, etc.

The information sent to the SWF file is controlled by the Corvid developer, so information on any variables can be sent if it is needed. The SWF file either reconfigures the display according to the XML data, or can even load another SWF file and pass the data to it. This process can be repeated as often as needed to run the system.



PROS:
A really great place for scuba. Many outstanding places to snorkel. Many very fine restaurants. An island with great birding.

CONS:
None

83 official dive sites with 100-150' visibility! The whole island is a protected Marine Park keeping it untouched and unspoiled. Plenty of off-shore 24/7 variety and short boat trips for divers and snorkelers. The island has 190 species of birds including flamingos, and the protected Lora, with good birding at Washington/Slagbaai National Park. Along with bonefishing, anglers can go after sailfish, marlin, tuna and tarpon, and the island hosts a number of local and International fishing tournaments. Bonaire's 50 plus restaurants have taken high honors at international culinary competitions two years in a row. Lac Bay is popular with wind surfers with a 90% chance of favorable winds.

Flash system using sliders, pull-downs, mouse overs and dynamically built system results

Using the Adobe Flash interface requires knowledge of building Flash systems and Flash Action Script. Sample Action Script files and the code necessary for the Flash/Corvid interface is provided. Outside of that code connecting the Flash SWF to the Corvid Servlet Runtime, the rest of the Flash program can be anything that can be done in Flash, and can use any other Flash interfaces. Flash based interfaces allow the most complex end user interaction.

Running Standalone (off-line) as a Java Executable

hp
Applet Frame
11:13
DV Camcorder Selection Expert
What type of video do you plan to take:
Family and vacation
Documenting items for insurance or identification
Sports
Scenery
Animals and birds
Making my own movies
Business - recording meetings
Documentary
iPAQ

While there are many reasons to deliver systems online, some systems need to be run standalone. This can be due to a lack of web connection, a need to run on some mobile devices, commercial based or subscription applications or a desire to download and run certain systems anonymously. Corvid makes this easy to do. Any Corvid system using the Corvid Applet Runtime can be run as a standalone system by calling the Java Applet Runtime JAR file as a Java executable. The system will run in a window identical to the applet window when running online. The same user interface designs, images and options used for the Applet Runtime apply. When running standalone, some of the Java security restrictions are reduced, so the system can read and write files to the local machine and execute other local programs. This can open up additional functionality.

Systems can be run standalone on any computer that has a Java Virtual Machine. These are available for a wide range of platforms including essentially all PCs and some small handhelds such as the HP iPAQ.

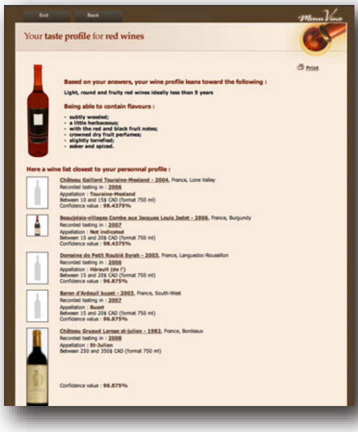
Running Embedded Under Another Program that Provides the End User Interface

Corvid systems can also be run tightly integrated with another program that provides the user interface, or an interface with instrumentation that drives a system. Data can be passed to process or can work in the background to analyze data streams in real time. This can be done with the Applet Runtime locally or using the Corvid Servlet Runtime, which can be sent data items and that will return XML data.

For special situations Corvid has an API and allows new functions to be added to the Inference Engine in Java. Custom functions or interfaces can be added to Corvid to perform special calculations or to communicate with instrumentation and other hardware. This level of customization can produce very powerful systems and is often used for process control, monitoring and predictive maintenance types of systems.

Displaying Results

There are many ways to display the results of a user session. Many systems simply display the conclusion or advice in the Applet Runtime window. Others can build more complex reports or results pages. These pages can be created in HTML, RTF or PDF formats.



Your taste profile for red wines

Based on your answers, your wine profile leans toward the following :

Light, round and fruity red wines ideally less than 5 years

Being able to contain flavours :

- subtly wooded;
- a little herbaceous;
- with the red and black fruit notes;
- crowned dry fruit perfumes;
- slightly torrefied;
- sober and spiced.

Here a wine list closest to your personnal profile :

Château Gaillard Touraine-Mesland - 2004, France, Loire Valley
Recorded tasting in : 2006
Appellation : Touraine-Mesland
Between 10 and 15$ CAD (format 750 ml)
Confidence value : 98.4375%

Beaujolais-villages Combe aux Jacques Louis Jadot - 2006, France, Burgundy
Recorded tasting in : 2007
Appellation : Not indicated
Between 15 and 20$ CAD (format 750 ml)
Confidence value : 98.4375%

Domaine de Petit Roubié Syrah - 2003, France, Languedoc-Roussillon
Recorded tasting in : 2006
Appellation : Hérault (de l')
Between 15 and 20$ CAD (format 750 ml)
Confidence value : 96.875%

Baron d'Ardeuil buzet - 2003, France, South-West
Recorded tasting in : 2007
Appellation : Buzet
Between 15 and 20$ CAD (format 750 ml)
Confidence value : 96.875%

Château Gruaud Larose st-julien - 1982, France, Bordeaux
Recorded tasting in : 2008
Appellation : St-Julien
Between 250 and 350$ CAD (format 750 ml)

Confidence value : 96.875%

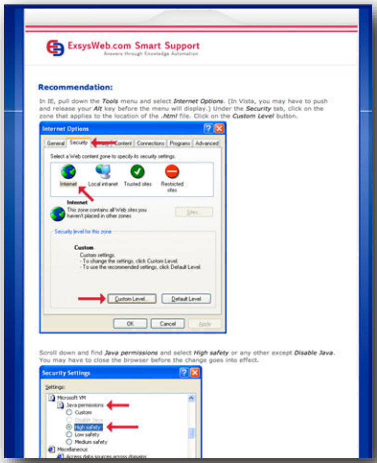
Results page displaying recommendations and why

Corvid's Collection variables are designed for dynamically creating reports. Various rules in the system can contribute to the overall content,. Template files can be used which have the overall format and look-and-feel for the results page, and which contain special commands to embed the detailed content and advice for the particular session. This approach makes it easy to build graphically interesting pages, which only require rules that add specific items of text.

Corvid's WINK (What I Need to Know)® concept can be used to dynamically build web pages based on the end users interests or requirements.

Results pages can be saved in a way that allows them to be bookmarked for later reference or shared with others.

Systems can be designed to generate editable documents in RTF format that automatically open in MS Word. This is very useful for systems that generate complex documents, which the end user may wish to edit or add to.



Recommendation:

In IE, pull down the *Tools* menu and select *Internet Options*. (In Vista, you may have to push and release your *Alt* key before the menu will display.) Under the *Security* tab, click on the zone that applies to the location of the *.html* file. Click on the *Custom Level* button.

Select a Web content zone to specify its security settings.

Internet
This zone contains all Web sites you haven't placed in other zones

Security level for this zone

Custom
Custom settings.
- To change the settings, click Custom Level.
- To use the recommended settings, click Default Level.

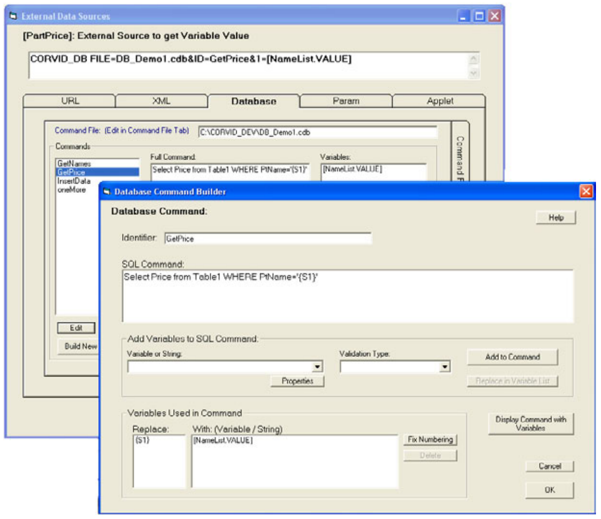
Scroll down and find *Java permissions* and select *High safety* or any other except *Disable Java*. You may have to close the browser before the change goes into effect.

Automated tech support system showing dynamically generated results, which displays visual step-by-step instructions

For non-editable documents, they can be created as PDF format documents that open in Adobe Acrobat and can be saved like any other PDF file.

Databases and Other External Program Access

Corvid systems can be integrated with the existing IT infrastructure. This allows connecting the expert system to databases, XML files, other data files and other programs.



[PartPrice]: External Source to get Variable Value

CORVID_DB FILE=DB_Demo1.cdb&ID=GetPrice&1=[NameList.VALUE]

Command File: (Edit in Command File Tab) C:\CORVID_DEV\DB_Demo1.cdb

Commands: GetNames, GetPrice, InsertData, oneMore

Full Command: Select Price from Table1 WHERE PtName='{S1}'

Variables: [NameList.VALUE]

Database Command:

Identifier: GetPrice

SQL Command:
Select Price from Table1 WHERE PtName='{S1}'

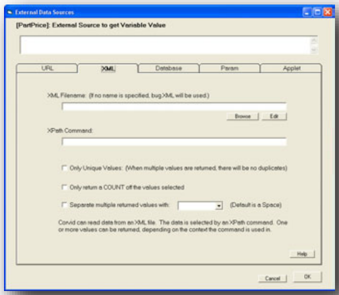
Add Variables to SQL Command:
Variable or String:
Validation Type:

Variables Used in Command

Replace:	With: (Variable / String)
{S1}	[NameList.VALUE]

Corvid can read and write data from a database using standard SQL commands. Any ODBC/JDBC complaint database can be used, which includes virtually all standard database programs. When data needed by the systems is available in the database, just add an SQL command to read it and Corvid will get the data when needed. The system status can be saved to the database either to document the session, or to build systems that allow the end user to break off mid-session and return later.

XML files can be read using standard XPath commands. As with databases, just associate a XML file and XPath command with a Corvid variable and it will automatically get the data when needed. Other programs or server resources addressed as a URL can be called as needed and they can return data to Corvid.



[PartPrice]: External Source to get Variable Value

XML Filename: (If no name is specified, bug.XML will be used.)

XPath Command:

Only Unique Values: (When multiple values are returned, there will be no duplicates)

Only return a COUNT off the values selected

Separate multiple returned values with: (Default is a Space)

Corvid can read data from an XML file. The data is selected by an XPath command. One or more values can be returned, depending on the context the command is used in.

Where Exsys Corvid Knowledge Automation Expert Systems are Most Effective

To best implement expert knowledge in advisory systems, the problem-solving logic should:

- Be well understood and documented
- Be based on logical steps, procedures or business rules
- Not involve intuition, guesses, or be based on arbitrary "personal taste" decisions

The Exsys approach and techniques of knowledge capture are extremely effective for a broad range of project areas. Since 1983, our customers have successfully implemented thousands of interactive decision-making systems, which are in use worldwide, and they're realizing significant efficiency gains and return on their investment.

Successful Knowledge Automation Expert System Categories

Problem-Solving Diagnostics – When experts identify malfunctions or interpret complex data, they quickly look for symptoms indicative of particular problems. The knowledge of how to handle these problems is ideal for conversion into a knowledge automation expert system. The knowledge of the best experts can be made widely available over the Web. By making this knowledge accessible by employees or customers that need it, such systems reduce downtime, increase productivity, and free up experts to handle more complex problems and projects. These systems are also beneficial in capturing the problem-solving expertise of seasoned experts that may be retiring or changing jobs.

Customer/Product Support – These problems are in most ways similar to diagnostic problems, but often go beyond the diagnosis of a malfunction and include following a precise sequence or specific procedures. A knowledge automation expert system is the ideal front-end to a more traditional Help Desk. The system can handle the first (and often higher) levels of interaction with the user. In many cases, the system can solve the problem, without using support staff resources. If the problem is unusual and not part of the system, Corvid can interface to a traditional help desk, opening a case ticket and passing on information about what has already been done to fix the problem. Knowledge automation systems interfaced with traditional help desks bring less experienced staff up to speed quickly without repeated training or interruptions, and provide a more personalized and effective online support environment.

Regulatory Compliance/Business Rules – Though regulations can be very complex, regulations are generally documented in a form very similar to the IF/THEN form of business rules. Corvid regulatory systems insure that all relevant regulations are considered and all policies followed consistently. The logic behind a decision can be automatically documented. These systems provide significant cost savings by helping companies stay within industry compliance, protect employees, and avoid potential fines and possible bad publicity. Many government agencies use Exsys Corvid on their Web sites to provide regulatory information to the public.

Website Concierge – Large Web sites have many pages of information tailored for specific visitors and interests. Finding the desired information is increasingly difficult, even in well-structured sites. An alternative to complex linking strategies is a Corvid W.I.N.K. (What I Need to Know) ® system. It provides an "expert consultation" with the visitor to determine what parts of the site will be relevant to them. Corvid then dynamically builds a customized page with just the information they need, and links to the appropriate parts of the site.

Unlike some other database techniques that allow limited "customization", with Corvid the full power of the Inference Engine is used to determine what to display. The result is a unique and very pleasant experience for visitors.

Decision Support – Business Intelligence and advice is becoming a strategic part of many companies' assets and competitive advantage. Many types of decisions and recommendations can be converted into knowledge automation expert systems.

They provide business users with the ability to access, analyze, and share information stored in a company's databases, manufacturing systems and other data sources, in ways not practical with other technologies. Exsys Corvid systems make the analytical knowledge of the best experts available to all. Systems can be accessed on-demand, or run in the background to monitor for developing situations. Effectively disseminating and providing access to expert systems greatly increases a firm's potential in many ways, and is an ideal way to disseminate specialized skills with minimal training.

Smart Questionnaires – One of most common uses of knowledge automation expert systems is to make questionnaires more intelligent. The logic of an expert system leads to asking only pertinent questions, which have been determined to be relevant due to previous answers the user has provided. No irrelevant questions are asked, and no important issues are overlooked. Several questions can be asked on a single page and the system can "save state" and be accessed when convenient. This produces a far superior user "consultation". Collected data can be sent to other programs, or stored in databases.

Product Selection / Recommendation – Selecting which products best meet a customer's needs and requirements can be a very complex process. With Corvid this can be expressed in logical rules relating to customer needs and product specifications. Unlike case-based or "learning" approaches, expert systems handle conflicting requirements by using expert knowledge to weight the factors that are most important for a particular customer or situation. They also always give a recommendation of the best fit, even when all customer desires cannot be met. A website is a "virtual salesperson" - it should not say: "we have nothing that matches all of your needs" - instead it should recommend the best options, just as a human salesperson would.

Configuration – Configuring complex systems with many pieces usually requires an expert with many years of experience. They must determine which pieces are required, but avoid any incompatibilities. This is an ideal problem for the precise and meticulous analysis of an expert system, which can guarantee valid and complete configurations.

Modularity allows an Exsys Corvid system to be built in small, manageable modules that automatically combine to provide the high-level logic needed to solve the problem. The system can also create a report explaining the basis for its recommendation. External factors such as inventory and customer deadlines can also be incorporated into the system.

Data Analysis – There are many types of problems, which require a combination of numeric analysis and non-numeric data to be analyzed logically. Knowledge automation expert systems are an ideal way to handle such problems. Purely numeric analysis can be handled internally, or it can be interfaced to other external programs.

The addition of powerful rule-based logic analysis makes it possible to handle non-numeric relationships that are often cumbersome in traditional numerical analysis software. Persons interacting with the systems don't require the extensive knowledge of interpreting the data. The expert system analysis process can even be completely embedded within other systems to appear invisible to end-users.

Inconsistency Detection – Knowledge automation expert systems can monitor customer transactions or events against policies and procedures to detect inconsistencies that may indicate problems like fraud or other irregularities. The ability of the expert systems to handle complex logic allows such systems to be much "smarter", both in detection and in recognition of illegitimate activities that might not trigger other systems.

Manufacturing / Process Control – Knowledge automation expert systems have a long, proven history of use in controlling processes to detect and correct problems before they become serious. From DuPont to Eastman Chemical, many of the major industrial and manufacturing companies rely on expert systems. They run invisibly in the background to analyze various data tags throughout a process and alert operators of potential problems.

Exsys Corvid System Requirements

- Microsoft Windows 7, Vista, 2000, 2003, XP
- Microsoft Internet Explorer ver. 5 or higher
- 150 MB Free Disk Space
- Minimum Screen Resolution: 1024 x 768 with standard fonts or 1152 x 864 with large fonts

Conclusion

Ideally, people would have immediate contact with human experts in every area of specialty that they might need, 24 hours a day. But this can't happen, and many decisions can't wait for or afford access to an expert advisor.

Most organizations have ways to disseminate data. Expert systems developed and deployed with Exsys Corvid directly deliver knowledge - "know-how", advice, and recommendations, rather than just information. This enables people to solve complex decision-making problems without having to learn the underlying logic or needing specialized training. This is the power that Exsys Corvid Knowledge Automation provides – direct delivery of knowledge to the people that need it, when they need it.

Once Exsys knowledge automation expert systems are built and deployed, the problem-solving skills they contain are available to everyone that needs them. Staff with minimal training can immediately perform at a much higher level. On the Web, anyone can access the systems, and employees that are online can also run the same systems stand-alone. Their easy to use interfaces work well on wireless devices including PDAs, laptops, kiosks and appliances.

Online knowledge automation expert systems provide traffic-building interaction with your customers, promote customer loyalty and confidence in selecting products, and reduce the load on support staff, and keep a variety of processes running smoothly. Exsys Corvid knowledge automation expert systems provide a very effective and efficient way to provide people - prospects, customers, employees and even advisors themselves, with a way to have 'round-the-clock access to top-level expert decision-making knowledge and consistent advice.

Contact an Exsys representative today to discuss your ideas, find out about our many customer case studies, plan application strategies, and discover the best approaches for efficient and effective expert system projects.

EXSYS®
Capture Knowledge
Deliver Answers

EXSYS, Inc.
6301 Indian School Rd. NE
Suite 700
Albuquerque, NM 87110 U.S.A.
Tel: +1.505.888.9494
Fax: +1.505.888.9509
info@exsys.com
www.exsys.com