Neota | Logic

Technology WHITE PAPER

What We Do

Neota Logic builds software with which the knowledge of experts can be delivered in an operationally useful form—as applications embedded in business systems or consulted interactively in a browser.

Neota Logic transforms expertise into answers and action—in law, compliance, risk management, accounting, human resources, environmental regulation, medicine and other fields.

How We Do It

Our technology, the Neota Logic System, solves problems in many fields just as Microsoft Excel solves financial and numerical problems—without programmers, quickly and efficiently.

The Neota Logic System combines individually powerful reasoning methods—decision trees, decision tables, if/then rules, weighted factors, spreadsheets, calculations, case-based reasoning and others—into a single expert engine that can manage problems of great complexity and subtlety for thousands of users and millions of transactions.

How Our Applications Work

A typical Neota Logic application, when run interactively in a browser, will:

1. **Ask questions** of the user to collect facts—The interview is intelligent and efficient, driving the selection and sequence of questions by the logic of the application, the answers given to earlier questions and the intermediate conclusions already reached.

2. **Collect data** from other sources—Based on the user's answers to questions, the application may collect data from other sources, such as databases, web services and other Neota Logic applications.

3. **Apply reasoning** to the facts and data—As it collects facts and data, the application will continuously and automatically apply all the types of reasoning necessary to replicate the reasoning of the domain expert.

4. **Reach conclusions** based on the reasoning—By applying reasoning to facts and data, the application, like a human expert, reaches intermediate and final conclusions and can explain why it reached those conclusions.

5. **Execute actions** based on the conclusions—When the application reaches its conclusions, it can perform a variety of actions, including:

   - Display an on-screen summary of results.

   - Create a detailed, customized report or memo in Word or HTML to be e-mailed or printed.

   - Send an e-mail to the user or any other person.

   - Update a database or send data to another application.

   - Send facts and conclusions to a document assembly program to generate transactional documents.

   - Trigger a step in a workflow.

Embedded Applications

When invoked from other software, such as a transactional system or a workflow engine, a Neota Logic application need not necessarily interact with a user on screen. The application can collect necessary facts and data from databases and other sources, then apply reasoning, reach conclusions and execute actions just as it does when run interactively. A single application can function in either or both modes simultaneously.

Data Analysis

As an application is used, questions asked and answers given are written to a database, creating a systematic record of decisions, exercises of discretion and processes that have hitherto been informal, undocumented and difficult to analyze, review and manage.

Data analysis and visualization tools are applied to inform management and counsel about patterns of activity and potential issues or risks.

Neota Logic Technology

Neota Logic provides an integrated suite of tools with which to develop, test, maintain and deliver expert applications, which can be embedded in business systems or consulted interactively in a browser, on a computer or smartphone. Neota Logic's technology is unique in these key respects:
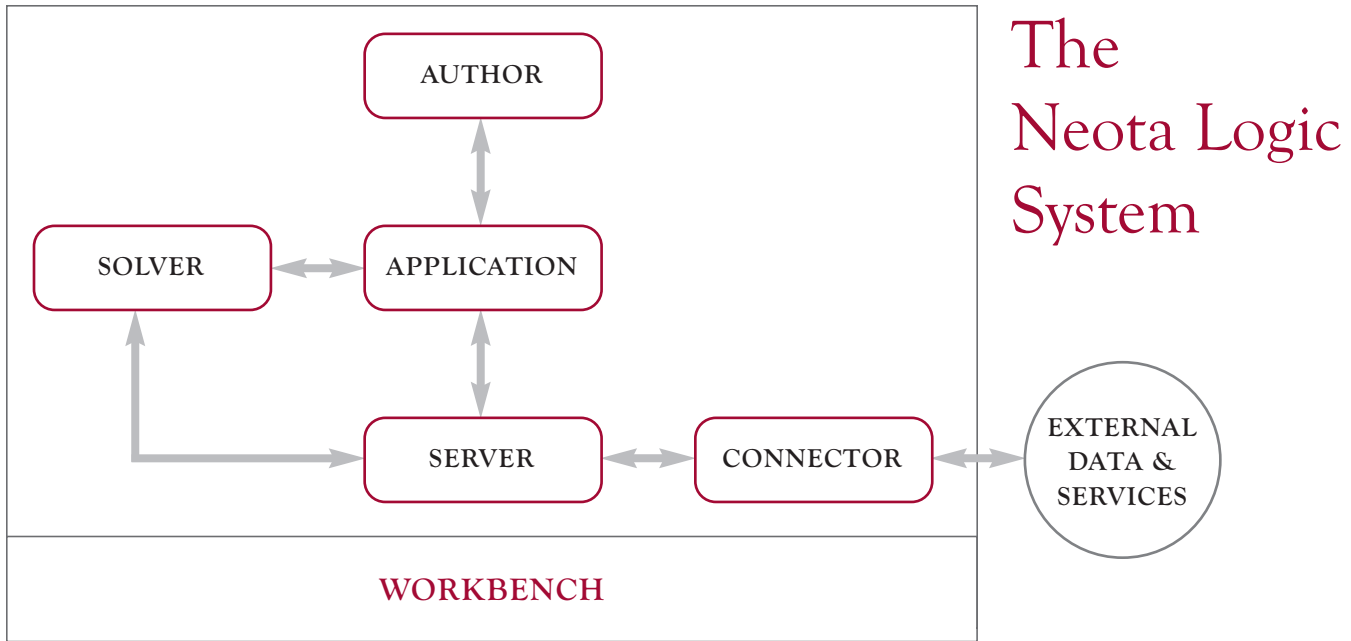
N|L solves very complex problems that cannot feasibly be solved with conventional software tools, including standard programming languages and business rules systems. A Neota Logic application may include hundreds or thousands of intricately inter-related rules that are represented in clear diagrams and plain English, and can easily be verified by domain experts.

N|L combines many different reasoning methods in a single graphical authoring tool. Any or all of the reasoning methods—decision trees, decision tables, if/then rules, weighted factors, spreadsheets, calculations, case-based reasoning and others—can be combined in a single application.

N|L automatically integrates rules and reasoning methods as well as external data queries and actions, under the control of a state-of-the-art inferencing engine, logical support mechanism and prioritization algorithm.

N|L enables people who are not trained software developers to build, maintain and deploy very complex applications. Rules are constructed in graphical editors. Relationships among the rules, including the sequence in which they are applied, are managed automatically by the Neota Logic Server. Dynamic web user interfaces are constructed automatically, and may be styled by authors and designers. Authors and domain experts can focus on the subject matter rather than programming mechanics. IT staff can focus on source databases and business systems with which Neota Logic applications are integrated.

N|L is dramatically efficient. Building applications takes days or weeks rather than months or years. Time-to-market is constrained only by the availability of the relevant rules and domain experts, not by the technology. Application maintenance—which is critical as laws, regulations and business policies change—is simple and quick.

```
        ┌──────────┐
        │  AUTHOR  │
        └──────────┘
             ↕
┌────────┐  ┌─────────────┐
│ SOLVER │↔ │ APPLICATION │
└────────┘  └─────────────┘
     ↑           ↕
     └──→ ┌──────────┐   ┌───────────┐      EXTERNAL
          │  SERVER  │ ↔ │ CONNECTOR │ ↔    DATA &
          └──────────┘   └───────────┘      SERVICES

              WORKBENCH
```

The Neota Logic System

The Neota Logic System includes:

Workbench
The portal to Neota Logic applications, software and services.

Applications
A representation of facts, conclusions, data queries, rules and all other aspects of experts' reasoning on a topic.

Author
A visual rapid application development (RAD) tool with which business professionals build and maintain expert applications.

Server
The reasoning engine that runs applications to gather data and return answers.

Connector
Tools for connecting applications to databases, web services and custom software.

Solver
A visual tool with which applications are analyzed and tested.

Neota Logic Workbench

Applications are managed from the Workbench. Like a portal or document management system, Workbench provides control of users and groups, permissions and versions. Users can run applications for which they have permissions. Authors can edit, version and test applications, manually and with Solver. Domain experts can test applications and review AutoDocs, the documentation automatically generated by Author.

Neota Logic Author

Applications are created in Neota Logic Author, a visual rapid application development (RAD) tool that enables business professionals ("authors") to create very powerful applications without programmers.

Working with domain experts, authors construct an outline of the issues and define the conclusions to be reached and facts to be gathered. Reasoning—the domain expert's approach to analyzing the issues—is then constructed in visual editors for each of the reasoning methods.

Applications are declarative rather than procedural. That is, the domain expert and the author define the application's goals (key conclusions to be reached) and actions (alerts, reports, e-mail, database updates). Neota Logic Server does the rest—automatically determining which facts must be obtained by presenting questions to the user or querying a database and which rules must be applied; setting intermediate as well as final conclusions; and triggering on-screen alerts to the user, reports in HTML or Word, e-mail messages and database updates.

Large problems may be divided into smaller, logical units, which can be built and maintained more easily and re-used in more than one application. For example, rules as to application of a particular statute may be relevant in several contexts. Building those rules in a separate application allows them to be used in each context but maintained in one place.

To validate and verify the application's reasoning and results, Neota Logic Author produces detailed documentation (AutoDocs) in plain language that can be reviewed on paper by subject matter experts. For example, this AutoDocs excerpt shows a decision tree in a financial application:

FROM Canada TO product qualified	IF applicable law = Canada
	1. IF product type = bullion option (cash-settled) OR bullion option (physically-settled) OR total return swap OR weather index transaction
	THEN product qualified = Highly Likely
	2. IF product type = margin loan
	THEN product qualified = No
	3. IF product type = basis swap OR bond option OR cap transaction (interest rate) OR cap transaction (commodity price) OR collar transaction (interest rate) OR interest rate swap OR swap option (cash-settled)
	THEN product qualified = Yes

Neota Logic Author also includes testing, analysis and debugging tools that display graphically all aspects of an application both before and while it is being used.

Neota | Logic

Neota Logic Server

Applications created in Neota Logic Author are executed by Neota Logic Server, which contains the reasoning engines for the many, integrated, hybrid reasoning methods supported by the Neota Logic System. The capabilities available in Server include:

▶ **Situations**—Combinations of facts and conclusions, including complex Boolean sets, are constructed as Situations, either automatically from the values of selected inputs or individually one at a time.

▶ **Mappings**—If /Then rules are created visually by drag-and-drop. Mappings may include facts, conclusions and situations.

▶ **Decision Trees**—Complex rules may be created in tree form, also by drag-and-drop. Trees are sometimes the clearest view of the domain expert's reasoning about an aspect of a problem.

▶ **Decision Tables**—Multiple rules of the same pattern can be created very quickly in spreadsheet form.

▶ **Optimization**—Conclusions may be set to Optimize, e.g., set Default Risk to Optimize-to-Maximum, which causes an application to seek the highest (or lowest) value possible for that conclusion using all available logic paths. This technique, unique to Neota Logic System, enables extraordinarily compact representation of very complex rules, and therefore very rapid construction and easy maintenance of such rules.

▶ **Multi-value** facts and conclusions—Conclusions may have multiple values, e.g., Required Compliance Steps, each of which is determined by different logic paths. Logic Server will automatically traverse all relevant logic paths to be sure that all possible values are determined.

▶ **Instances**—Applications can apply reasoning to tabular data structures, included nested structures, of the sort typically defined in a relational database. For example, a financial services product may have several aspects (type, duration, risk and governing contract); the contract may also have several aspects (name, date). An application might apply rules based on product type, contract and duration to assess the risk for every product in the database.

▶ **Weighted Factor Matrixes**—Multiple factors that may affect a conclusion are identified. Each factor is weighted, using one of several weighting schemes, and scored on a scale, as defined by the domain expert. Factor weights and scores are combined mathematically to set a result conclusion, which may in turn be used in other reasoning. Weighted Factor Matrixes are particularly suited to the balancing tests often used in legal analysis.

▶ **Spreadsheet & Mathematical Models**—An application can use formulas in a Microsoft Excel spreadsheet or call upon Mathematica for exceptionally complex calculations.

▶ **Calculations**—With the Calculations Editor, authors can specify many different types of operations to be performed on facts and conclusions, including arithmetic, date and text operations of the kind available in Microsoft Excel.

▶ **Case-Based Reasoning (CBR)**—A set of input facts is matched to the characteristics of existing cases using a nearest-neighbor best fit algorithm. Cases with the closest match to the input facts are retrieved and can be displayed and used in other reasoning processes. CBR is particularly suited to diagnostic and customer assistance problems, where new patterns or exceptions may be captured as new cases.

▶ **User-Defined Functions**—If none of the built-in functions solves a problem at hand, a custom function can be written (with a Neota Logic Editor or any programming language) and added to the Logic Server.

▶ **Custom Logic Engines** for specific problems—For example, a text analytics engine to classify users' natural language answers to questions in accordance with a taxonomy could be integrated via the Logic Server's extensible architecture.
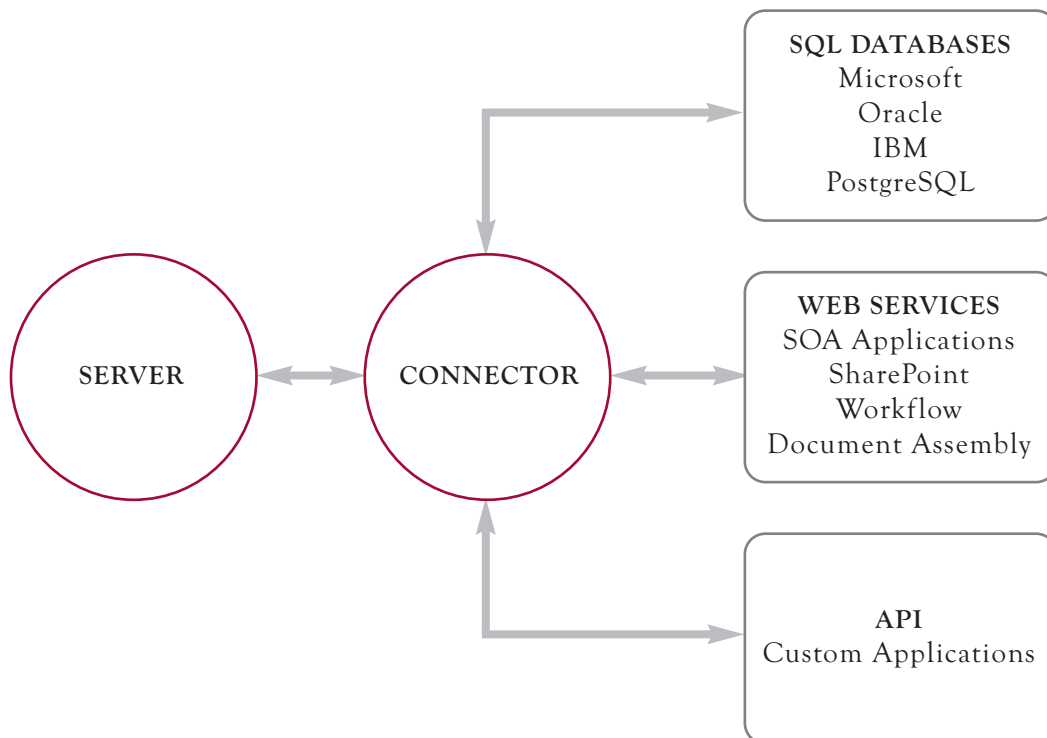
Neota Logic Server (continued)

All reasoning methods are automatically integrated and prioritized. In contrast to business rules systems, it is not necessary to construct explicit ruleflows to determine the sequence in which rules are executed. This unique capability of the Neota Logic System enables rapid creation and easy maintenance of very sophisticated, subtle and complex applications that replicate the thinking of the best experts in a domain.

Neota Logic applications can run interactively—presenting questions and returning answers to a user in a web browser on a computer or smartphone—or can run invisibly as part of other business applications and processes such as transaction auditing.

Neota Logic Server can be configured in a cluster of 10 or 100 or 1,000 virtual servers. It is thus scalable to handle thousands of users and millions of inquiries. In the Amazon Web Services cloud, which provides virtual servers on demand for as long or short a time as needed, Neota Logic Server can scale up (and down) to handle very large peak loads for transaction audit and other intensive assignments.

Neota Logic Connector

Neota Logic applications can integrate with—for example, read data from and write answers to—relational databases such as Microsoft SQL Server, Oracle and IBM DB2; non-relational data sources such as Microsoft SharePoint; web services via REST and SOAP; document assembly engines such as Brightleaf, DealBuilder, Exari and HotDocs; workflow systems such as Bonitasoft, K2 Blackpearl and SharePoint; and custom software of almost any kind via an Application Programming Interface (API).

SQL DATABASES
Microsoft
Oracle
IBM
PostgreSQL

SERVER

CONNECTOR

WEB SERVICES
SOA Applications
SharePoint
Workflow
Document Assembly

API
Custom Applications

Neota Logic Solver

Complex applications may have thousands or millions of possible input facts and outcomes. Even simple applications will have hundreds. To validate applications by manual testing is impractical. Neota Logic Solver enables automated testing, as well as other kinds of analysis.

Domain experts and authors use Solver to verify that an application is complete and accurate. Authors building applications run regression tests to verify that updates to an application or to Neota Logic software have not introduced errors. Administrators run performance stress tests to assure that service level agreements will be met.

Solver What If explores fact patterns incrementally and systematically. A test scenario is defined by selecting useful values for relevant facts—all possible values of all input facts, or only subsets or ranges of specific input facts. Solver What If computes all the fact patterns in the scenario, sends each pattern to the application, and then records the results in a database or a file that can be opened in Excel. Solver What If can also generate tables of common or important fact combinations, which may be convenient to users as quick reference guides to the application.

Solver Monte Carlo simulates real-world use of an application based on probability. In a test scenario, a probability distribution curve is specified for each input fact. Solver Monte Carlo then runs many sessions varying inputs in accordance with the distributions. Domain experts can examine the data from the sessions to identify recurring fact patterns, average or expected values of key conclusions and variance of key conclusion values (e.g., best case and worst case).

Contact

For information, please contact Michael Mills, Chief Executive Officer, mills@neotalogic.com, (646) 402-6214.

Neota | Logic

60 East 42nd Street
Suite 1810
New York, NY 10165
neotalogic.com