Black Pearl Knowledge Broker

User Guide

Version 1.1

Contents

List of Figures

Chapter 6

Defining Rules • 113

Chapter 7

Defining a Predictive Model • 143

Chapter 8

Example: Defining an Ontology • 179

Chapter 9

Example: Defining Rules • 197

Chapter 10

Example: Defining a Predictive Model • 205

Appendix A

Navigating the Black Pearl Knowledge Broker Interfaces • 223

Introduction

Welcome to the *Black Pearl Knowledge Broker User Guide,* which provides information for understanding and using the Black Pearl™ Knowledge Broker™. The Black Pearl Knowledge Broker is an all-Java, net market platform that applies business rules to data captured from multiple sources—existing enterprise databases and the Internet—to create real-time recommendations. These recommendations take into account buyer/seller goals, preferences, behaviors, and market context. The result is more accurate and timely recommendations that both increase the likelihood a transaction will occur, and enhance long-term customer satisfaction. With the Black Pearl Knowledge Broker you can:

- Use point-to-point, asynchronous communications to access and process disparate data located in JDBC-compliant data sources and XML files, including Web, relational database, and flat files
- Define and apply business rules that are relevant to your industry or company
- Employ complex analytic models and optimizing algorithms to identify information patterns and develop what-if scenarios
- Generate real-time recommendations that are easily accessible through any standard Web browser or Java application

What's New in Version 1.1

New features in version 1.1 include the following:

Improved Rules. The Black Pearl Knowledge Broker now supports the combination of rules using the logical AND operator.

Concept Comparisons. The new Concept Comparison feature enables you to compare properties from different concepts in rules.

EJB Application Server Integration. By deploying the Knowledge Broker within a J2EE 1.2-compliant EJB application server, you gain access to many enterprise, messaging, and database access functions while improving the Knowledge Broker's interoperability across heterogeneous networks.

Configure JDBC Datasource Drivers During Installation. You can now connect to JDBC-compliant datasources using an easy-to-use installation panel.

AdvisorBean API. The new AdvisorBean API lets you interface with the Black Pearl Knowledge Broker. See the Black Pearl Knowledge Broker online documentation in HTML and JavaDoc format for this information.

How This Guide Is Organized

Part 1	**Foundations**
Chapter 1	*Transforming Data into Knowledge* provides an introduction to the human process of transforming data into knowledge.
Chapter 2	*Developing a Knowledge Base* provides an introduction to using the Black Pearl Knowledge Broker to transform data into knowledge.
Part II	Getting Started
Chapter 3	*Installing the Black Pearl Knowledge Broker* details how to download, install, configure, and uninstall the Black Pearl Knowledge Broker.
Chapter 4	*Using the Black Pearl Knowledge Broker* describes basic procedures for using the Black Pearl Knowledge Broker.
Part III	Procedures
Chapter 5	*Defining an Ontology* provides step-by-step procedures for defining the datasources, concepts, and contexts that form an ontology.
Chapter 6	*Defining Rules* provides step-by-step procedures for defining business rules.
Chapter 7	*Defining a Predictive Model* provides step-by-step procedures for defining a predictive model.
Part IV	Tutorial
Chapter 8	*Use Case: Defining an Ontology* provides sample procedures for defining an ontology for an investment brokerage.
Chapter 9	*Use Case: Defining Rules* provides sample procedures for defining rules for an investment brokerage.
Chapter 10	*Use Case: Defining a Predictive Model* provides sample procedures for defining a predictive model for an investment brokerage.
Part V	Appendices
Appendix A	*Navigating the Black Pearl Knowledge Broker Interfaces* provides an overview of the application's User Interfaces (UIs), menus, toolbars, and command buttons.
Appendix B	*Error Messages* provides a quick reference to all Black Pearl Knowledge Broker messages.
Glossary	*Glossary* provides terms and definitions.

Audience

This guide is intended for people who will use the Black Pearl Knowledge Broker to perform a variety of tasks:

User	Tasks	Important Chapters
System Administrators	Download, install, and monitor the Black Pearl Knowledge Broker	• Chapter 3, • Appendices A and B
Application Administrators	Build a knowledge management system	• Chapters 1, 2, 4, 5, and 8; • Appendices A and B
Business Managers	Define rules and predictive models	• Chapters 1, 2, 5, 6, 9, 10; • Appendices A and B
End-users	View real-time recommendations	• Chapters 1 and 2

Document Conventions

This guide uses a variety of formats to identify different types of information.

Convention	Function
`courier`	Identifies syntax statements, on-screen computer text, and path, file, drive, directory, database, and table names.
`<courier>`	Identifies variable names.
`bold courier`	Identifies text you must type.
italics	Identifies document and chapter titles, special words or phrases used for the first time, and words of emphasis.
underline	Identifies URLs, domain names, and email addresses.
Initial Caps	Identifies Window, menu, command, button, option, tab, keyboard, and product-specific names.
ALL CAPS	Identifies acronyms and abbreviations.
[ ]	Identifies an optional item in syntax statements.
{ }	Identifies an optional item that can be repeated as necessary within a syntax statement.
>	Identifies a separation between a menu and an option.
|	Identifies a separation between items in a list of unique keywords when you may only specify one keyword.

Special Message Conventions

Identifies information that will help prevent equipment failure or loss of data.

Identifies information of importance or special interest, including Notes and Tips.

Menu Conventions

This guide uses the Menu > Option convention. For example, "Click Format > Style" is a shorthand instruction for "Click the Format menu, then select the Style option."

Additional Help

For additional information or advice, contact:

Contact Information

Application	Online Help and Readme text file
Phone	(415) 357-8300
Facsimile	(415) 357-8399
Internet	http://www.blackpearl.com
Email	sales@blackpearl.com techsupport@blackpearl.com techpubs@blackpearl.com
Postal	Black Pearl, Inc. 400 Second Street, Suite 450 San Francisco, CA 94107

Part

I

Foundations

The Black Pearl Knowledge Broker is an all-Java, net market platform that transforms data and information into knowledge. It lets you intelligently apply business rules and predictive models to data captured from multiple sources—existing enterprise databases and the Internet—to generate real-time recommendations. These recommendations, which incorporate buyer/seller goals, preferences, behaviors, and market context, both increase the likelihood a transaction will occur, and enhance long-term customer satisfaction.

But what does this mean? What are data, information, and knowledge? What are business rules, predictive models, and real-time recommendations?

Part I: Foundations answers those questions and provides a foundation for understanding the Black Pearl Knowledge Broker.

Chapter 1

Transforming Data Into Knowledge

For nearly 3,000 years, questions about human learning and knowledge have been a source of study, discourse, and debate for writers, philosophers, and theologians. Within the last century, social and cognitive scientists, neuro-physiologists, physicists, computer scientists, and engineers became interested in the subject.

It is not within the scope of this guide to present all the discourse on human learning and knowledge. Instead, this chapter will focus on how humans transform data into knowledge, viewed from a perspective relevant to understanding the Black Pearl Knowledge Broker.

This chapter is deliberately constructed in a linear fashion. Each section builds upon the ideas presented in the previous sections. It is recommended that you read this chapter in sequential order, and before reading Chapter 2.

What is a Data Element?

A *data element* is a word, number, measurement, or fact that represents or refers to an object. That object can be any person, place, thing, thought, idea, or event that exists (or is believed to exist) in space and/or time. A data element is only a pointer to the object and not the object itself, and as such, is without any intrinsic or inherent meaning.

People, however, naturally apply meaning to data elements. For example, to most English-speaking people, *May* is the name for a month in a calendar year. It comes after the month of April, and before the month of June. It has 31 days. In the Northern Hemisphere, May is the last full month in the spring season. May is also the name for a woman. And, if the word is written as *may,* then it is a verb denoting the right to do something. In other words, the data element *May* has meaning because we understand it in relationship (*context)* to the data elements *name*, *month*, *calendar*, *year*, and *action*.

To keep track of and organize these relationships, we develop an abstract image (*concept)* about the who, what, when, and where of those relationships. Thus, *May* is a data element when viewed as a discrete, stand-alone object. It is a concept when viewed as being in relationship with other data elements or concepts.

Concepts, however, develop over time and as a result of our dynamic interactions with the world and other people. Our experiences—and the learning that is a consequence of those experiences—help us to contextualize and conceptualize the world. For example, a child encounters a bright, flickering object. Someone says the word "candle." This scene is repeated several times until the day arrives when the child sees the object and says "candle." The child experiences something that exists in space and time (the bright, flickering object), relates the object to the sound "candle" (another object that exists in space and time), creates a mental image of the connection between the objects, and learns the concept "candle." This process is repeated hundreds of times each day, with many different objects, until the child develops an extensive vocabulary of concepts and contexts that describe the world.

It is these concepts and contexts that enable the child to function within the world, because that world now has meaning—as defined by those concepts and contexts. These concepts and contexts, which are built upon data elements, are the foundation for information.

What is Information?

Information is the association of concepts and contexts whose meanings are understood. For example, "May has 31 days" is information if you or another person understands that:

- The data element "May" is transformed into the concept "May" when understood in relationship with name, months, calendar, and year.
- The data element "31" is transformed into the concept "31" when understood in relationship with numbers.
- The data element "days" is transformed into the concept "days" when understood in relationship with measurements of time.

Information, which can be communicated through spoken and written words, pictures, symbols, movements, touch, scents, and other methods, enables you to learn, create more and more complex concepts, contexts, and information, and develop critical thinking. For example, you learn that the bright object is a candle when you decide one day to touch the brightness. Instead of obtaining the object of desire, you scream out in pain. Your mother or father says "no, hot." You have no idea what "hot" means, but do know you don't like the feeling in your fingers! You associate the sound "hot" with the unpleasant sensation in your fingers and conceptualize that "Candles are hot. No!" What you've accomplished is to relate two discrete concepts (candle, hot) with an experience (pain in your fingers) to create a new set of information (don't touch candles because they are hot).

This process of using information to learn, create ever more complex concepts, contexts, and information, and develop critical thinking, is the basis for knowledge.

What is Knowledge?

Knowledge is four interrelated things:

- The understanding of patterns of information that consistently and completely repeat themselves over time (*pattern knowledge*). For example, you are *pattern knowledgeable* if you understand that May is the fifth month of the calendar year, always has 31 days, your mother's birthday is on May 23, she lives in Florida, and she always expects you to join her for a birthday dinner.
- The recognition of the implications of those patterns on future behavior (*implication knowledge*). For example, you are *implication knowledgeable* if, on April 15, you understand that there are exactly thirty-eight days before your mother's birthday, you live in California, and you must arrange to fly to Florida for her birthday dinner.
- The capacity to make plans and recommendations based on those implications (*strategy knowledge*). For example, you are *strategy knowledgeable* if you understand that you must purchase an airline ticket by April 23 if you want to take advantage of the 30-day advance fare rates.
- The ability to act on those plans (*action knowledge*). For example, you are *action knowledgeable* if you actually purchase the ticket by April 23.

Each level of knowledge is gained through learning, and learning can only take place through interaction, whether that interaction is:

- Sensorial (learned through seeing, touching, tasting, smelling, or hearing something)
- Experimental (learned by testing a hypothesis to determine whether or not something is true)
- Informational (learned through conversation, reading, or social/educational norms about something)
- Mental (learned through internal, abstract conceptualizations about something)

Daily interactions afford you with opportunities to learn about interconnected webs of patterns, implications, strategies, and actions. If you learn to recognize and understand the patterns, you are pattern knowledgeable. If you learn to recognize and understand the implications, you are both pattern and implication knowledgeable. And so on. The more levels of knowledge you gain, the deeper your understanding. You gain more levels of knowledge, and the you are better able to orient yourself in a particular environment, overcome difficulties, make decisions, and solve problems.

But knowledge is not a static, one-time event. It is both dynamic and iterative. It is *dynamic* if it makes use of existing data elements, concepts, contexts, information, and knowledge to create additional knowledge. For example, if you know that all mammals breathe, that humans are mammals, and that John is a human, then you can say, quite knowledgeably, that John breathes, even though no one explicitly told you that John

breathes. It is *iterative* if it repeats the learning process for a particular concept, context, or information in order to increase your understanding. For example, you may first learn the concept "red" by seeing something red and being told "red." Next, you may learn that red is a "color." Later, you may learn that the color red is a "primary color." Still later, you may learn that the primary color red is a part of the "visible spectrum of light." In each case, you are deepening your understanding of the concept "red" through an iterative process.

The dynamic and iterative nature of knowledge is based on two doctrines:

- Association of ideas
- Compositionality

The *association of ideas* is the ability to unite in some manner two or more concepts, based on resemblance and/or contiguity. *Resemblance* states that when two concepts are considered similar, whatever properties associated with one concept are automatically associated with the other concept. Thus, the statement "John looks like a human, therefore he must breathe since all humans are mammals and all mammals breathe" is an example of the association of ideas through resemblance. *Contiguity* states that when concepts are frequently experienced together, they are always mentally associated together. Thus, the statement "Red is a primary color that is a part of the visible spectrum of light" is an example of the association of ideas through contiguity.

Compositionality is the ability to infer knowledge from the meanings, patterns, and relationships of individual concepts or information, based on a system of rules. *Rules* are logical statements that describe how an object (person, place, thing, or event) will behave in a particular situation. There are three basic types of rules:

- *Legal rules,* which are mandated by a federal, state, city, or other warrantable agency (including SEC, OSHA, FCC, EPA, and unions), cannot be broken unless the issuing agency amends the rule. For example, an SEC rule mandates a "quiet period" before a company's initial public offering.
- *Domain-specific rules,* which are defined by a specific business or field of knowledge (including physics, chemistry, and law), can also not be broken unless amended by the issuing domain. For example, a brokerage house may stipulate that stockbrokers employed with the firm for less than six months may not offer speculative stock advice.
- *Common sense rules,* which include opinions, hunches, and best practices, can be broken if circumstances warrant. For example, a brokerage house may institute a "best practices" policy whereby speculative stocks will not be recommended to a person who is 70 years old and living on a fixed income.

Thus, "You touch the candle flame with your fingers. Your fingers are burned. Do not touch a burning candle," is an example of knowledge inferred through compositionality. It is based on two types of rules: domain-specific and common-sense. The domain-specific rules states that fire is hot and will burn other objects. The common-sense rule states that if you touch a candle flame, you will burn your fingers.

Knowledge gained, whether through association of ideas or compositionality, is implicit or explicit. *Implicit knowledge* is a set of personal experiences, know-how, and mental images that is rarely (if ever) communicated with others. *Explicit knowledge* is a set of codified experience and know-how that is communicated through formal language, including data, concepts, contexts, information, and rules.

The collection of implicit and/or explicit knowledge is a *knowledge base,* which is the foundation for logical predictions and actions.

Chapter 2

Developing a Knowledge Base

Whereas a human knowledge base can include both implicit and explicit knowledge, a computerized knowledge base must be explicit. The Black Pearl Knowledge Broker enables you to create an explicit knowledge base that includes information about datasources, concepts, contexts, and rules. It then uses that knowledge base to discern patterns in your raw data, understand the implications of those patterns, make real-time recommendations, and dynamically and iteratively refine and deepen the knowledge base.

This chapter provides a conceptual overview of the Black Pearl Knowledge Broker objects and processes that enable you to define and refine a knowledge base, including:

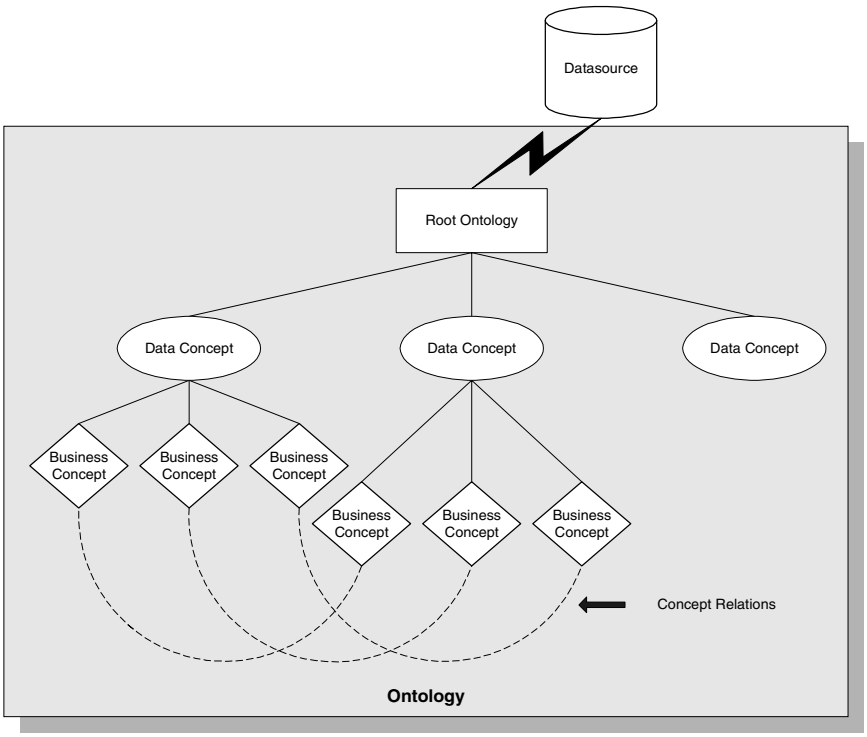
- Ontologies
- Rules
- Knowledge Discovery

What is an Ontology?

An *ontology* is a collection of concepts and contexts that provides a common vocabulary used for defining rules, analyzing information patterns, recognizing implications, and making real-time recommendations. The Black Pearl Knowledge Broker ontology consists of four components:

- *Datasource Connections,* which are URLs, strings, or pathnames to disparate datasources—Web, relational database, and flat files—containing data pertinent to your business
- *Data Concepts*, which are the vocabulary terms that map directly to one or more of your datasources
- *Business Concepts*, which are the vocabulary terms defined by you that are relevant to your industry or company and map to other concepts in the ontology
- *Concept Relations*, which define the connections between concepts

Figure 2-1. Ontology Schema



The concepts and concept relations defined within the ontology are the means for "teaching" the Black Pearl Knowledge Broker about your business—as it is contained within your Web, relational database, and flat file datasources—and communicating with your datasources to discover knowledge and generate real-time recommendations.

You can use the Black Pearl Knowledge Broker Data Concept, Business Concept, and Concept Relation Editors to easily define an ontology linked to your datasources and relevant to your business requirements.

What are Rules?

Rules are logical statements that describe how an object (person, place, thing, or event) will behave in a particular situation. For example, if the air temperature falls below 32 degrees Fahrenheit, then water will freeze. Or, if the account is 30 days past due, then add a late fee.

We learn rules through interactions with people, schools, institutions, and the environment. A computer, however, needs written rules that state the relationship between concepts and specify the response constraints. Like an ontology, these rules guide the computer, and enable it to know what datasources to query, patterns to analyze, implications to recognize, and real-time recommendations to make.

There are three basic types of rules:

- *Legal rules,* which are mandated by a federal, state, city, or other warrantable agency (including SEC, OSHA, FCC, EPA, and unions), cannot be broken unless the issuing agency amends the rule. For example, an SEC rule mandates a "quiet period" before a company's initial public offering.
- *Domain-specific rules,* which are defined by a specific business or field of knowledge (including physics, chemistry, and law), can also not be broken unless amended by the issuing domain. For example, a brokerage house may stipulate that stockbrokers employed with the firm for less than six months may not offer speculative stock advice.
- *Common sense rules,* which include opinions, hunches, and best practices, can be broken if circumstances warrant. For example, a brokerage house may institute a "best practices" policy whereby speculative stocks will not be recommended to a person who is 70 years old and living on a fixed income.

The Black Pearl Knowledge Broker currently supports these subcategories of rules:

- *Constraint Satisfaction*, which enables you to identify the rules, concepts, and data that should be processed by the Black Pearl Knowledge Broker.
- *Priority*, which enables you to specify rules in terms of degrees of importance.

Future releases of the Black Pearl Knowledge Broker will support these additional subcategories of rules: *Sequencing*, which enables you to specify that some rules must be activated before another rule may be processed. *Duration*, which enables you to specify that a rule is valid only for a specified time period. *Control*, which enables you to include both global and local rules.

Each rule is of the basic form:

`if <SUBJECT> then <VERB > to <DIRECT OBJECT>` or, simply, "if X, then perform action Y to Z".

More complex rules are of the form:

```
IF
          <concept>
            such that {<property_value>
                       <operator>
                       <comparison_value>}
          AND
                      {<concept.property>
                       <operator>
                       <concept.property>}
THEN
     <action>
          <concept>
            such that {<property_value>
                       <operator>
                       <comparison_value>}
          AND
                      {<concept.property>
                       <operator>
                       <concept.property>}
```

where `<concept>` is a concept in the ontology

`<property_value>` is a property of the concept, retrieved from the ontology

`<operator>` is one of: `=`, `!=`, `<`, `>`, `<=`, `>=`

`<comparison_value>` is any arbitrary value

`<action>` is currently one of: recommend buy, recommend sell, notify, predict

For example:

```
if customer's age is less than 45 then recommend buy adventure sports
```

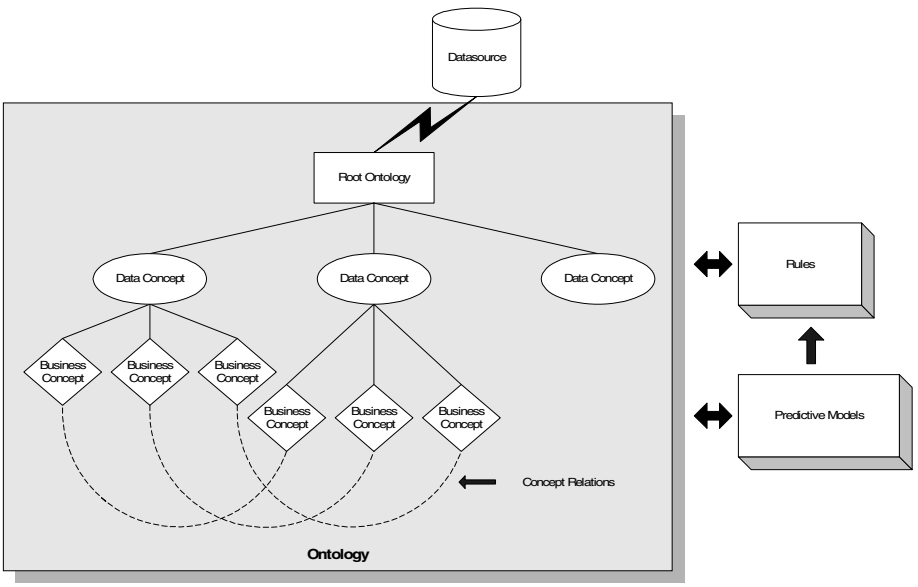
In this example, the *subject* is a concept (customer), the *attribute* is a property (age), and the *attribute value* is a filter parameter (less than 45). The *direct object* is a concept (adventure sports). The *verb* (recommend buy) acts on the direct object, which is the recipient of the action.

You can use the Black Pearl Knowledge Broker Rule Editor to define rules that describe two things:

- How your business typically responds to individual situations, and
- What seemingly similar situations will require different responses.

These rules are the means for "teaching" the computer how to respond to a variety of situations relevant to your business requirements.

Figure 2-2. Ontology and Rule Schema



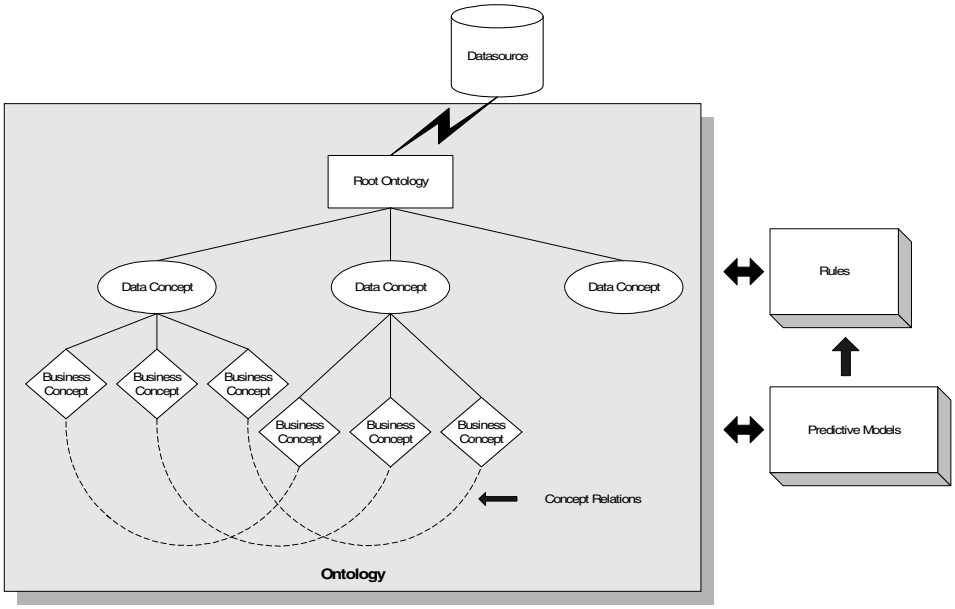
What is Knowledge Discovery?

Knowledge discovery is an iterative process that analyzes historical data to determine patterns, relationships, and trends that can be used to predict future values and real-time recommendations. The process depends on *models*—an abstract conception of a portion of reality—to uncover these patterns and relationships. A *predictive model* is a particular type of knowledge discovery process. It consists of three objects:

- Input values, known as the *independent* or *predictor variables*, that consist of a sample of your raw data.
- An *algorithm* that is applied to the independent variable to discover patterns and relationships.
- An output value, known as the *dependent* or *target variable*, that consists of predicted values for other data. The output value is the foundation for real-time recommendations.

You can use the Black Pearl Knowledge Broker Knowledge Discovery Center to define predictive models.

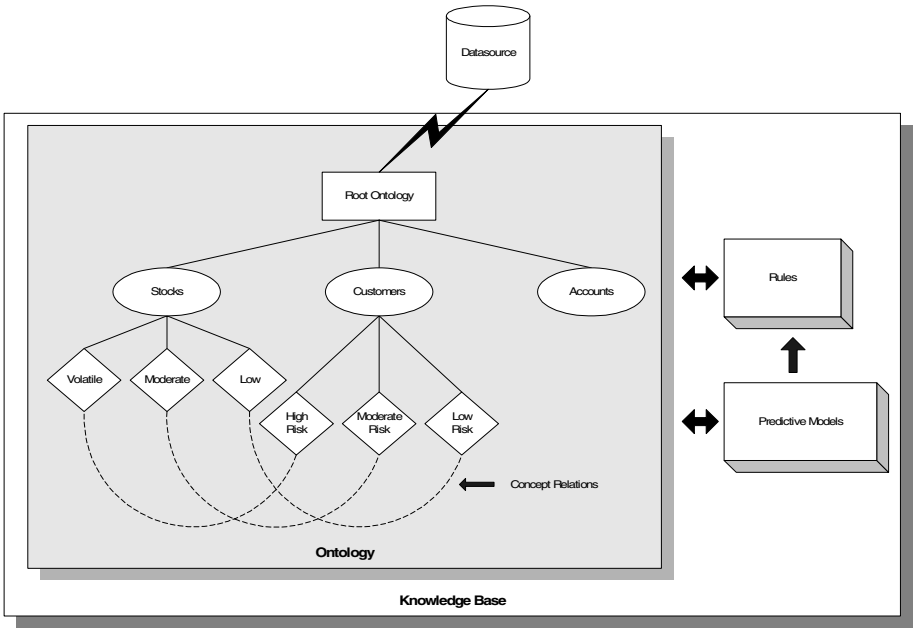
Figure 2-3. Predictive Model Schema



What is a Knowledge Base?

A *knowledge base* is the collection of datasource connections, ontology concepts and concepts relations, rules, and predictive models related to a particular industry or business domain.

Figure 2-4. Knowledge Base



Part

II

Getting Started

The Black Pearl Knowledge Broker an easy-to-use application that includes self-explanatory installation and use procedures. However, a quick introduction to the basic procedures required to install and use the Black Pearl Knowledge Broker will increase your familiarity with the application.

Part II: Getting Started provides you with information for downloading, installing, configuring, and using the Black Pearl Knowledge Broker.

Chapter

3

Installing the Black Pearl Knowledge Broker

The Black Pearl Knowledge Broker can be either downloaded from the Black Pearl website or obtained on a CD-ROM. It is bundled in a Java Edition InstallAnywhere application, and is designed for quick and easy installation.

This chapter provides step-by-step instructions for installing and uninstalling the Black Pearl Knowledge Broker on a standalone, client, or server machine that uses the Windows NT 4.0 operating system. The Knowledge Broker also supports deployment as an Enterprise JavaBean (EJB) within suitable EJB application servers. This chapter also includes information about system and configuration requirements for Windows NT and EJB application server environments.

System Requirements

To use the Black Pearl Knowledge Broker successfully, your Windows NT machine requires the following:

Item	Windows NT Requirements
Operating System	• Server: Windows NT Server 4.0, Service Pack 4 or higher; • Client: Windows NT Workstation 4.0, Service Pack 4 or higher.
CPU	Pentium II-class CPU (or greater)
Disk Space	Minimum 500 MB
Memory	Minimum 64 MB RAM
Database Engine	JDBC 2.0 compliant
Web Browser	Internet Explorer 4.0 (or higher), or Netscape 4.0 (or higher)
Network	TCP/IP

Downloading the Black Pearl Knowledge Broker

The Black Pearl Knowledge Broker resides on a secure directory on the Black Pearl website. Contact your Black Pearl sales representative to obtain a User Name and Password, both of which are required to access the directory.

1. Establish an Internet connection.
2. Type **`http://www.blackpearl.com/download`**.
3. Type the User Name and Password assigned by your Black Pearl sales representative.
4. Double-click the `broker.exe`. The Save As dialog displays.
5. Click Save. The Saving Location dialog displays, indicating that the selected file is being downloaded.
6. Exit the Black Pearl website after downloading the appropriate files.

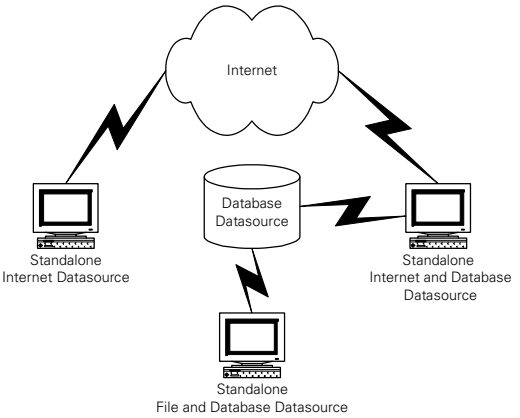
Types of Installations

There are three types of installation: standalone, server, or client.

Standalone Installation

Selecting the *Standalone* option installs both the Server and Client components on a single machine. The Standalone user can access all Datasource Connection, Task Editor, and Knowledge Discovery Center functionality.

Figure 3-1. Standalone Installation



Server Installation

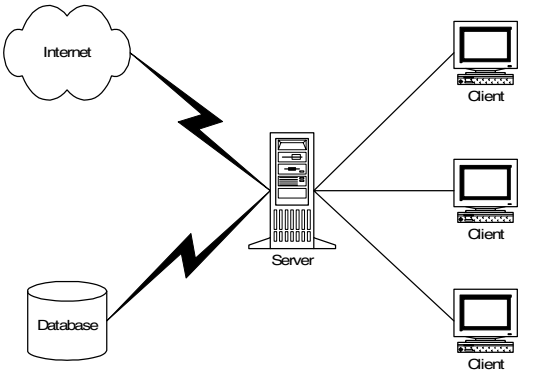
Selecting the *Server* option installs only the Server components. Server users cannot directly access the Task Editor and Knowledge Discovery Center functionality, nor can they directly add, modify, or delete Datasource Connections. This functionality can be accessed through the Client's graphical user interface (GUI). For further information, see *Navigating the Black Pearl Knowledge Broker Interfaces on page 223*.

You must initialize the Router (a component of the Knowledge Broker Server that handles network connections) before launching the Knowledge Broker Server. For further details, see *Initializing the Router on page 58*.

Client Installation

Selecting the *Client* option installs only the Client components. Although Client users can add, modify, and delete datasources, there is an important restriction on this functionality (see *Understanding Client-Server Location Constraints on page 73*).

Figure 3-2. Client/Server Installation



Server Root Directory

The client installation requires the root directory of your Blackpearl server installation. When prompted during the client installation for the server installation location, either choose or type the directory path. You should include only the path to the top-level server installation. For more information on locating the server directory, see the information about identifying the *Server Name on page 48*.

Installation Sequence
Install the server first, followed by the client. For convenience, on NT you can map the server's installation directory to a network directory and note its name.

Installing in Windows NT

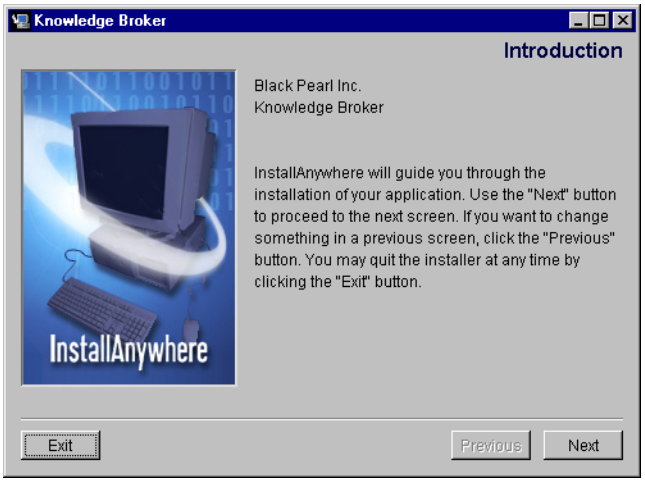
The Black Pearl Knowledge Broker installation process enables you to select from a set of options. After you select the appropriate options, installation proceeds automatically.

1 Complete one of the following options:

   a If you downloaded the Black Pearl Knowledge Broker, open the directory containing the application file and double-click `broker.exe`. The InstallAnywhere application will automatically run and the Knowledge Broker Introduction dialog displays.

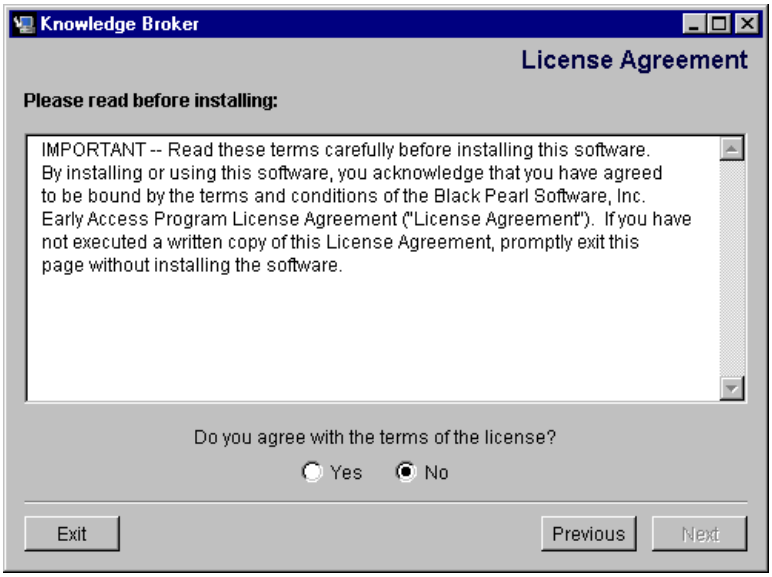
   b If you received the Black Pearl Knowledge Broker on a CD-ROM, insert the CD into the appropriate CD-ROM drive. The InstallAnywhere application will run and the Knowledge Broker Introduction dialog displays.

Figure 3-3. Knowledge Broker Introduction



2 Click Next. The License Agreement dialog displays.

Figure 3-4. License Agreement

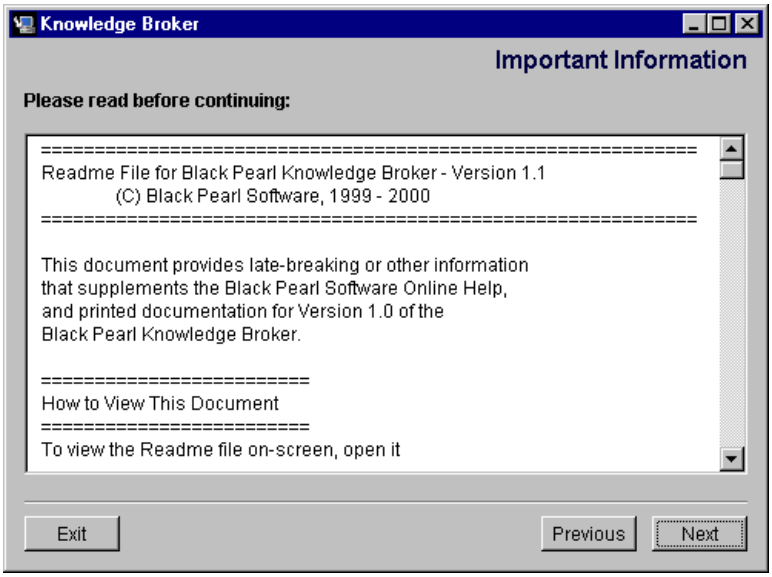


3 Select one of the following options:

- **a** If you do not accept the Black Pearl Knowledge Broker license agreement, click Exit. Click Yes when the Confirm Exit dialog displays.
- **b** If you accept the Black Pearl Knowledge Broker license agreement, select Yes.

4 Click Next. The Important Information dialog displays.

Figure 3-5. Important Information

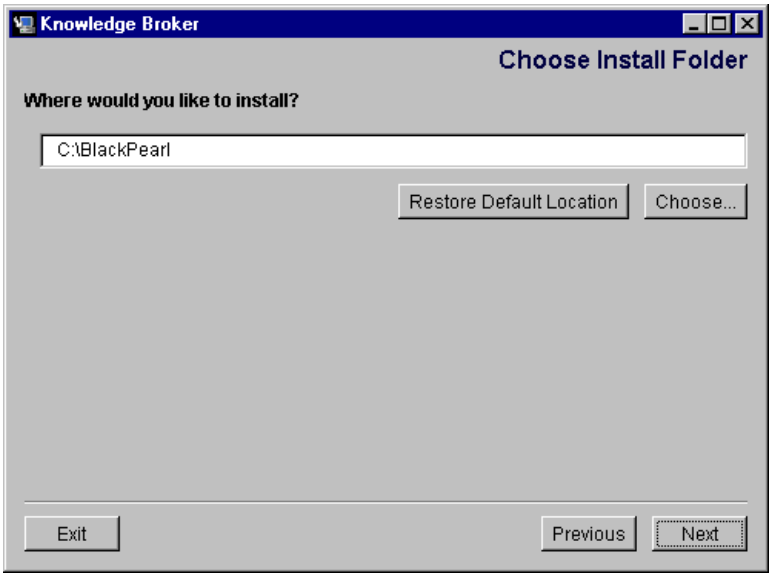


5 (Optional) Scroll down the Important Information dialog to review the Readme file prior to installation of the Black Pearl Knowledge Broker.

After installing the Black Pearl Knowledge Broker, you may access the Readme file by opening file in a standard text editor such as Notepad.

6 Click Next. The Choose Install Folder dialog displays.

Figure 3-6. Choose Install Folder



7 Complete one of the following options:

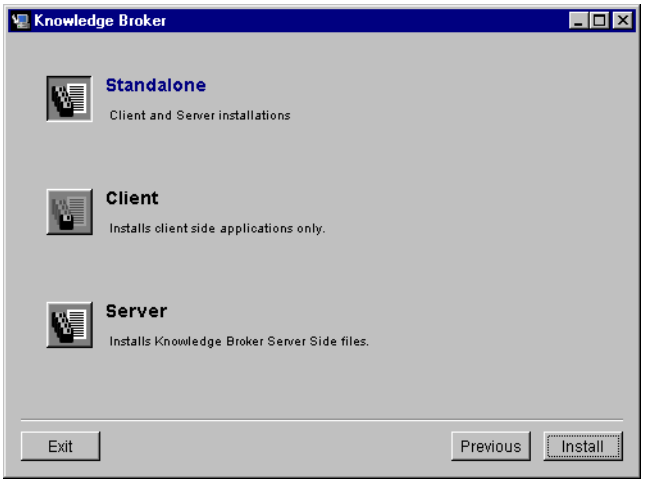
a Accept the default installation settings. By default, all the Black Pearl Knowledge Broker files install to the `C:\BlackPearl\` folder.

b Type the exact path to designate a destination folder. By default, all the Black Pearl Knowledge Broker files install to this folder.

c Click Choose and use the file selection dialog to select a drive and folder. By default, all the Black Pearl Knowledge Broker files install to this folder.

8 Click Next. The Installation Type dialog displays.

Figure 3-7. Installation Type



9 Complete one of the following installation options:

See *Types of Installations on page 41* for more information about your installation choices.

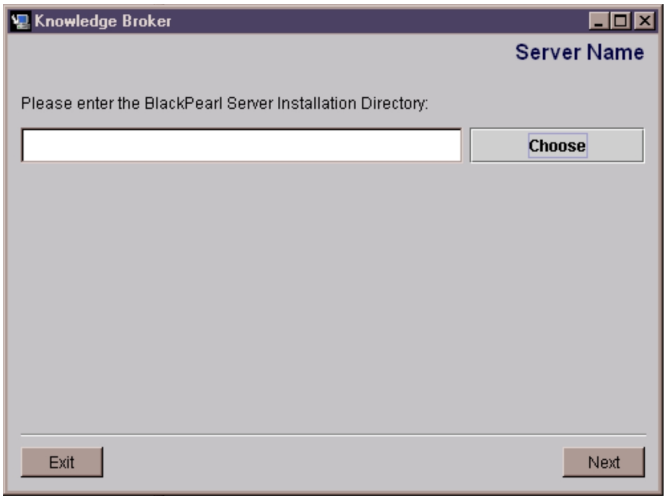
a To install both the Server and Client components on a single machine, select Standalone and click Install. The Black Pearl Knowledge Broker installs to your standalone machine.

b To install only the Server components, select Server and click Install. The Black Pearl Knowledge Broker installs to your server machine.

There is an important functionality constraint to consider concerning the location of Client and Server installations. For non-problematic Datasource Connection functionality, you should install a Client and Server version on the same machine and use that Client to manipulate Datasource Connections. See *Understanding Client-Server Location Constraints on page 73* for further details.

c To install only the Client components, select Client and click Install. The Black Pearl Knowledge Broker installs to your client machine. You must now tell the Black Pearl Client installation process where to find the Server installation components. The Server Name dialog displays:

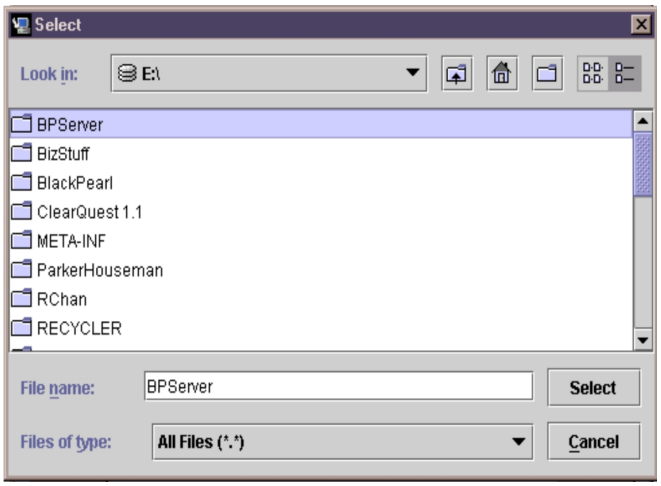
Figure 3-8. Server Name



i Type the name of the Server directory where the Black Pearl Knowledge Broker is installed.

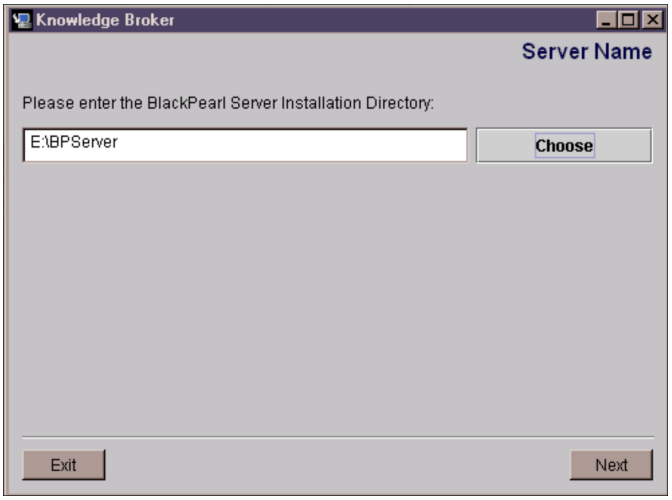
ii Alternatively, click Choose to browse to the location of the Server directory (see Figure 3-9).

Figure 3-9. Locating the Server Installation Directory



iii Click Next. The Black Pearl Knowledge Broker completes the client component installation.

Figure 3-10. Server Name Completed

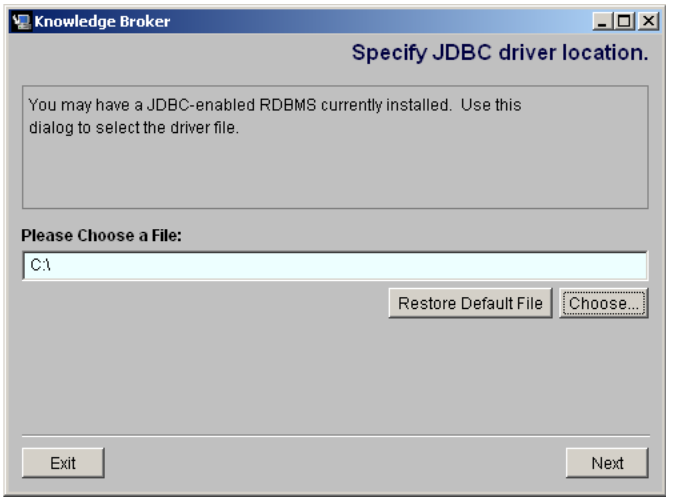


10 You can connect to one or more JDBC-compliant datasources (for more information, see *Configuring JDBC Data Source Driver Locations on page 53*). Note that within

Windows NT, you may instead use the Microsoft ODBC bridge mechanism (see *Configuring ODBC Data Source Drivers on page 52*) and click Next to skip this section.

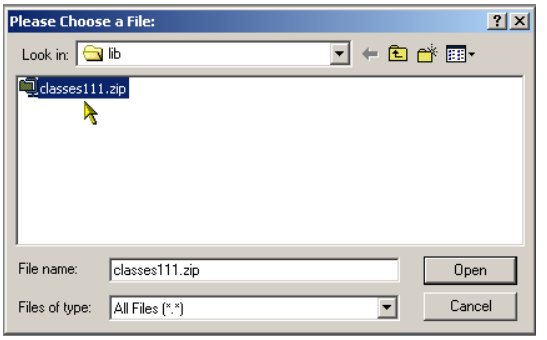
The Specify JDBC Driver Location dialog displays, allowing you select a suitable JDBC driver.

Figure 3-11. Specify JDBC Driver Location



11 You can enter the file location, or click Choose to use the standard Windows file browser dialog to identify the file that contains a suitable JDBC driver:

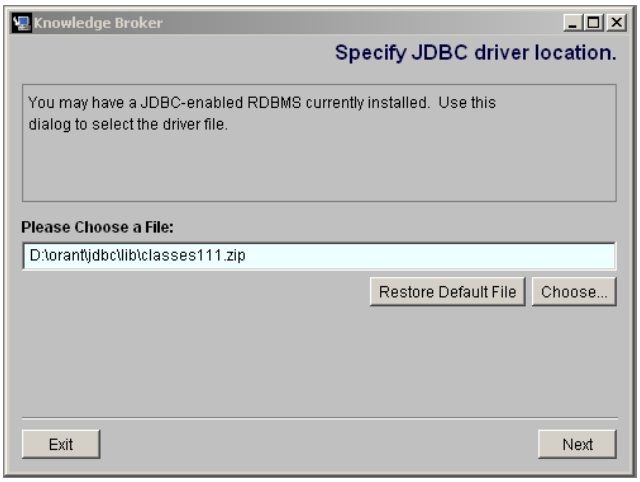
Figure 3-12. Specify JDBC Driver Location



The name of the `.jar` or `.zip` file required depends on which database you wish to access. Your network administrator or DBA (Database Administrator) can help you locate suitable files that contain JDBC drivers.

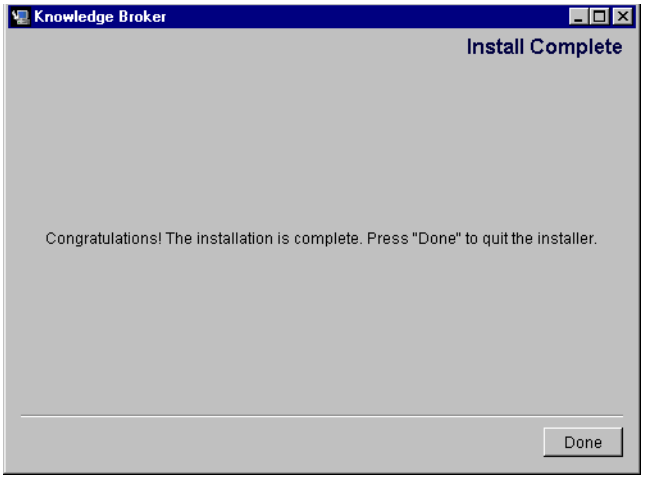
12 Click Open to select the JDBC driver file. The Specify JDBC Driver Location dialog displays again, this time with the file location entered. Click Next to continue.

Figure 3-13. Installation Complete



13 After all files install, the Installation Complete dialog displays. Click Done.

Figure 3-14. Installation Complete



Configuring ODBC Data Source Drivers

Because it is possible to use the Black Pearl Knowledge Broker with a variety of datasource drivers, you must identify the driver before you can use the Black Pearl Knowledge Broker. The Microsoft ODBC Data Source Administrator provides a convenient single location that aggregates many data source drivers for easy access. To create and specify system-wide ODBC drivers, follow these steps:

1. Select Start **>** Settings **>** Control Panel **>** ODBC Data Sources (Windows NT4) or Start **>** Settings **>** Control Panel **>** Administrative Tools **>** Data Sources (ODBC) (Windows 2000). The ODBC Data Source Administrator displays, listing all currently configured datasource drivers.
2. Click System DSN. The ODBC Data Source Administrator/System DSN dialog displays.
3. Click Add. The Create New Data Source dialog displays.
4. Select the appropriate datasource driver.
5. Click Finish. The ODBC Data Source Administrator dialog re-displays.
6. Click OK. The ODBC Setup dialog displays.
7. Complete each of the Setup fields, including the path to the datasource.

Contact your Database Administrator or read the database documentation for the exact information to enter in the ODBC Setup dialog.

8. Click OK. The System DSN dialog re-displays.
9. Click OK again. The Control Panel re-displays.
10. Exit the Control Panel.

If you change datasource types, you will need to modify the ODBC Setup dialog information. Select Start **>** Settings **>** Control Panel **>** ODBC Data Sources. Click System DSN. Select the driver to be modified. Click Modify. Make the appropriate changes.

Configuring JDBC Data Source Driver Locations

Some database vendors provide alternative database access methods in addition to the Microsoft ODBC mechanism. Commonly, they provide Java drivers that use the JDBC mechanism. These drivers are usually installed during a database client install process and frequently reside in `.jar` (a common Java file format) or `.zip` compressed files. Oracle, for example, provides a JDBC driver within an installed file known as `classes111.zip`.

You can select a suitable file during installation of the Knowledge Broker (see Step 10 of *Installing in Windows NT on page 43*). Alternatively, you can specify or update JDBC driver locations after installation:

1. Navigate to the directory where you have installed the Knowledge Broker.
2. In the `bin` sub-directory, edit the `launch.bat` file using a text editor.
3. This batch command indicates the location of the JDBC driver file:

```
if "%JDBC_Drivers%"    == "" set
JDBC_Drivers=D:\orant\jdbc\lib\classes111.zip
```

4. Change this line to reflect the directory path and name where your `.jar` file resides. For example:

```
if "%JDBC_Drivers%"    == "" set
JDBC_Drivers=C:\MyNewRDBMS\lib\jdbc_access.jar
```

5. Save your changes to `launch.bat`. When restarted, the Knowledge Broker will search for a JDBC driver in the new location.

You can specify multiple file locations where the Knowledge Broker will search for suitable JDBC drivers. You use a semicolon character (`;`) to separate the file locations, similar to the `CLASSPATH` format in Java. For example:

```
if "%JDBC_Drivers%"    == "" set
JDBC_Drivers=C:\OracleClient\classes\Oracle.jar;
D:\BlackPearl\lib\DB2.jar;D:\BlackPearl\lib\Sybase.jar
```

Installing the Knowledge Broker in an EJB Framework

The Knowledge Broker supports installation within application servers that support Enterprise Java Beans (EJB). Application servers encapsulate much of the business logic within enterprise systems by providing a component abstraction layer. They handle database access, transaction coordination, and provide an integrated component framework for distributed applications along with administrative and security functions.

Enterprise JavaBeans (EJB) defines a server-side technology model for developing, integrating, and deploying Java components and applications in a scalable, secure, and transactional fashion. For further information, refer to: http://www.javasoft.com/products/ejb/.

By deploying the Knowledge Broker within an EJB application server, you gain access to many enterprise, messaging, and database access functions while improving the Knowledge Broker's interoperability across heterogeneous networks. The Knowledge Broker currently supports deployment within the BEA WebLogic Server 5.1 environment.

Installing the Knowledge Broker in BEA WebLogic Server

The Knowledge Broker supports installation within the BEA WebLogic Server version 5.1 according to the EJB 1.1 specification. You should refer to the WebLogic Server support pages (http://www.weblogic.com/docs51/) to ensure your system exceeds the minimum hardware and software installation requirements. For version 5.1, these requirements are detailed in Table 3-1:

Table 3-1. BEA WebLogic 5.1 Installation Requirements

Item	Windows NT Requirements
Operating System	Windows NT Server 4.0, Service Pack 4 or higher
CPU	Pentium II-class 200 MHz CPU
Disk Space	Minimum 80 MB
Memory	Minimum 64 MB RAM
WebLogic Server JVM	• MS SDK for Java 4.0 • JDK 1.2.2
Client JVM	• MS SDK for Java 4.0 • JDK 1.2.2

To deploy the Knowledge Broker as an EJB within WebLogic Server EJB, your installation contains a suitable `ejbWL51.jar` file that contains deployment descriptors

and classes in accordance with the EJB 1.1 specification. Deployment is a three-stage process:

1 Edit the `weblogic.properties` using a text editor. Append a `BLACK PEARL Advisor registration` section at the end of the file. Include the following line:

```
weblogic.ejb.deploy = InstallDirectory/lib/ejbWL51.jar
```

2 Edit the WebLogic `setenv.cmd` file to include references to Black Pearl's Java classes. The commands you should include are (modified to suit your own installation configuration):

```
@set BP_HOME=C:\BlackPearl\
@set JDBC_HOME=E:\orant\JDBC\lib\classes111.zip

@rem Must be able to find properties files, and JDBC driver
@rem Standalone install configuration, others use \server or \client
@set BP_CLASSPATH=%BP_HOME%\properties\standalone
@set BP_CLASSPATH=%BP_CLASSPATH%;%JDBC_HOME%

@rem CORE Knowledge Broker JARs
@set BP_CLASSPATH=%BP_CLASSPATH%;%BP_HOME%\lib\ai.jar
@set BP_CLASSPATH=%BP_CLASSPATH%;%BP_HOME%\lib\applib.jar
@set BP_CLASSPATH=%BP_CLASSPATH%;%BP_HOME%\lib\dis.jar
@set BP_CLASSPATH=%BP_CLASSPATH%;%BP_HOME%\lib\dis_message.jar
@set BP_CLASSPATH=%BP_CLASSPATH%;%BP_HOME%\lib\ontology.jar
@set BP_CLASSPATH=%BP_CLASSPATH%;%BP_HOME%\lib\server.jar
@set BP_CLASSPATH=%BP_CLASSPATH%;%BP_HOME%\lib\client.jar
@set BP_CLASSPATH=%BP_CLASSPATH%;%BP_HOME%\lib\ui.jar
@set BP_CLASSPATH=%BP_CLASSPATH%;%BP_HOME%\lib\util.jar

@rem Additional JARS required by Knowledge Broker
@set BP_CLASSPATH=%BP_CLASSPATH%;%BP_HOME%\lib\j2ee.jar
@set BP_CLASSPATH=%BP_CLASSPATH%;%BP_HOME%\lib\fscontext.jar
@set BP_CLASSPATH=%BP_CLASSPATH%;%BP_HOME%\lib\providerutil.jar
@set BP_CLASSPATH=%BP_CLASSPATH%;%BP_HOME%\lib\VisualNumerics.jar

@rem Add the BP CLASSPATH to the Weblogic PATH
@set CLASSPATH=%BP_CLASSPATH%;%CLASSPATH%
```

3 Start the WebLogic server and console. On the left-hand panel of the WebLogic console, use the `Distributed Objects > EJB > Deployed Beans` property inspector to verify that the beans have been registered. The two Knowledge Broker EJBs are `Advisor` and `Accessor`. If these are present, Knowledge Broker has deployed correctly.

The following directory contains example files suitable for modification to suit your local environment:

InstallDirectory/appservers/weblogic

Uninstalling in Windows NT

1. Close all applications.
2. Complete one of the following options:
   a. Select Start > Programs > Black Pearl > Uninstall Knowledge Broker. The Black Pearl Knowledge Broker screen displays, followed by a warning to exit all applications. Installed files are removed from your machine.
   b. Click Start **>** Settings **>** Control Panel **>** Add/Remove Programs.
      i. Select Knowledge Broker.
      ii. Click Uninstall. The Black Pearl Knowledge Broker screen displays, followed by a warning to exit all applications. Installed files are removed from your machine.
3. Click OK when the Uninstall Complete dialog displays.

You may need to manually remove some Black Pearl Knowledge Broker files. The Uninstall Complete dialog will indicate if any Black Pearl Knowledge Broker files remain on your machine.

Chapter

4

Using the Black Pearl Knowledge Broker

Although the Black Pearl Knowledge Broker is designed to be an easy-to-use application, a quick introduction to basic procedures will increase your familiarity with the application.

This chapter provides you with information for basic operations, including launching the application, opening a Task Editor, saving your changes to a knowledge base, closing a Task Editor, and ending a session.

Starting the Black Pearl Knowledge Broker

The Black Pearl Knowledge Broker is installed either in Standalone or Client/Server mode (see *Types of Installations on page 41*). There are slightly different ways to launch the Knowledge Broker, depending on the installed version.

Starting the Standalone Version of the Knowledge Broker

To start the Standalone version of the Knowledge Broker, select the following item from the Start menu:

- Start > Programs > Black Pearl > Knowledge Broker

Alternatively, you may double-click the Black Pearl Knowledge Broker icon on your desktop (if this is present).

Two windows appear. One is a command-line window that displays the status of the Knowledge Broker. The other window displays the graphical Launcher. You can minimize the command-line window.

Starting the Server Version of the Knowledge Broker

Starting the Server version of the Knowledge Broker requires you to initialize the Router once (a network application that handles interactions between the Server and the Client).

Initializing the Router

After completing server installation and before running the Knowledge Broker Server initially, you must start the server using the script file:

```
InstallDirectory/bin/installBPserver.bat
```

You should see a command-line window with output text that confirms the initialization of the Router. Once complete, quit the Knowledge Broker. This completes the Router initialization phase.

Once completed, the Router should not require re-initialization. Use the Start Menu item `Knowledge Broker Server` to launch the Server in future.

Running the Server

Once initialized, to launch the Server, use the Start menu to select:

- Start Programs > Black Pearl > Knowledge Broker Server

Two command-line windows appear. Once window displays the status of the Black Pearl Router, while the other window displays the status of the Black Pearl Server. You can minimize both windows.

Starting the Client Version of the Knowledge Broker

You should have configured the Client during installation to indicate the Server's directory. You must start the Server before starting the Client. When the Server is running, use the Start menu to select the following:

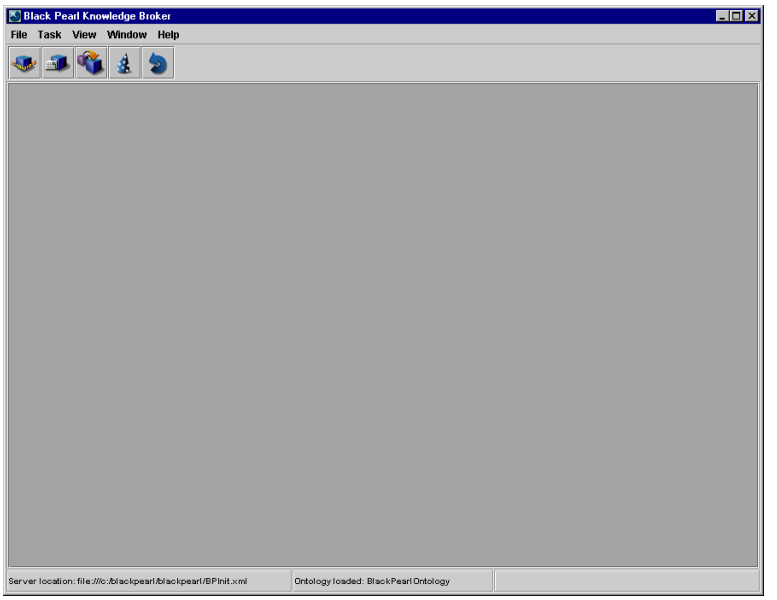
- Start Programs > Black Pearl > Knowledge Broker Client

Two windows appear. One is a command-line window that displays the status of the Knowledge Broker Client. The other window displays the graphical Launcher. You can minimize the command-line window.

Using the Launcher

The Black Pearl Knowledge Broker Launcher enables you to access all application functionality. The Launcher is the graphical display you see when either the Standalone or the Client installations of the Knowledge Broker execute.

Figure 4-1. Black Pearl Knowledge Broker Launcher



The Launcher remains active throughout a *session*, which begins when you start the Black Pearl Knowledge Broker and ends when you exit the Launcher and the associated command-line status windows of either the Standalone or the Client versions of the Knowledge Broker. During a session you may add, modify, delete, or save ontology items, rules, and predictive models.

For detailed information about the Launcher, see *Appendix A, Navigating the Black Pearl Knowledge Broker Interfaces on page 223*.

Customizing the Toolbar

The Launcher enables you to customize the appearance of the Toolbar icons both on the launcher and individual Task Editors or Centers. You may view:

Small icons 

Large icons 

Small icons with text 

Large icons with text 

The default setting is large icons only. To change the appearances of the icons:

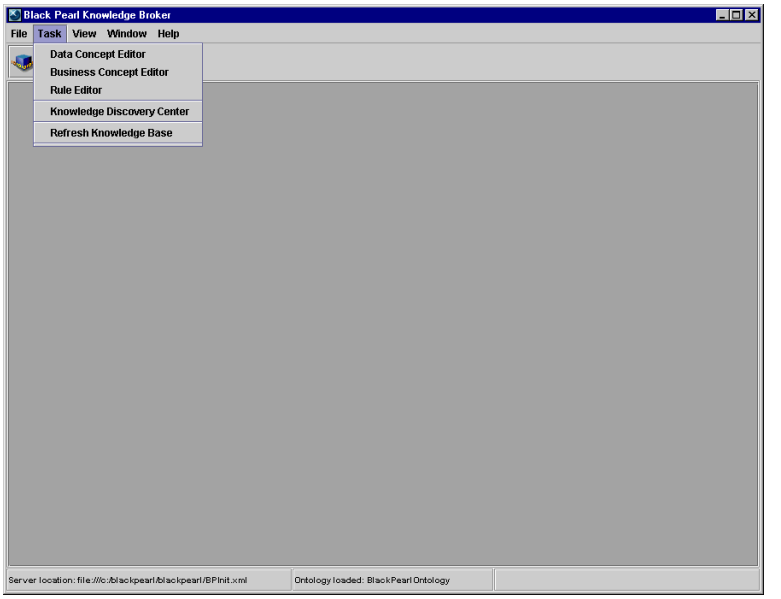
1. Select Launcher **>** View.
2. Complete *one* of the following options:
   - **a** Select Toolbar Labels to view large icons with text.
   - **b** Select Small Toolbar Icons to view small icons only.
   - **c** Select Small Toolbar Icons and Toolbar Labels to view small icons with text.
3. Click anywhere inside the Launcher to close the View menu.

Opening a Task Editor or Center

The Launcher includes menu and toolbar options enabling you to access various Task selections, including Data Concept, Business Concept, Concept Relation, and Rule Editors, and the Knowledge Discovery Center. To access a Task, you may either:

- Click the appropriate toolbar icon. For example, to access the Rule Editor, click the Rule icon.
- Select the appropriate Task item. For example, to access the Rule Editor, select Task **>** Rule Editor.

Figure 4-2. Task Menu

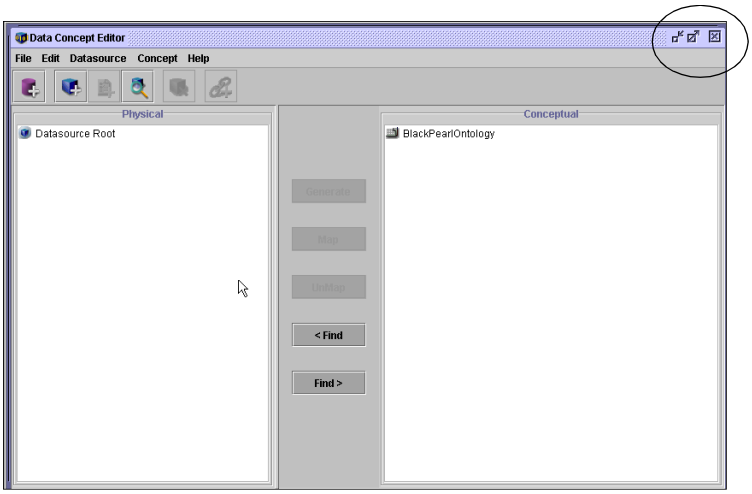


For detailed information about navigating the various Task Editors or Centers, see *Appendix A, Navigating the Black Pearl Knowledge Broker Interfaces on page 223*.

Re-sizing a Task Editor or Center

You can control the size of any Task Editor by using the Size Control buttons.

Figure 4-3. Size Control Buttons



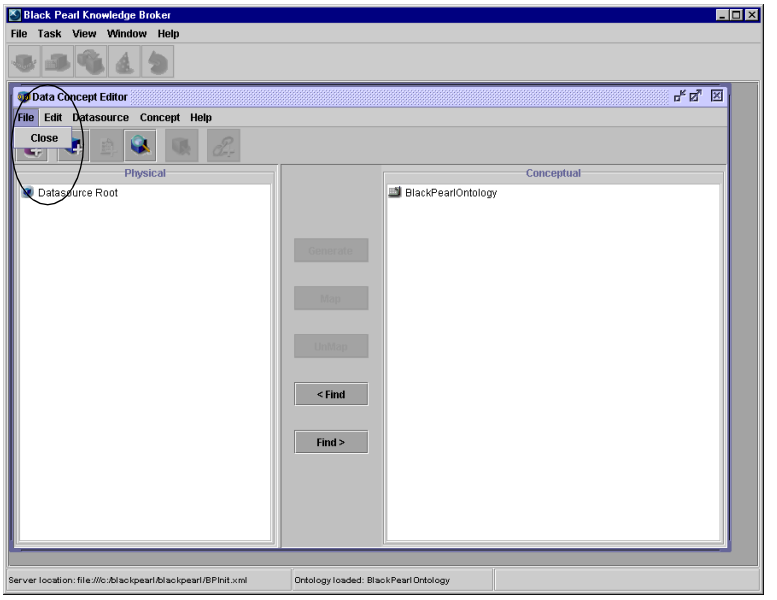
Closing a Task Editor or Center

Only a single Task Editor or Center may be active at any one time, although each Task may have one or more windows open at the same time. To access another Task, close the current Task, then open the new Task.

To close Task Editor or Center, you may either:

- Select File **>** Close in the open Task Editor or Center window.
- Click the Close button in the upper right-hand corner of the open Task Editor or Center.

Figure 4-4. Closing a Task Editor or Center

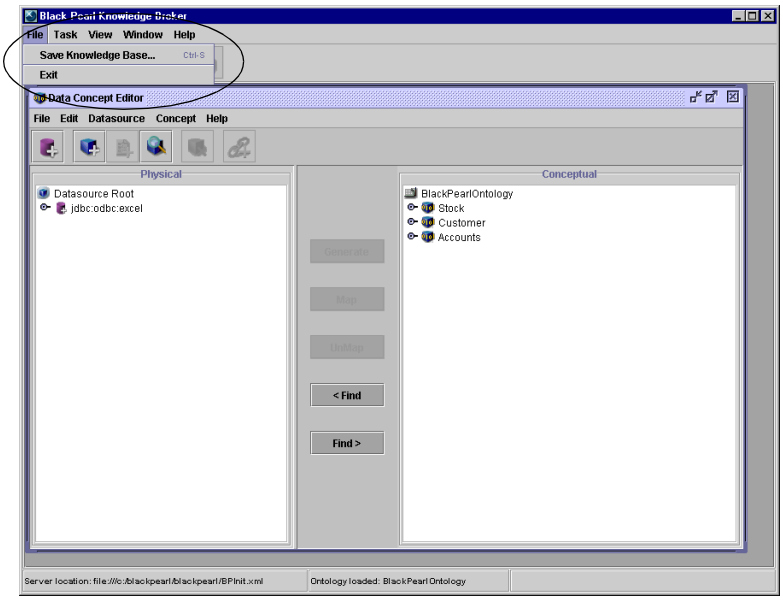


Saving Changes to the Knowledge Base (Single User)

The *knowledge base* is a collection of data, ontologies, rules, actions, and predictive models defined for your business.

To save changes to the knowledge base, you must select File > Save on the Launcher.

Figure 4-5. Saving Changes to the Knowledge Base



Saving Changes to the Knowledge Base (Multi-User)

When multiple Knowledge Broker sessions occur at one time, and a user selects File > Save to Knowledge Base on the Launcher, whichever session chooses Save first will be allowed to save changes to the knowledge base. All other users will be unable to save their changes to the knowledge base.

The system will only allow you to save if your "read" timestamp is greater than the last "write" timestamp. If no other users performed a save before you, you may perform a save. If another user performed a save before you, you must perform a Refresh, which re-reads the latest version of the ontology, and moves your read timestamp forward.

All changes made during sessions subsequent to the Save session will be lost.

Saving Changes to a Rulebase or Knowledge Discovery

The Rule Editor and Knowledge Discovery Center enable you to temporarily save your work and evaluate the effectiveness of the rules and predictive models.

The temporary save is only in effect for the duration of the current session. If you exit the Black Pearl Knowledge Broker, your temporary saves will be deleted unless you save your work to the knowledge base.

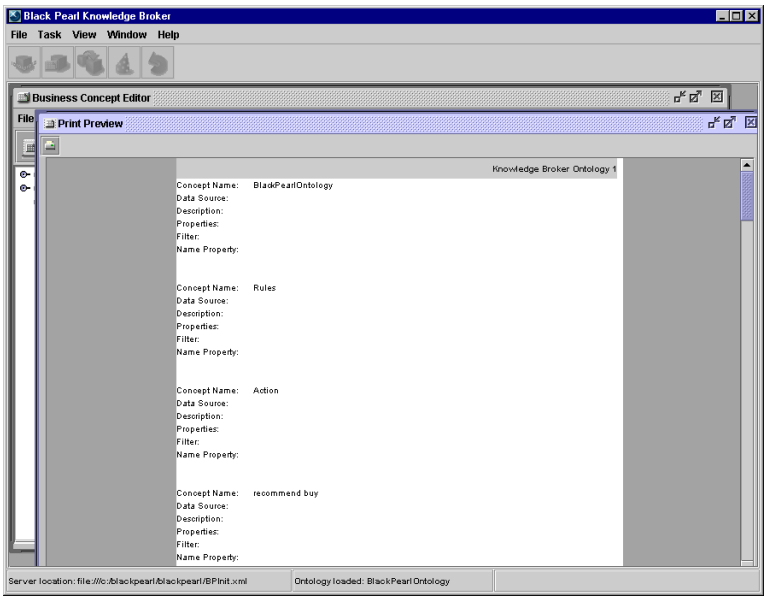
- To save changes to the Rule Editor, select File **>** Save Rule.
- To save changes to the Knowledge Discovery Center, either select File **>** Save, or click the Save button in individual Knowledge Discovery Center windows.

Previewing a Knowledge Base Report

The Print Preview functionality enables you to view all ontology and rule items currently saved to the knowledge base. This functionality is accessible through the Business Concept, Concept Relation, or Rule Editor when you:

- Select File **>** Print Preview.
- Press CTRL+P
- Click the Print Preview icon.

Figure 4-6. Print Preview



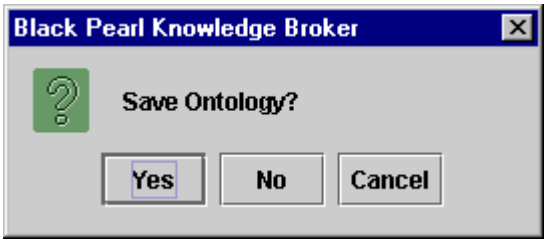
To print a report, click the Print icon in the upper left-hand corner of the Print Preview window.

Ending a Session

A *session* starts when you launch the Black Pearl Knowledge Broker and ends when you exit the application.

1 Select File **>** Exit. The Black Pearl Knowledge Broker dialog displays, asking if you want to save any changes that you made during the session. (If you saved your changes prior to selecting File > Exit, the dialog does not display.)

Figure 4-7. Save Ontology



2 Complete one of the following options:

- **a** Click Yes to save the changes to the knowledge base.
- **b** Click No to end the session without saving the changes.
- **c** Click Cancel to continue the current session.

Part

III

Procedures

Using the Black Pearl Knowledge Broker requires that you define an ontology and rules. Optimal use of the application includes defining a predictive model.

Part III: Procedures provides step-by-step instructions for using the Black Pearl Knowledge Broker to complete these tasks.

Chapter

5

Defining an Ontology

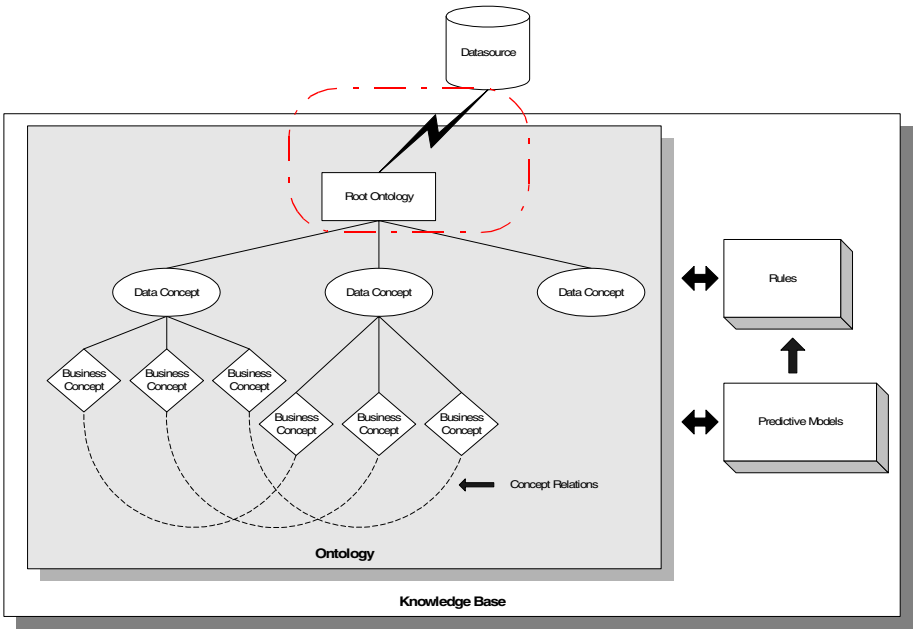
This chapter provides step-by-step instructions for defining an ontology that links to your datasources and is relevant to your business requirements. The Black Pearl Knowledge Broker ontology consists of four components:

- *Datasource Connections,* which are URLs, strings, or pathnames to disparate datasources—Web, relational database, and flat files—containing data pertinent to your business
- *Data Concepts*, which are the vocabulary terms that map directly to one or more of your datasources
- *Business Concepts*, which are the vocabulary terms defined by you that are relevant to your industry or company and which map to other concepts in the ontology

Managing Datasource Connections and Datasources

Datasource Connections are the URLs, strings, or pathnames to your datasources. *Datasources* are the JDBC-compliant databases and structured files containing data relevant to your business. You use the Data Concept Editor to manage your Datasource Connections and Datasources.

Figure 5-1. Datasource Connection and Datasources



Manipulating Multi-User Datasource Connections

Server users cannot directly add, modify, or delete Datasource Connections. The Server version of the Knowledge Broker lacks the graphical user interface (GUI) necessary for these tasks. The Client and Standalone versions of the Knowledge Broker incorporate a suitable GUI (see *Navigating the Black Pearl Knowledge Broker Interfaces on page 223*) that lets users add, modify, and delete Datasource Connections.

Understanding Client-Server Location Constraints

Although you can install the Client and Server on different machines, for optimal data connection management, you should install both Client and Server on the same machine. This is because the data connection information is generated on and is relative to the Client machine but is then stored within the Server machine's directory.

For example, on the Client installation machine, an ODBC datasource connection might be set to the Client-specific directory `C:\foo`. However, if the Client and Server are located on different machines, then the datasource location on the Server might be `D:\bar`. If the Client overwrites the `D:\bar` setting with its own `C:\foo` setting, an attempt by the Server to access data will generate errors.

Adding a Datasource Connection

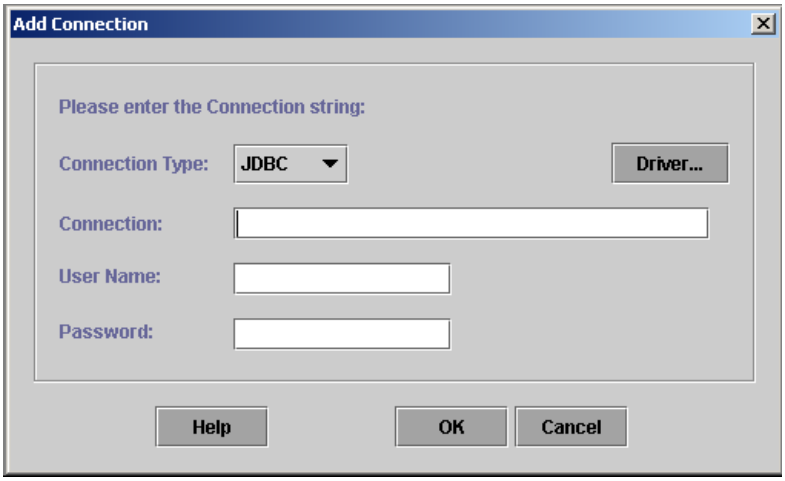
1 Start the Data Concept Editor, if it is not currently open.

2 Complete one of the following options:

   a Select Datasource > Add Connection. The Add Connection dialog displays.

   b Click Add Connection on the Data Concept Editor toolbar.

Figure 5-2. Add Connection



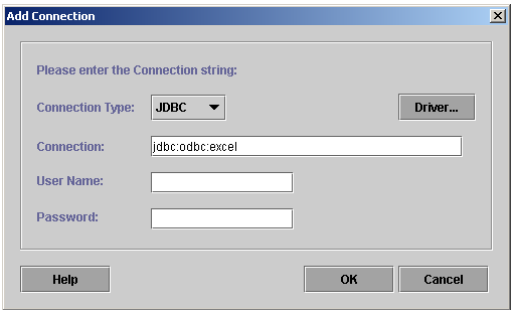
3 Select JDBC from the Connection Type drop-down list.

Currently, JDBC is the only option available. However, future software updates of the Black Pearl Knowledge Broker will introduce support for other Connection Types.

4 Complete one of the following options:

a To connect to an ODBC-compliant datasource, type a suitable Connection String into the **Connection:** field. For example, to connect to an Excel file specified using the ODBC Data Source Administrator (see *Configuring ODBC Data Source Drivers on page 52*) you should enter `jdbc:odbc:excel`:

Figure 5-3. Add Connection - Excel



b To connect to an RDBMS requires a complex Connection String containing parameters and values conforming to that RDBMS. Oracle, for example, requires a specific Connection String format.

If you are adding an Oracle datasource, you must use the Connection String format:

```
connection string@<server>:<TCP port>:<Instance Name>/
<schema>/ [<tables>/<columns>]
```

For example:

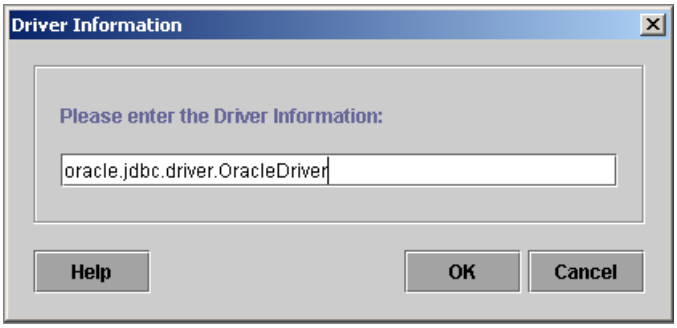
```
jdbc:oracle:thin:@bpd1:1521:orcl/PARKER/*/*      where
```

Connection string is `jdbc:oracle:thin:`
Server is `bpd1:`
TCP port is `1521:`
Instance Name is `orcl`
Schema is `/PARKER`
Table is `/*` (where `*` specifies all tables; this is optional)
Column is `/*` (where `*` specifies all columns; this is optional))

In addition, you may have to specify a suitable database driver name. You may have installed a `.jar` file that contains JDBC Drivers (see *Step 10 on page 49*). During installation, you located the file that contains the driver. Now you must specify the JDBC name that uniquely identifies the driver within that file. For example, when attaching to an Oracle datasource, you should click **Driver** and type:

```
oracle.jdbc.driver.OracleDriver:
```

Figure 5-4. JDBC Driver Name Information



Your network administrator or DBA (database administrator) can help you locate suitable files that contain JDBC drivers, and identify the particular JDBC driver names.

5 (Optional) Type your User Name and Password in the appropriate fields, if the datasource is designated as restricted access.

6 Click OK. If the connection is successful, the Connection Successful dialog displays.

Figure 5-5. Connection Successful

7 Click OK to exit the dialog. The Datasource displays as a hierarchical tree in the left-hand Physical panel.

The term *physical* refers to the fact that the Black Pearl Knowledge Broker accesses your data for the purpose of making real-time recommendations and predictive models. The Black Pearl Knowledge Broker does not make any changes to your data. It accesses your data for information processing only.

Figure 5-6. Datasource Tree

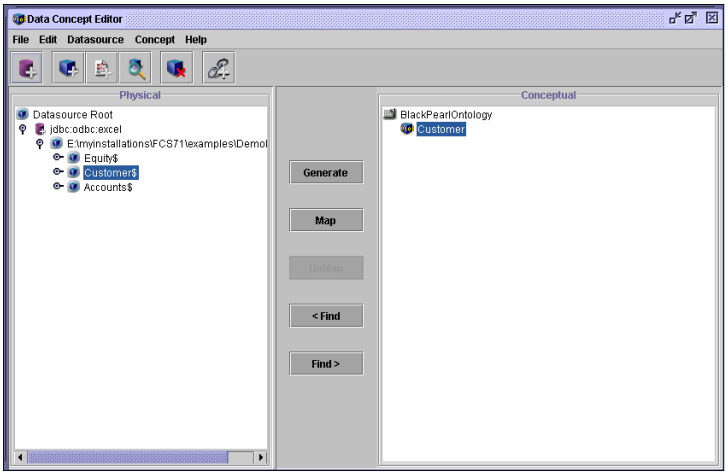


Figure 5-6 displays an expanded view of the added Datasource Connection. The hierarchy is:

- *Datasource Root*, which is the non-removable, default Black Pearl Knowledge Broker setting.
- *Datasource Connection*, which is the URL, string, or pathname to the Datasource; for example, `jdbc:oracle:excel`. There can be multiple Datasource Connections within the Datasource Root.
- *Datasource*, which is the JDBC-compliant database or XML directory; for example, DemoParker. There can be multiple Datasources within a single Datasource Connection.
- *Datasource Component*, which is the database table or XML file; for example, Equity. There can be multiple Datasource Components within a single Datasource.
- *Datasource Properties*, which are the database columns or XML pages; for example, Beta, Industry, Symbol. There can be multiple Datasource Properties within a single Datasource Component.

When referring to several items listed in the Physical panel, we shall use the term Data Item. When referring to an individual item, we shall use Datasource Connection, Datasource, Datasource Component, or Datasource Property (as appropriate.)

Deleting a Data Item

As your business needs change, you may no longer need a particular Data Item. You use the Delete Connection option to remove these items from the Data Concept Editor.

If you delete a Data Item, you remove it from the ontology. This may affect any previously defined concept relations, rules, or predictive models.

1. Start the Data Concept Editor, if it is not currently open.
2. Select the Data Item to be deleted.
3. Select Datasource > Delete Connection. The Data Item is removed from the Data Concept Editor.

If you delete a Datasource Connection, you also delete all Datasources, Data Components, and Datasource Properties associated with the Datasource Connection. However, Data Items are not permanently removed until you save your changes the Knowledge Base.

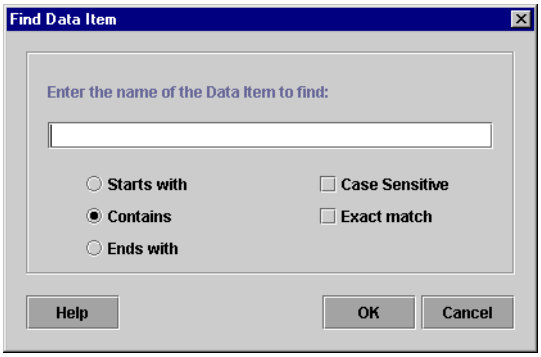
Finding a Data Item

If you have long lists of items displayed in the Physical panel, you may use the Find option to locate quickly a specific Data Item.

1 Start the Data Concept Editor, if it is not currently open.

2 Select Datasource > Find (or click the <Find button). The Find Data Item dialog displays.

Figure 5-7. Find Data Item



3 Type a keyword or string in the text field.

4 Complete one of the following options:

- **a** Select Starts With to restrict the search to those items that start with a particular keyword or string.
- **b** Select Contains to restrict the search to those items that contain a particular keyword or string.
- **c** Select Ends With to restrict the search to those items that end with a particular keyword or string.

5 (Optional) Select Case Sensitive to restrict the search to those items that match the case structure for the keyword or string.

6 (Optional) Select Exact Match to restrict the search to the keyword or string specified in the text field.

7 Click OK. If the search is successful, the selected item is highlighted.

Select Datasource > Find Again to search for any additional items that match your search parameters.

Viewing Data Item Details

1. Start the Data Concept Editor, if it is not currently open.
2. Select a Data Item.
3. Select Datasource > View Details. The Detail dialog displays.
4. Click OK to exit the View Details dialog.

Figure 5-8. View Details: Scope

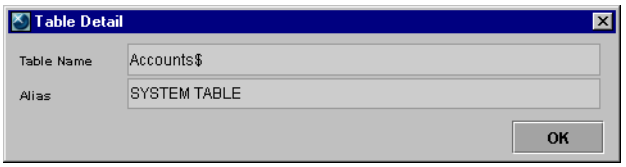
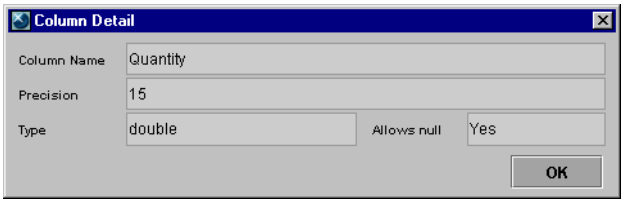


Figure 5-9. View Details: Property



Both Datasources and Datasource Components are Table elements. A *Table* element contains other Data Items that will be deleted if the Table element is deleted. Figure 5-8 displays information for a Datasource Component.

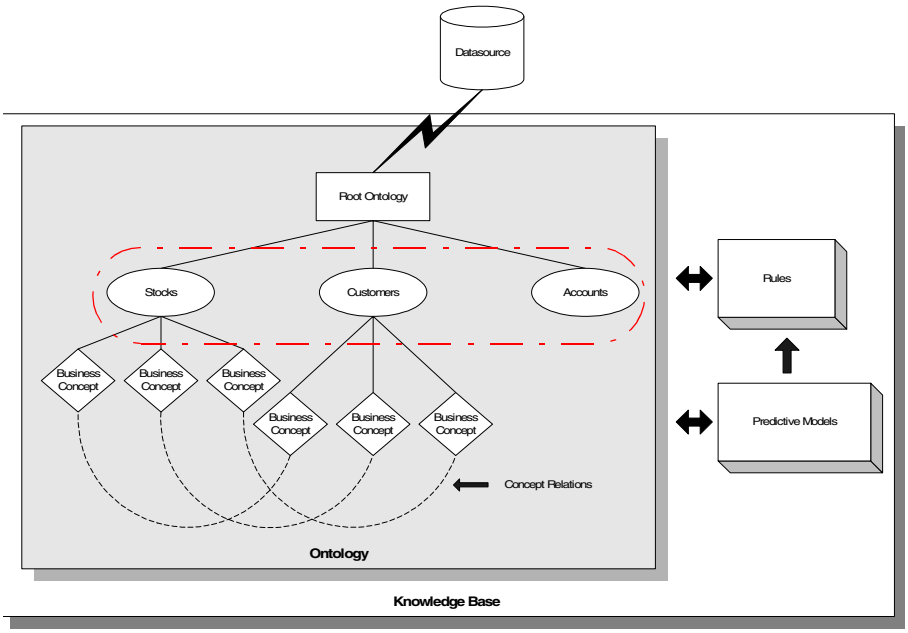
A Datasource Property is detailed as a Column element. A *Column* element, while associated with a Table element, can be deleted without impact on the Table element. Figure 5-9 displays information for a Datasource Property.

Managing Data Concepts and Properties

*Data Concept*s are vocabulary terms that map directly to one or more of your datasources. For example, you can create a Stocks Data Concept and map it to an Equity table in your database. *Data Concept Properties* are vocabulary terms that function as attributes or defining characteristics for a Data Concept. For example, Beta, Industry, and Symbol can be characteristics of a Stock.

The Black Pearl Knowledge Broker uses the Data Concepts and Properties that you define in your ontology to communicate with your datasources. Data Concepts and Data Concept Properties can be either automatically generated from your Data Components, or manually added.

Figure 5-10. Adding Data Concepts to an Ontology



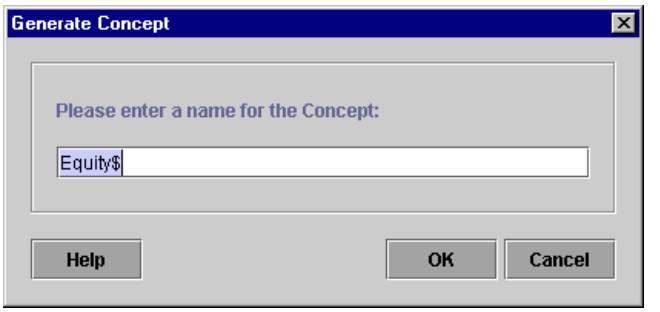
Generating a Data Concept

1 Start the Data Concept Editor, if it is not currently open.

2 Select a Datasource Component.

3 Click Generate (or select Concept > Generate Concept). The Generate Concept dialog displays.

Figure 5-11. Generate Concept



4 (Optional) By default, the name for the Data Concept is identical to the Datasource Component. Complete one of the following options:

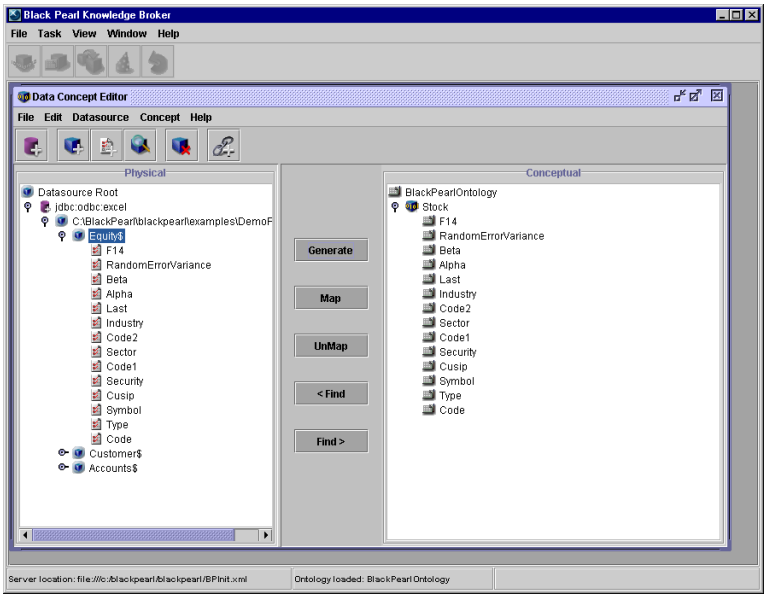
   a Type a new name and click OK. The generated Data Concept (and all associated Properties) displays in the Conceptual panel. (See Figure 5-14 for an expanded view.)

   b Click OK to accept the default name. The generated Data Concept (and all associated Properties) displays in the Conceptual Panel. (See Figure 5-14 for an expanded view.))

If you generate a Data Concept, you automatically generate Data Concept Properties. The Data Concept maps to the Datasource Component, while the Data Concept Properties map to the Datasource Properties. If you want to change the default mapping, you must manually change the mapping parameters. (See *Mapping a Direct Join on page 94* for detailed instructions.)

The Data Concept displays as a hierarchical tree in the right-hand Conceptual panel. The term *Conceptual* refers to the fact that Data Concepts and Data Concept Properties are a notational vocabulary pointing to your Data Items.

Figure 5-12. Data Concept Generated



The hierarchy is:

- *Root Ontology*, which is the non-removable, default Black Pearl Knowledge Broker setting.
- *Data Concept*, which functions as a general category within the ontology and maps to your Datasources; for example, Stock maps to Equity$ in Figure 5-12.
- *Data Concept Properties*, which function as the attributes or characteristics of the Data Concept; for example, Beta, Industry, Symbol are characteristics of Stock in Figure 5-12.

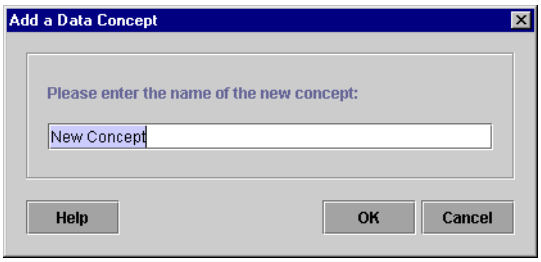
When referring to several items listed in the Conceptual panel, we use the term *Concept Item*. When referring to an individual item, we use *Data Concept* or *Data Concept Property* (as appropriate.)

Adding a Data Concept

1 Start the Data Concept Editor, if it is not currently open.

2 Select Concept > Add Concept. The Add Data Concept dialog displays.

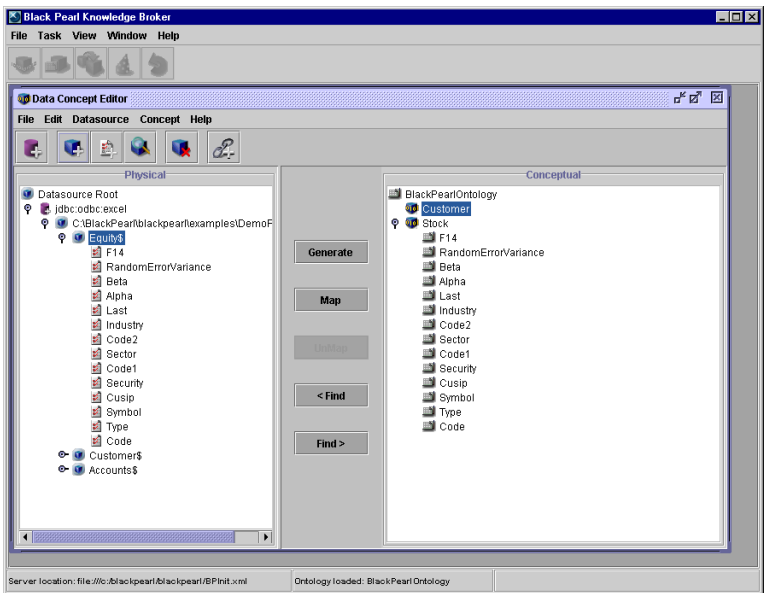
Figure 5-13. Add Data Concept



3 Type a name for the new Data Concept; for example, Customer.

4 Click OK. The new Data Concept displays in the Conceptual panel. (Figure 5-14 displays an added Customer Data Concept. Each new Data Concept appears at the top of the Conceptual list.)

Figure 5-14. Data Concept Added



Adding a Data Concept Property

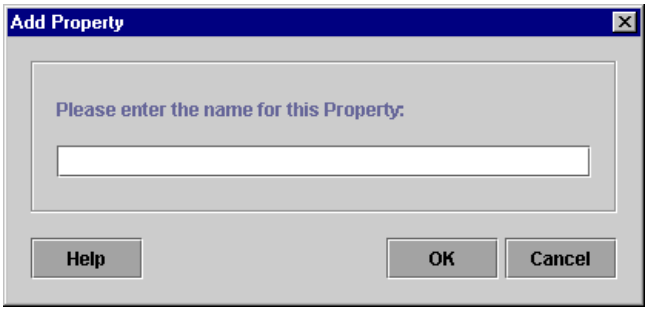
If you add a Data Concept, you must manually add Data Concept Properties. Data Concept Properties are vocabulary terms that function as attributes or defining characteristics for a Data Concept. For example, Beta, Industry, and Symbol can be characteristics of a Stock.

1 Start the Data Concept Editor, if it is not currently open.

2 Select the Data Concept to which you want to add a Data Concept Property.

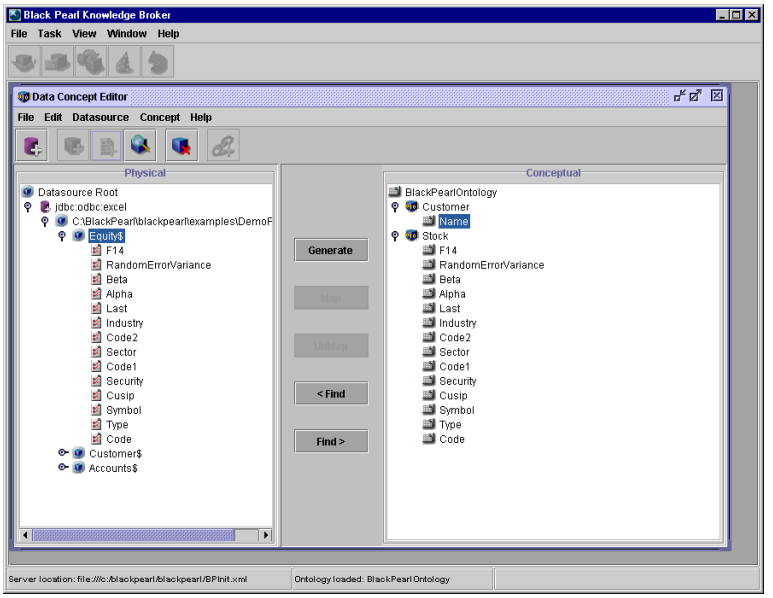
3 Select Concept > Add Property. The Add Property dialog displays.

Figure 5-15. Add Property



4 Type the name for the new Data Concept Property. A new Data Concept Property displays under the selected Data Concept. Figure 5-16 displays a Data Concept with a single added Data Concept Property.

Figure 5-16. Concept and Properties Added



Adding Calculated Properties

Calculated Properties enable you to define a set of values that work as a single value when determining real-time recommendations and running predictive models. For example, you can use Calculated Properties to create a single debt-to-income ratio to predict credit risk, rather than use debt and income as two independent variables. The Black Pearl Knowledge Broker supports the following Calculated Properties:

Predictive Model Calculators

MLPCalculator

- Embeds a pre-computed neural network. These can be generated within the Knowledge Discovery Center (see *Creating and Training a Multi-Layer Perceptron on page 167*).
- The only map parameter is `MLP_urlString`. This indicates the location of the XML file that stores the neural network.

DecisionTreeCalculator

- Embeds a pre-computed decision tree. These can be generated within the Knowledge Discovery Center (see *Generating and Updating a Decision Tree on page 161*).
- The only map parameter is `DT_urlString`. This indicates the location of the XML file that stores the decision tree.

RuleSetCalculator

- Embeds a pre-computed rule set. These can be derived from decision trees within the Knowledge Discovery Center (see *Pruning a Decision Tree on page 162*).
- The only map parameter is `RS_urlString`. This indicates the location of the XML file that stores the rule set.

Mathematical Calculators

WeightedSumCalculator

- Performs a weighted sum of property values. If you use only a single property as input, you can use this calculator to multiply that property's values by an arbitrary constant.
- There is one initial parameter: `Sum_Number_Of_Args`. Set this to the number of properties whose values you want to sum.
- The number of map and additional parameters is equal to, respectively,
  `2 * Sum_Number_Of_Args` and
  `2 * Sum_Number_Of_Args + 1`
- The map parameters `Sum_Input_Para_i` refer to the properties whose values are to be summed (with a range of `0 <= i < Sum_Number_Of_Args`).
- The additional parameters `Sum_Weight_Para_i` refer to the weights used in the summation (with a range of `0 <= i < Sum_Number_Of_Args`).
- The additional parameter `Sum_Ignore_Illegal_Values_Flag` can be set to either `TRUE` or `FALSE`. If this is set to `TRUE`, then if any parameter values are not legal (for example, a weight that is a string rather than a number) the calculator will return nothing. If set to `FALSE`, the calculator will raise an exception that will cause a dialog box to appear.

WeightedMinMaxCalculator

- Performs a weighted minimum or maximum of property values.
- There is one initial parameter: `Min_Max_Number_Of_Args`. Set this to the number of properties whose values you want to minimize or maximize.
- The number of map and additional parameters is equal to, respectively,
  `2 * Min_Max_Number_Of_Args` and
  `2 * Min_Max_Number_Of_Args + 1`
- The map parameters `Min_Max_Input_Para_i` refer to the properties whose values are to be minimized or maximized
  (with a range of `0 <= i < Min_Max_Number_Of_Args`).
- The additional parameters `Min_Max_Weight_Para_i` refer to the weights used in the minimization or maximization
  (with a range of `0 <= i < Min_Max_Number_Of_Args`).
- The additional parameter `Min_Max_Ignore_Illegal_Values_Flag` can be set to either `TRUE` or `FALSE`. If this is set to `TRUE`, then if any parameter values are not legal (for example, a weight that is a string rather than a number) then the calculator will return nothing. If set to `FALSE`, the calculator will raise an exception that will cause a dialog box to appear.

WeightedAverageCalculator

- Performs a weighted average of parameter values. There is one initial parameter: `Average_Number_Of_Args`. Set this to the number of properties whose values you want to average.
- The number of map and additional parameters is equal to, respectively:
  `2 * Average_Number_Of_Args` and
  `2 * Average_Number_Of_Args + 1`
- The map parameters `Average_Input_Para_i` refer to the properties to be averaged (with a range of `0 <= i < Average_Number_Of_Args`).
- The additional parameters `Average_Weight_Para_i` refer to the weights used in the averaging (with a range of `0 <= i < Average_Number_Of_Args`).
- The additional parameter `Average_Ignore_Illegal_Values_Flag` can be set to either `TRUE` or `FALSE`. If this is set to `TRUE`, then if any parameter values are not legal (for example, a weight that is a string rather than a number) then the calculator will return nothing. If set to `FALSE`, the calculator will raise an exception that will cause a dialog box to appear.

DivisionCalculator

- Divides one concept property value by another. If the denominator equals zero (`0`), then the result is undefined.
- There are two map parameters: `Denominator_Value` and `Numerator_Value`. These refer to the classical components of the division calculation.

FunctionCalculator

- Computes a trigonometric or algebraic function on a property value.
- The one map parameter is `Function_Input`. Set this to indicate on which property you want to apply the function.
- There is one additional parameter: `Function_Name`. This indicates what function to apply. The available functions are described in *Table 5-1 on page 90*.

The definition of these functions is outside the scope of this manual. For further information and acceptable input value ranges, please consult a math reference.

Table 5-1. Functions Available in the FunctionCalculator Calculated Property

Function	Description
sin	Returns the sine in radians.
cos	Returns the cosine in radians.
tan	Returns the tangent in radians.
sinh	Returns the hyperbolic sine in radians.
cosh	Returns the hyperbolic cosine in radians.
tanh	Returns the hyperbolic tangent in radians.
asin	Returns the inverse sine.
acos	Returns the inverse cosine.
atan	Returns the inverse tangent.
asinh	Returns the inverse hyperbolic sine.
acosh	Returns the inverse hyperbolic cosine.
atanh	Returns the inverse hyperbolic tangent.
cot	Returns the cotangent in radians.
gamma	Returns the gamma.
log	Returns the natural logarithm.
log10	Returns the base 10 logarithm.
logGamma	Returns the natural logarithm of the gamma.
sqrt	Returns the square root.
abs	Returns the absolute value.

Format Transformation Calculators

AgeCalculator

- Converts a date into a floating point number representing the number of years elapsed from the input date to the present.
- The one map parameter is `Date_Attribute_Value`. This contains the input date.
- There is an additional parameter: `Date_Present_Value`. This lets you define a 'present date' different from today's date (for the purposes of the calculation).
- There is a formatting parameter: `Date_Format_String`. This is a standard Java-format string that governs which dates the calculator can parse.

  For instance, if the `Date_Format_String` is `yyyy/MM/DD`, then the calculator will parse `1976/08/05` as a date, but will reject `76/08/05` as not a date.

BooleanStateCalculator

- Compares the value of a specified property against an expected value and returns a value that indicates whether or not the values match.
- There is one map parameter: `Boolean_Input_Para_Attribute_Name`. This specifies the input parameter.
- There is one additional parameter: `Boolean_Input_Para_Attribute_Value`. Use this to specify your expected value.
- If the input value equals the expected value, the calculator returns the value specified by a `Boolean_Equal_Value` parameter.
- If they do not match, the calculator returns a value specified by a `Boolean_Not_Equal_Value` parameter. Some suggested values are: `false`, `untrue`, `incorrect`, and so on.

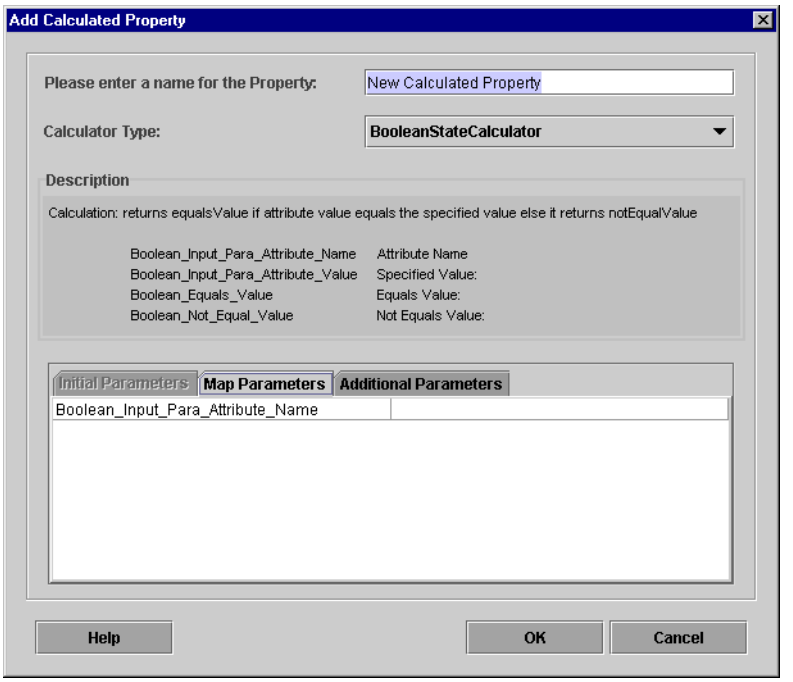
Adding a Calculated Property

1 Start the Data Concept Editor, if it is not currently open.

2 Select a Data Concept from the Conceptual list.

3 Select Concept > Add Calculated Property. The Calculated Property dialog displays.

Figure 5-17. Add Calculated Property



4 Type a name for the Calculated Property in the Name Property field.

5 Select a calculator from the drop-down list. The Initial Parameters for the selected calculator display in the lower panel.

6 Type a Map Parameter value for each Initial Parameter.

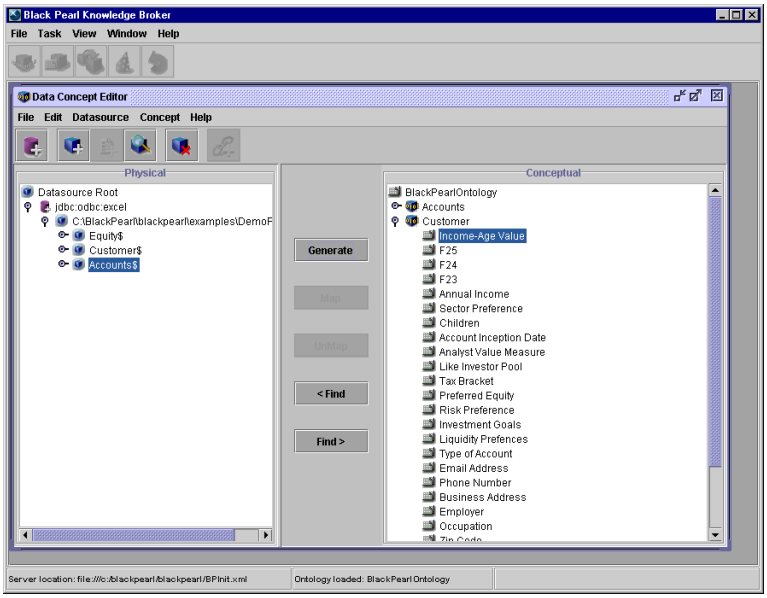
Some calculators provide drop-down lists that you can use to select the values for Initial Parameters.

7 (Optional) Click Additional Parameters and type a value for each item listed.

Not all calculators provide Additional Parameters.

8 Click OK. The new Calculated Property displays beneath the Data Concept.

Figure 5-18. Calculated Property Added



Mapping Data

Mapping creates a join—an explicit statement of the relationship between Data Items and Concept Items. The Black Pearl Knowledge Broker supports three types of joins:

- *Direct,* which maps a single Data Item to a single Concept Item. By default, whenever you generate a Data Concept from a Datasource Component, you map a Direct join.
- *Star*, which maps two different Data Items to a single Concept Item.
- *Cascade*, which maps three different Data Items to a single Concept Item by using a foreign key from the first to lookup the key field in the second, which points to the third.

Mapping a Direct Join

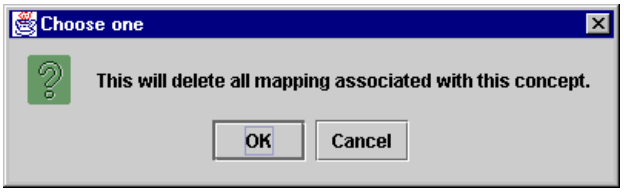
1 Start the Data Concept Editor, if it is not currently open.

2 Select a Data Item.

3 Select a Concept Item.

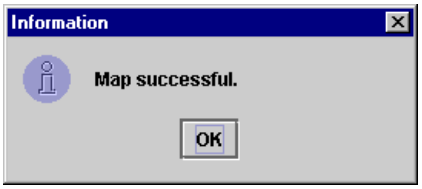
4 Click Map (or select Concept > Add Mapping, or click Add Mapping on the toolbar). If the two items are mappable, the Map Confirmation dialog displays, indicating all previously designated mapping for this Data Concept will be overwritten.

Figure 5-19. Map Confirmation



5 Click OK. The Mapping Successful dialog displays.

Figure 5-20. Mapping Successful



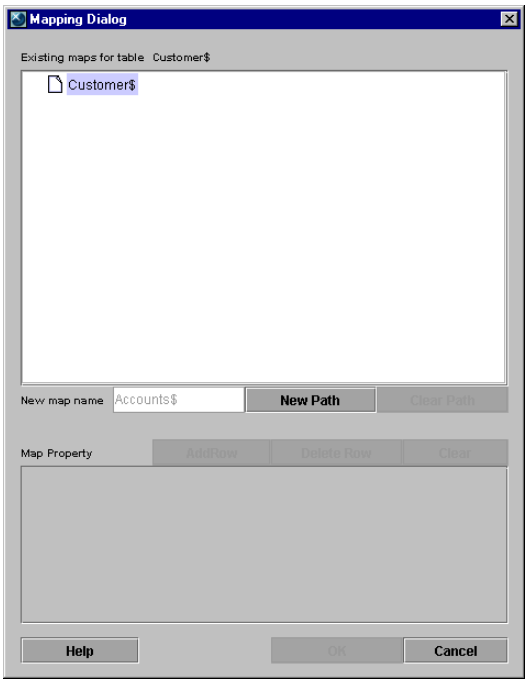
If the two items are not mappable, an Error dialog displays. Click OK to exit the dialog and terminate the mapping process.

6 Click OK. The Data Item and Concept Item are mapped.

Mapping a Star Join

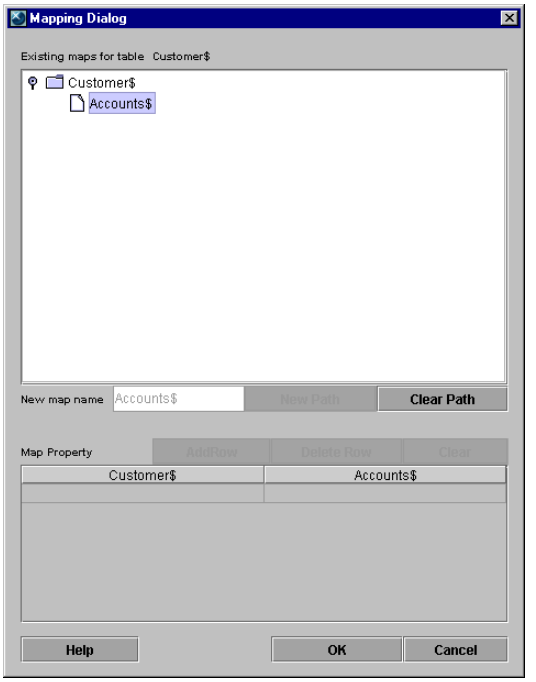
1 Start the Data Concept Editor, if it is not currently open.
2 Map a Direct join between a Datasource Component and a Data Concept.
3 Select the newly mapped Data Concept.
4 Select Concept **>** Add Property.
5 Type a name for the new Data Concept Property.
6 Click OK.
7 Select the newly added Data Concept Property.
8 Select a Datasource Property belonging to a Datasource Component different from the one selected in step 2.
9 Click Map. The Map dialog displays.

Figure 5-21. Star Join Mapping



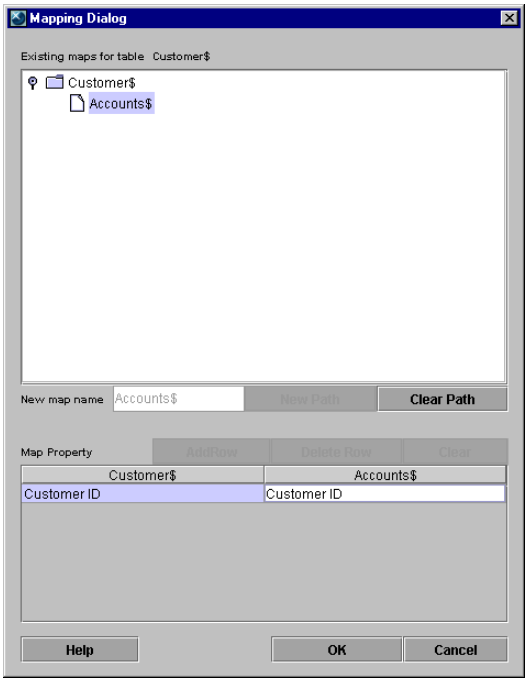
10 Click New Path. The Mapping dialog changes to reflect the new path.

Figure 5-22. Star Join Path



11 Select an item from both the left-hand and right-hand columns. This will be the key used to map the Star join.

Figure 5-23. Star Join Keys



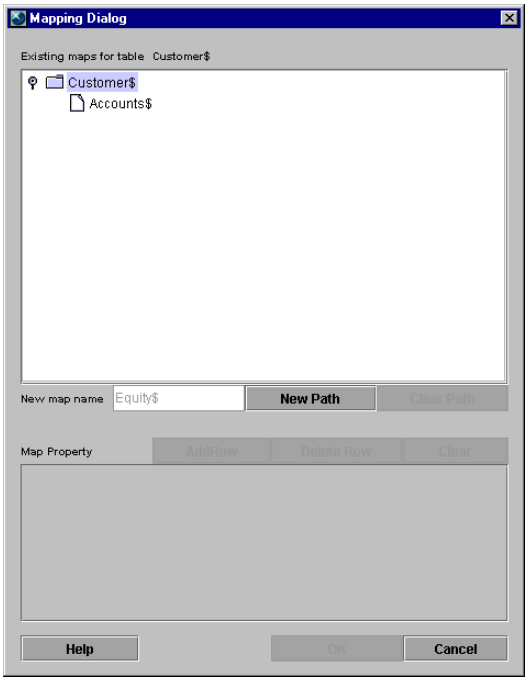
12 Click OK. The Map Successful dialog displays.

13 Click OK.

Mapping a Cascade Join

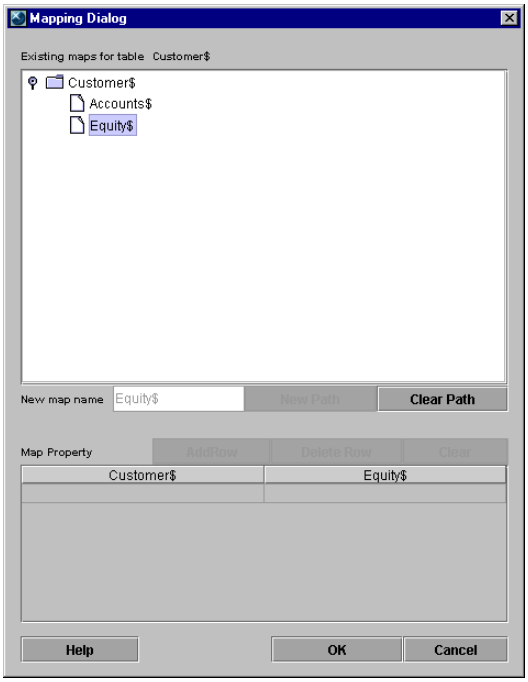
1. Start the Data Concept Editor, if it is not currently open.
2. Select Concept Item that already has a Star join.
3. Add a new Data Concept Property.
4. Select the newly added Data Concept Property
5. Select a Datasource Property not mapped to the item selected in step 2.
6. Click Map. The Map dialog displays.

Figure 5-24. Cascade Join Mapping



7 Click New Path. The Path dialog displays.

Figure 5-25. Cascade Join Path



8 Select the item you are mapping; for example, Equity$.

9 Select an item from both the left-hand and right-hand columns. This will be the key used to map the Cascade join.

10 Click OK. The Map Successful dialog displays.

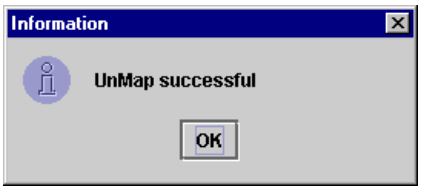
11 Click OK.

Unmapping a Concept Item

When you unmap a Concept Item, you are removing the connection between a Concept Item and a Data Item.

1. Start the Data Concept Editor, if it is not currently open.
2. Select the Data Item to be unmapped.
3. Select the corresponding Concept Item to be unmapped.
4. Click UnMap (or select Concept > UnMap). If the two items are unmappable, the UnMap Successful dialog displays.
5. Click OK.

Figure 5-26. UnMap Successful



If the two items cannot be unmapped, an Error Message displays. Refer to *Error 2002 on page 267* for detailed information.

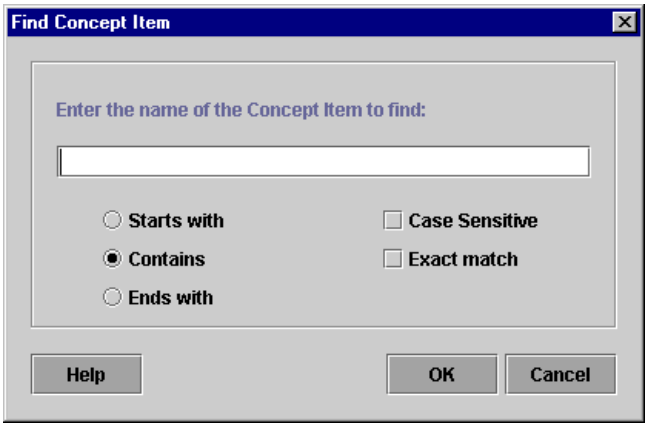
Finding a Concept Item

If you have long lists of items displayed in the Conceptual panel, you may use the Find option to locate quickly a specific Concept Item.

1 Start the Data Concept Editor, if it is not currently open.

2 Select Concept > Find (or click the Find > button, or click Find Data Concept on the toolbar). The Find Concept Item dialog displays.

Figure 5-27. Find Concept



3 Type a keyword or string in the text field.

4 Complete one of the following options:

   a Select Starts With to restrict the search to those items that start with a particular keyword or string.

   b Select Contains to restrict the search to those items that contain a particular keyword or string.

   c Select Ends With to restrict the search to those items that end with a particular keyword or string.

5 (Optional) Select Case Sensitive to restrict the search to those items that match the case structure of the keyword or string.

6 (Optional) Select Exact Match to restrict the search to the keyword or string specified in the text field.

7 Click OK. If the search is successful, the selected item is highlighted.

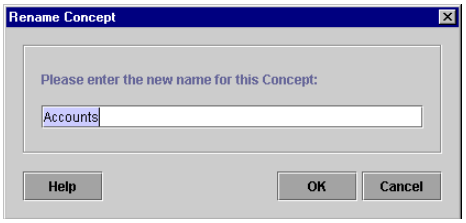
Select Concept > Find Again to search for any additional items that match your search parameters.

Renaming a Data Concept

When you rename a Data Concept, you are only changing the concept name, not its mappings.

1 Start the Data Concept Editor, if it is not currently open.
2 Select the Data Concept to be renamed.
3 Select Concept > Rename. The Rename Concept dialog displays.

Figure 5-28. Rename Concept



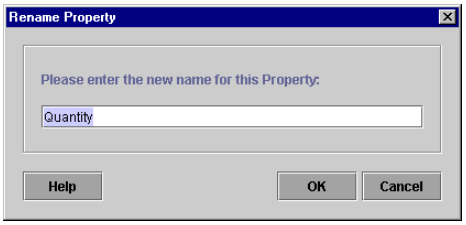
4 Type the new name for the Data Concept.
5 Click OK.

Renaming a Data Concept Property

When you rename a Data Concept Property, you are only changing the concept name, not its mappings.

1 Start the Data Concept Editor, if it is not currently open.
2 Select the Data Concept Property to be renamed.
3 Select Concept > Rename. The Rename Property dialog displays.

Figure 5-29. Rename Property



4 Type the new name for the Data Concept Property.
5 Click OK.

Deleting a Concept Item

From time-to-time you may need to delete a Data Concept or Data Concept Property. You use the Data Concept Editor's Delete function to remove unneeded Concept Items.

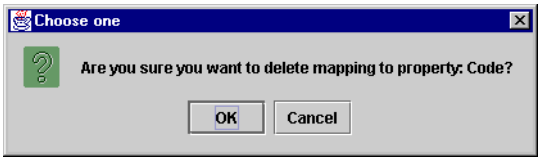
If you delete a Data Item, you are removing it from the ontology. This may affect any previously defined concept relations, rules, or predictive models.

1 Start the Data Concept Editor, if it is not currently open.

2 Select the Concept Item to be deleted.

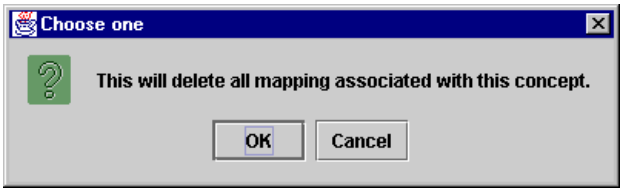
3 Select Concept > Delete (or click Delete Data Concept on the toolbar).

Figure 5-30. Confirm Delete



4 Click OK. The UnMap dialog displays, indicating that deleting this item also deletes any mappings to a Datasource Item.

Figure 5-31. Unmap Confirmation

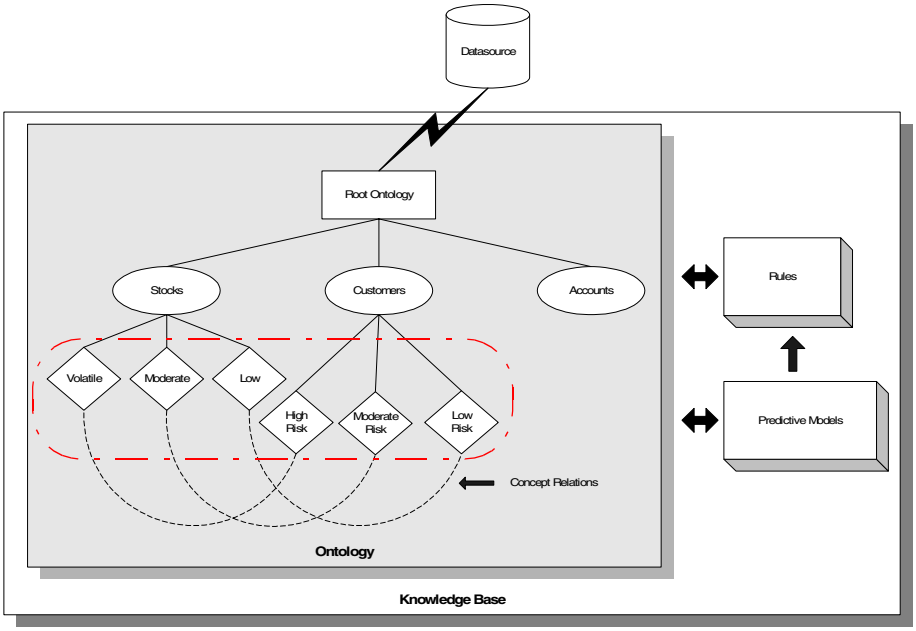


5 Click OK. The item is removed from the Conceptual panel.

Managing Business Concepts and Filters

A *Business Concept,* which is a vocabulary term relevant to your business, is a sub-category of a Data Concept or another Business Concept. Each Business Concept is differentiated from its Parent Concept by one or more filters. A *filter* specifies the parameters that define the Business Concept. For example, if you add a Volatile Business Concept that is a sub-category of the Stock Data Concept, you must specify the criteria that differentiate Volatile Stock from all other Stock.

Figure 5-32. Adding Business Concepts to an Ontology



Adding a Business Concept

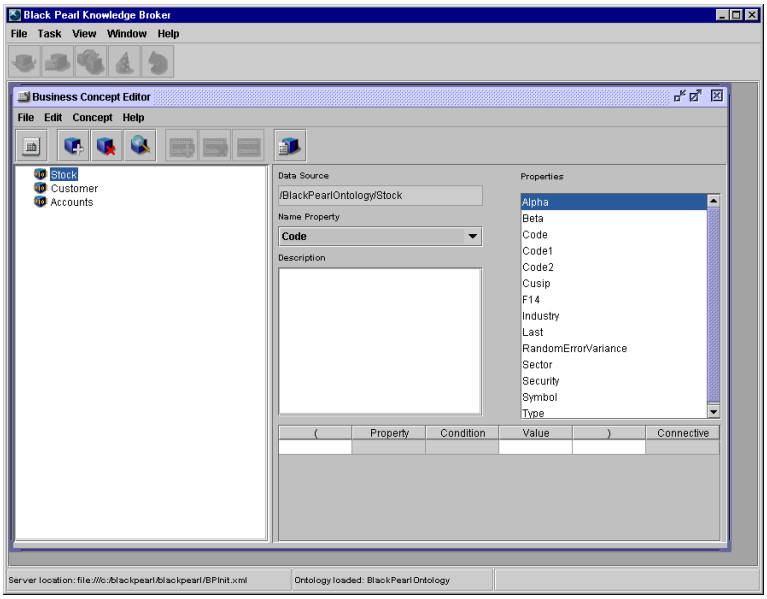
1 Complete one of the following options:

   a Select Task > Business Concept Editor on the Launcher menu bar. The Business Concept Editor displays.

   b Click Business Concept Editor on the Launcher toolbar.

2 Select an item from the Concept list. The item can be either a Data Concept or another Business Concept. The Business Concept Editor displays.

Figure 5-33. Business Concept Editor

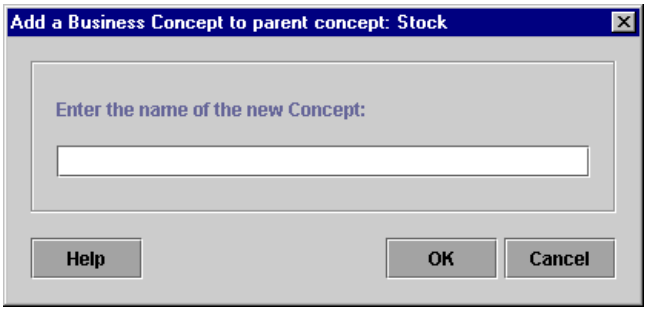


3 Complete one of the following options:

   a Select Concept > Add Business Concept. The Add a Business Concept dialog displays.

   b Click Add Concept on the Business Concept Editor toolbar. The Add a Business Concept dialog displays.

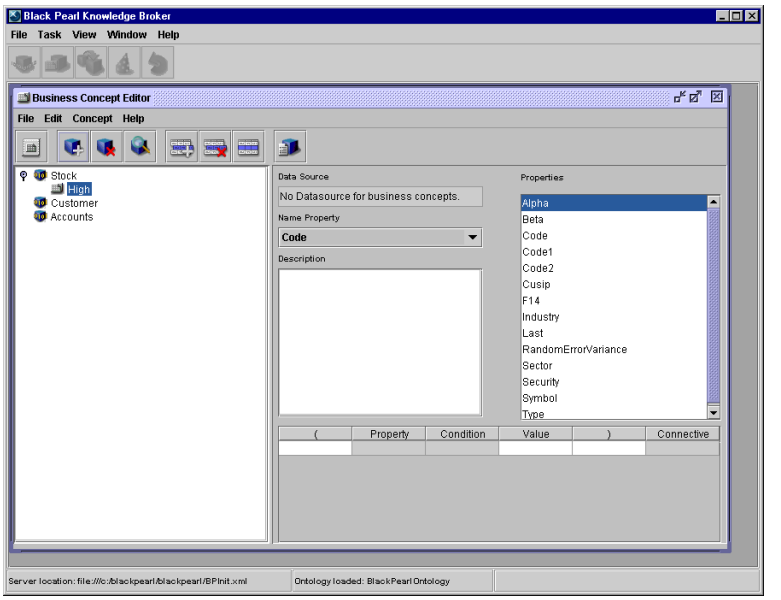
Figure 5-34. Add Business Concept



4 Type a name for the new Business Concept.

5 Click OK. The newly added Business Concept displays as the last item beneath the Parent Concept. In addition, all Properties assigned to the Parent Concept display in the Properties list.

Figure 5-35. Business Concept Added



The Name Property is the primary key used to organize properties for the Data and Business Concepts displayed in the Business Concept Editor, as well as the type of Instances that can be displayed. For example, you may list properties for the Customer Data Concept by Age, Name, or Customer ID. It is recommended that you select a primary key that will be a unique, non-duplicated identifier.

6 Complete one or both of the following options:

   a Add a filter. Go to *Adding a Filter on page 107* to add a filter to the new Business Concept.

   b Repeat steps 2 through 5 to add another new Business Concept.

Adding a Filter

A filter differentiates a Business Concept from its Parent Concept by specifying the parameters that the Black Pearl Knowledge Broker will use to evaluate, process, and sort data.

A filter can only be added to a Business Concept, not the associated Data Concept.

1 Start the Business Concept Editor, if it is not currently open.

2 Select a Business Concept.

3 Click the Property field, and select the appropriate Property from the drop-down list.

4 Click the Condition field, and select the appropriate Condition from the drop-down list.

5 Click the Value field, and type the appropriate Value.

6 (Optional) If a more complex filter is required, click the Connective field and select the appropriate Connective from the drop-down list. Repeat steps 3, 4, and 5 for each filter that you want associated with the Business Concept.

Figure 5-36. Filter Added

(	Property	Condition	Value	)	Connective
	Beta	>	5		

Adding a Filter Row

Each filter is defined on a single *filter row*. If you want to add a filter, you must use the Add Filter Row option before you can define the filter.

1 Start the Business Concept Editor, if it is not currently open.

2 Select a Business Concept from the Concept list.

3 Select Concept > Add Filter Row (or click Add Filter Row on the toolbar).

4 Complete steps 3 through 6 in *Adding a Filter*.

Figure 5-37. Multiple Filters Added

(	Property	Condition	Value	)	Connective
	Beta	>	5		and
	Beta	<	5.1		

Clearing a Filter

If you want to modify the parameters for a filter, you must use the Clear Filter option to remove the filter but retain the filter row.

1 Start the Business Concept Editor, if it is not currently open.

2 Select a Business Concept.

3 Select Concept > Clear Filter (or click Clear Filter Row on the toolbar). All filters associated with the selected Business Concept are removed.

Deleting a Filter Row

If you want to remove both the filter and filter row, you must use the Delete Filter Row option.

1 Start the Business Concept Editor, if it is not currently open.

2 Select a Business Concept.

3 Select a Filter Row.

4 Select Concept > Delete Filter Row (or click Delete Filter Row on the toolbar). The selected Filter Row is removed.

5 Repeat steps 3 and 4 for each Filter Row that you want to delete.

Renaming a Concept Item

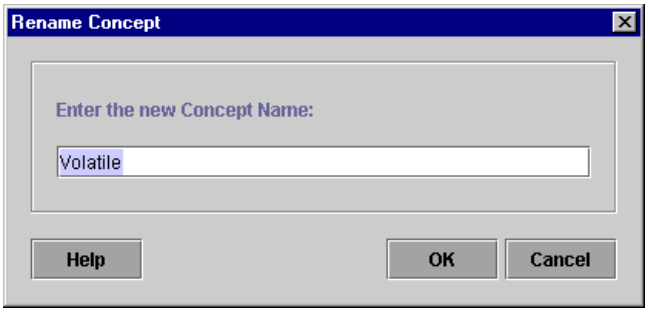
You can rename both Data and Business Concepts by using the Rename option. Renaming the Business Concept does not change its relationship with its Parent Concept. Renaming a Data Concept does not change its mapping to a Data Item.

1 Start the Business Concept Editor, if it is not currently open.

2 Select a Concept Item.

3 Select Concept > Rename. The Rename Concept dialog displays.

Figure 5-38. Rename Concept



4 Type the new name in the Concept Name field.

5 Click OK. The new name displays in the Concept list.

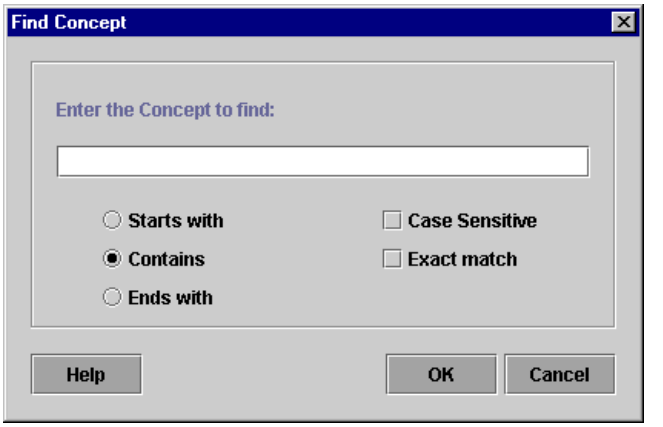
Finding a Concept Item

If you have long lists of items displayed in the Conceptual panel, you may use the Find option to locate quickly a specific Concept Item.

1 Start the Business Concept Editor, if it is not currently open.

2 Select Concept > Find Concept (or click Find Concept on the toolbar). The Find Concept dialog displays.

Figure 5-39. Find Concept



3 Type a keyword or string in the text field.

4 Complete one of the following radio buttons:

   a Select Starts with, if you want to restrict the search to those items that start with a particular keyword or string.

   b Select Contains, if you want to restrict the search to those items that contain a particular keyword or string.

   c Select Ends with, if you want to restrict the search to those items that end with a particular keyword or string.

5 (Optional) Click Case Sensitive, if you want to restrict the search to those items that match the case structure for the specified keyword or string.

6 (Optional) Click Exact Match, if you want to restrict the search to the keyword or string specified in the text field.

7 Click OK. If the search is successful, the selected item is highlighted.

8 (Optional) Select Concept > Find Again to search for any additional items that match your search parameters.

Showing Instances

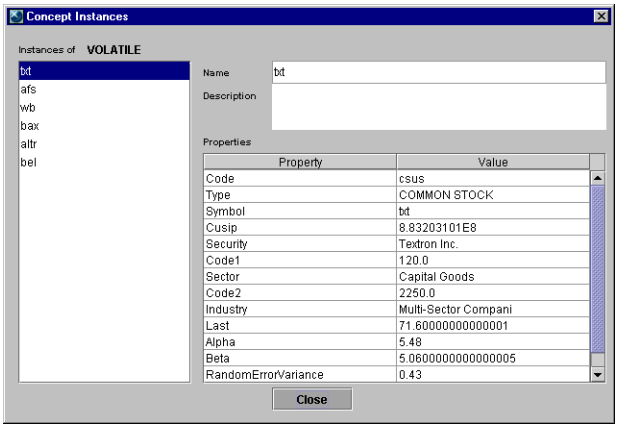
An *Instance* is a specific record associated with a Concept Item. For example, every record identified as a volatile stock is an Instance of the Volatile Business Concept.

1 Start the Business Concept Editor, if it is not currently open.

2 Select a Data or Business Concept.

3 Select Concept > Show Instances (or press CTRL+N, or click Show Instances on the toolbar). The Concept Instances dialog displays. All Instances associated with the selected Concept Item display in the Instances list, while the Name, Description, Properties, and Values for the first Instance display in the Detail panels.

Figure 5-40. Concept Instances



4 (Optional) To view information for each listed Instance, select an item from the Instances list.

5 Click Close to exit the Concept Instances dialog.

Deleting a Concept Item

When you delete a Concept Item, you are removing it from the ontology. This may affect any predictive models, rules, or concept relations that were previously saved to the Knowledge Base.

1 Start the Business Concept Editor, if it is not currently open.

2 Select a Concept Item.

3 Select Concept > Delete (or click Delete Concept on the toolbar). The Delete Concept Confirmation dialog displays.

Figure 5-41. Confirm Concept Delete

4 Click Yes. The item is removed from the Concept list.

If you delete a Parent Concept, all its Children Concepts are also deleted.

Chapter

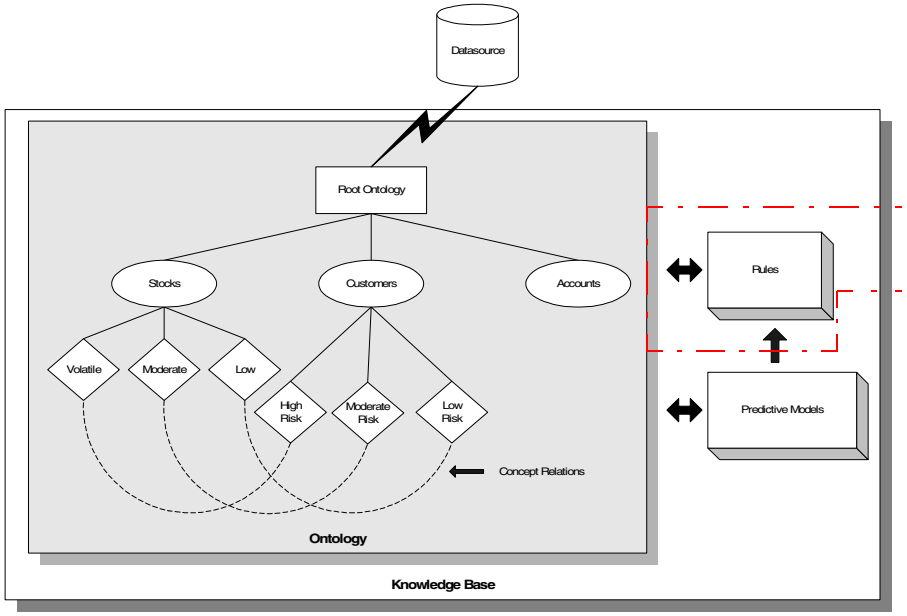
6

Defining Rules

This chapter provides step-by-step instructions for defining rules. *Rules* are logical statements that describe how an object (person, place, thing, or event) will behave in a particular situation. They are the means for "teaching" the computer which datasources to query, patterns to analyze, implications to recognize, and real-time recommendations to make.

The Black Pearl Knowledge Broker contains a Rule Editor that enables you to define rules that describe two things: how your business typically responds to individual transactions and what situations will require different responses to seemingly similar transactions.

Figure 6-1. Adding Rules to the Knowledge Base



There are three basic types of rules:

- *Legal rules,* which are mandated by a federal, state, city, or other warrantable agency (including SEC, OSHA, FCC, EPA, and unions), cannot be broken unless the issuing agency amends the rule. For example, an SEC rule mandates a "quiet period" before a company's initial public offering.
- *Domain-specific rules,* which are defined by a specific business or field of knowledge (including physics, chemistry, and law), can also not be broken unless amended by the issuing domain. For example, a brokerage house may stipulate that stockbrokers employed with the firm for less than six months may not offer speculative stock advice.
- *Common-sense rules,* which include opinions, hunches, and best practices, can be broken if circumstances warrant. For example, a brokerage house may institute a "best practices" policy whereby speculative stocks will not be recommended to a person who is 70 years old and living on a fixed income.

The Black Pearl Knowledge Broker currently supports a single category of rules:

- *Constraint Satisfaction*, which enables you to identify the rules, concepts, and data that should be processed by the Black Pearl Knowledge Broker.[1]

Each rule consists of three parts in the basic form:

```
if <SUBJECT> then <VERB > <DIRECT OBJECT>
```

or, simply, "if X is true, then perform action Y on Z".

To say that "X is true" actually means to find within the ontology a greater-than-zero number of items that can be described by X. For example, if X is "customer's age is less than 45", then there must be at least one customer satisfying that condition for the rule to activate. Thus, a *subject* that is "true" can be also be described as being "non empty".

1. Future releases of the Black Pearl Knowledge Broker will support these additional subcategories of rules: *Sequencing*, which enables you to specify that some rules must be activated before another rule may be processed. *Duration*, which enables you to specify that a rule is valid only for a specified time period. *Control*, which enables you to include both global and local rules. *Priority*, which enables you to specify rules in terms of degrees of importance. Global rules are defined for the entire organization, while local rules are defined by an individual office, department, or person.

More complex rules are of the form:

```
IF
          <concept>
            such that {<property_value>
                       <operator>
                       <comparison_value>}
          AND
                      {<concept.property_value>
                       <operator>
                       <concept.property_value>}
THEN
     <action>
          <concept>
            such that {<property_value>
                       <operator>
                       <comparison_value>}
          AND
                      {<concept.property_value>
                       <operator>
                       <concept.property_value>}
```

where `<concept>` is a concept in the ontology

`<property_value>` is a property of the concept, retrieved from the ontology

`<operator>` is one of: `=`, `!=`, `<`, `>`, `<=`, `>=`

`<comparison_value>` is any arbitrary value

`<action>` is currently one of: `recommend buy`, `recommend sell`, `notify`, `predict`, `do not recommend buy`, `do not recommend sell`, `do not notify`, `do not predict`

For example:

```
if customer's age is less than 45 then recommend buy adventure sports
```

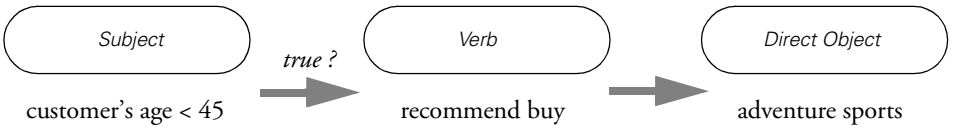
In this example, the *subject* (sometimes known as the antecedent) is a Business Concept (customer), the *property* is a property of the customer Business Concept (age), and the *property value* is a filter parameter (less than 45).

The *direct object* (sometimes known as the consequent) is a Business Concept (adventure sports).

The *verb* (recommend buy) acts on the direct object if the contents of the *subject* are true, or non-empty.

Another way to think about this is that the *direct object* (adventure sports) receives the action of the *verb* (recommend buy). The *verb* acts only if an assertion or operation in the preceding *subject* has been tested and found to be true, or non-empty.

Figure 6-2. Verb Acting on Direct Object

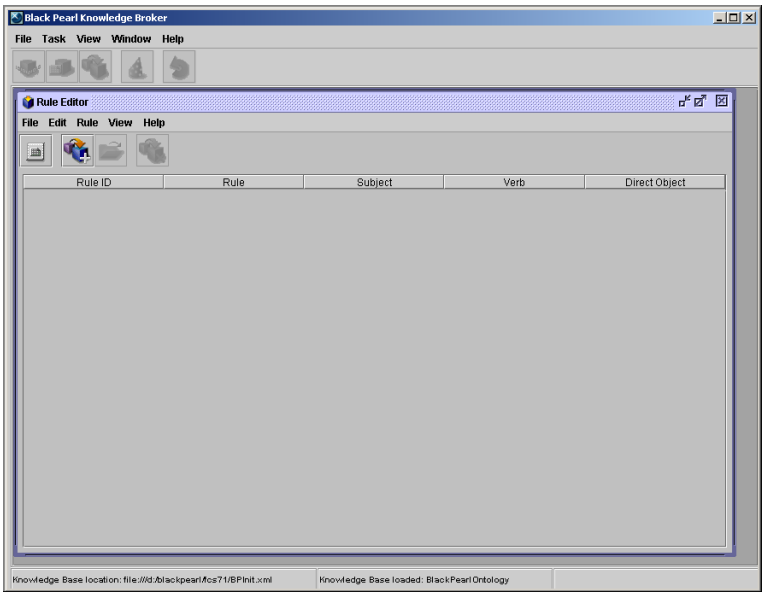


If the customer's age is greater than or equal to 45 then this rule will not activate.

Defining Rules

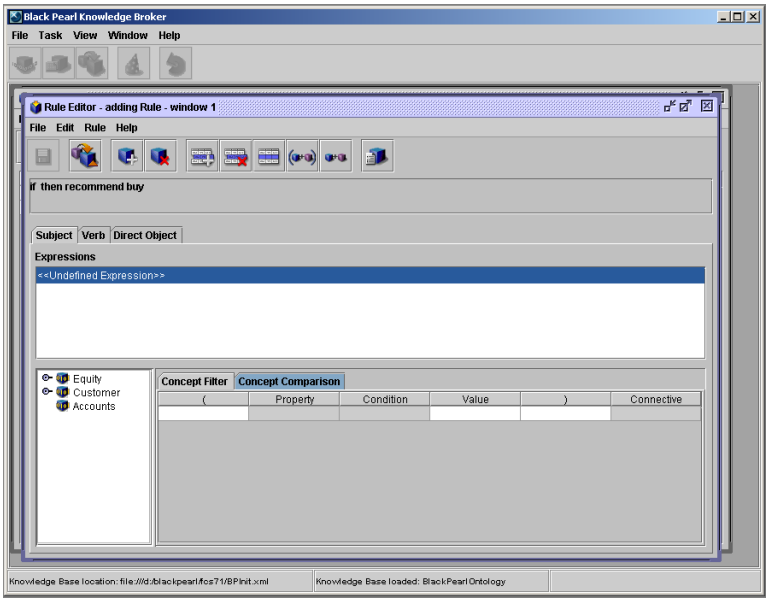
1 Start the Rule Editor, if it is not currently open.

Figure 6-3. Rule Editor



2 Select Rule > Add Rule (or click Add Rule on the toolbar). The Rule Definition Editor displays.

Figure 6-4. Rule Definition Editor



3 The Rule Definition Editor initially displays in Subject mode (with the **Subject** tab highlighted). The *Subject* is the qualifying condition that must be true for the rule to activate.

The top panel displays the default `if then recommend buy` statement. At all times, this top panel displays all the components of the rule you are editing.

Initially, `<<Undefined Expression>>` displays highlighted in the Expressions panel with the Subject tab highlighted. This indicates that you are defining components for *Subject* part of the rule.

4 The components of the Subject (and the corresponding Direct Object) are either Concept Filters or Concept Comparisons, or any combination of both components joined by the `AND` connective. You define these components as follows:

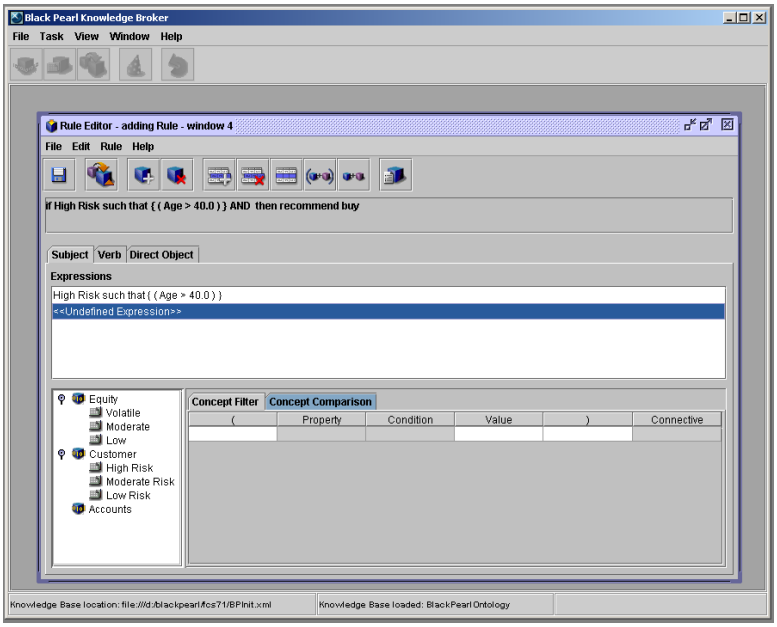
a (Optional) You can test the values of Concept Item Properties by adding specific filter conditions. These conditions must test true for the rule to activate. (See *Adding a Concept Filter on page 124*.)

b (Optional) You can compare the values of pairs of Concept Item properties by using Concept Comparisons. These comparison statements must test true for the rule to activate. (See *Adding a Concept Comparison on page 128*.)

5 When you begin to add Concept Filters or Concept Comparisons, the expressions you choose replace `<<Undefined Expression>>`.

6 (Optional) Select Rule > Add Expression when you have finished specifying a Concept Filter or a Concept Comparison and you want to add another expression. The expressions are automatically joined by the `AND` connective in the top panel to create combined *Subject* (or *Direct Object*) components:

Figure 6-5. Adding a New Expression

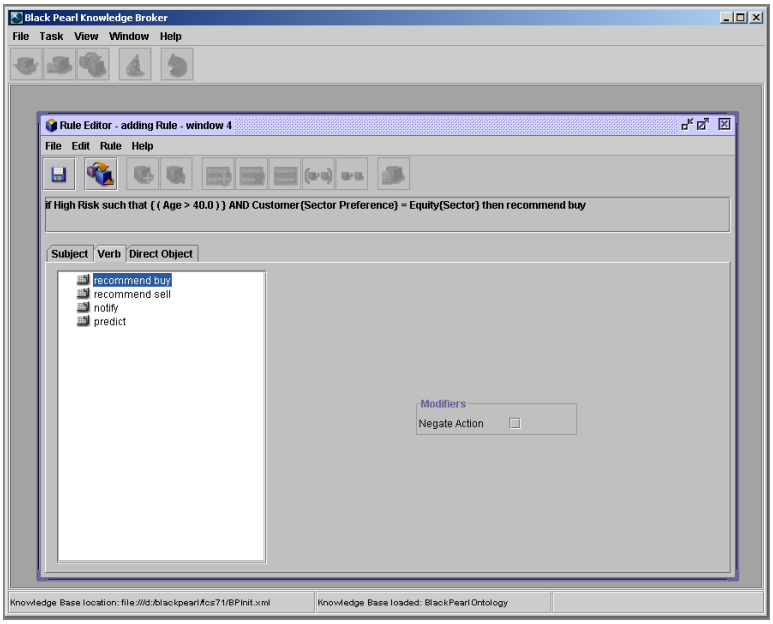


7 (Optional) Select Rule > Delete Expression to delete a selected expression.

8 Click the Verb tab to select an action. The options displayed are `recommend buy`, `recommend sell`, `notify`, and `predict`. Click any of the actions in the left-side

panel to change the Verb. The display in the top panel changes to reflect your selection.

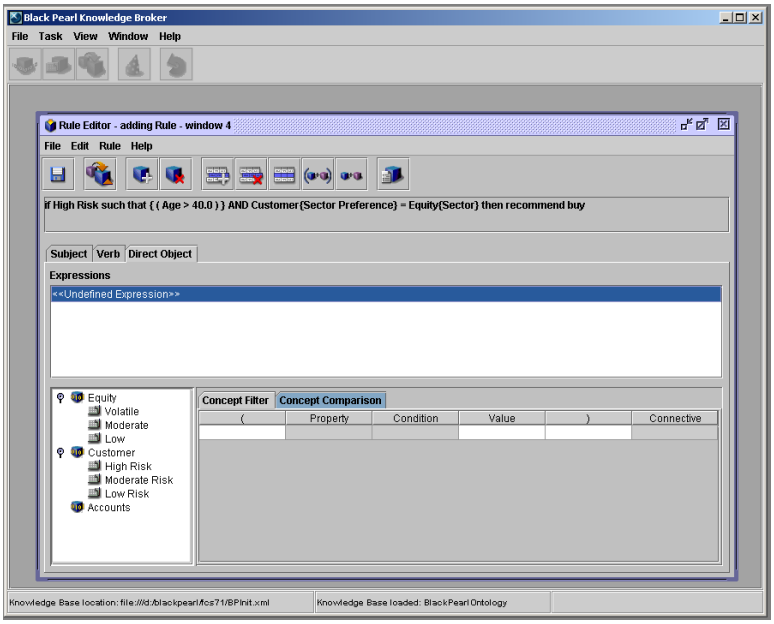
Figure 6-6. Selecting the Verb



9 (Optional) Select Negate Action to produce the verbs `do not buy`, `do not sell`, `do not notify`, and `do not predict`. A check mark appears in the Negate Action box and the top panel display changes to reflect your selection.

10 Click the Direct Object tab to enter Direct Object mode where you select the ontology objects to be acted on by the *Verb*. The initial display is similar to the Subject mode display in Step 3.

Figure 6-7. Defining the Direct Object



Initially, `<<Undefined Expression>>` displays highlighted in the Expressions panel with the Direct Object tab highlighted. This indicates that you are defining components for the *Direct Object* part of the rule.

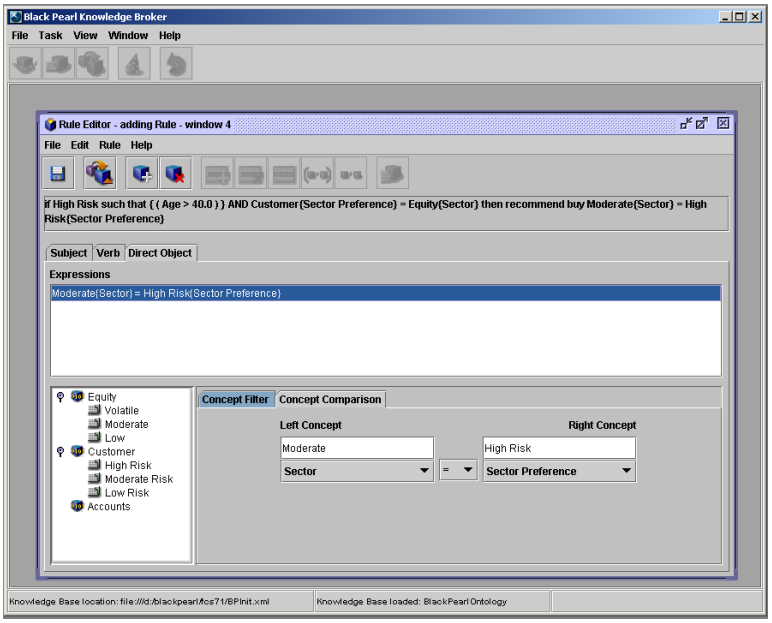
11 Creating the right-hand-side, *Direct Object* part of a rule uses the same Concept Filters and Concept Comparisons techniques as with the *Subject* part of a rule. You should follow Steps 4 through 9 to construct your Direct Object components.

The Verb applies the action to all ontology objects described by all the components of the Direct Object. A Concept Filter with too many conditions, therefore, might not be true for any ontology objects in the knowledge base. In this case, the Verb would not apply to any objects and the rule, although activated, would not return a recommendation.

You should exercise caution when constructing complex Direct Objects. For more instructions on rule manipulation, see *Creating Rules to Establish Marketing Objectives on page 198*.

12 The top panel displays your rule, with all Subject, Verb, and Direct Object components detailed.

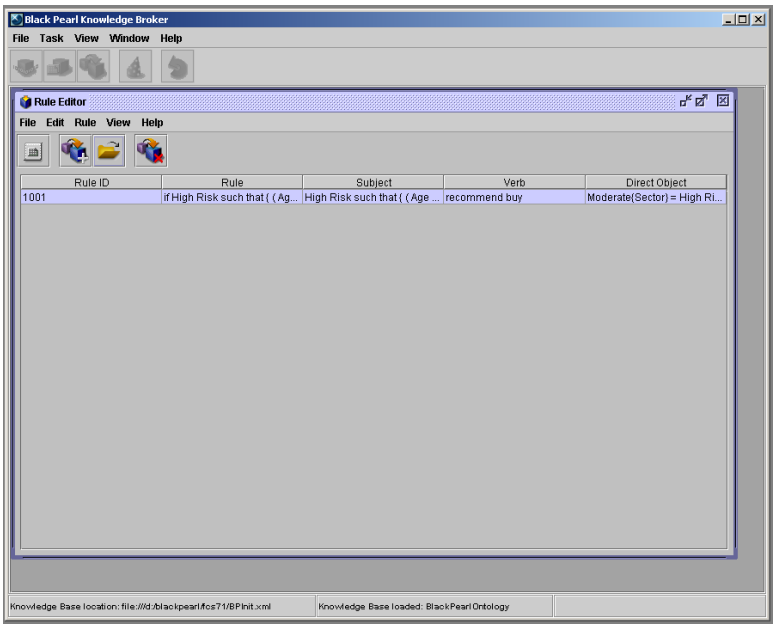
Figure 6-8. A Completed Rule in the Rule Definition Editor



13 (Optional) You can click any of the mode tabs (Subject, Verb, Direct Object) to add, delete, or modify your Expressions.

14 Select File > Close (or click the appropriate toolbar icon) to exit the Rule Definition Editor when you are finished defining your rule. The Rule Editor re-displays with your rule now displayed.

Figure 6-9. A Completed Rule in the Rule Editor



15 Select File > Save Rule (or click the appropriate toolbar icon) to save the rule to the Rule Editor.

Saving the rule within the Rule Editor does not save the rule to the knowledge base. You must first exit the Rule Editor and then select File > Save in the Black Pearl Launcher to save the rule to the knowledge base.

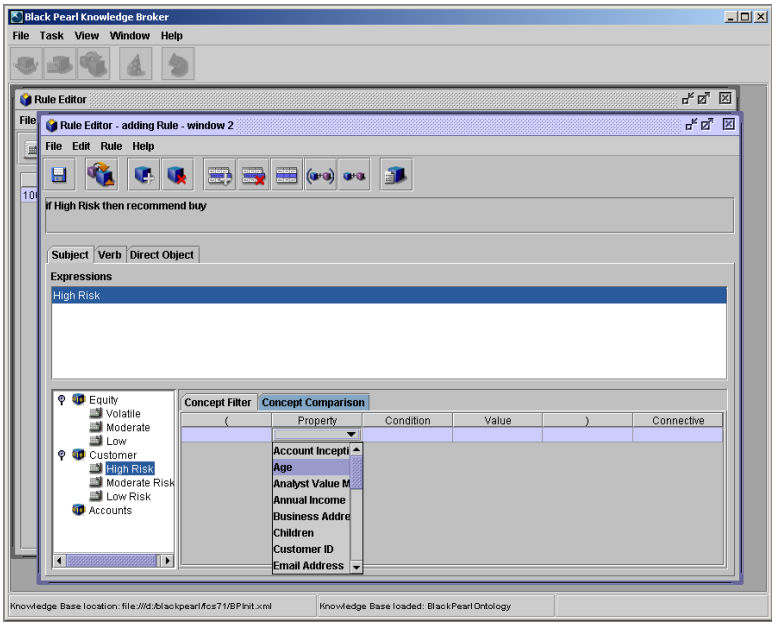
Adding a Concept Filter

A Concept Filter sets particular values and constraints on rule processing. Filters can be applied to *Subject* Concept Properties or to *Direct Object* Concept Properties.

You cannot apply filters to any of the Concept Properties during a Concept-Level Comparison. You can join together Concept-level comparison rule components and Concept filtered rule components using the Concept-level *and* connective.

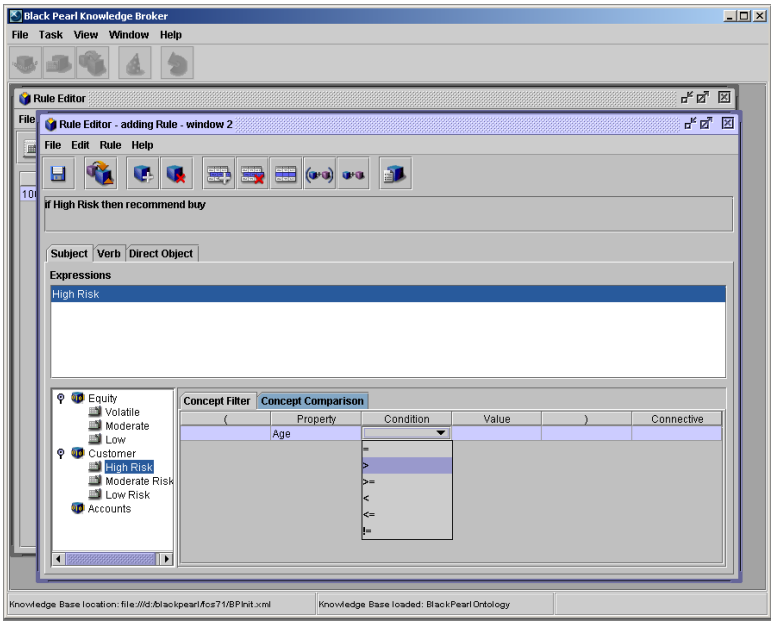
1. Select a Data Concept or Business Concept in the left-hand panel.
2. Click the Property field. When a Concept has been selected in the left-hand panel, a drop-down menu displays available Properties. Select the appropriate Property from the drop-down list.

Figure 6-10. Concept Filter Property Menu Drop-Down



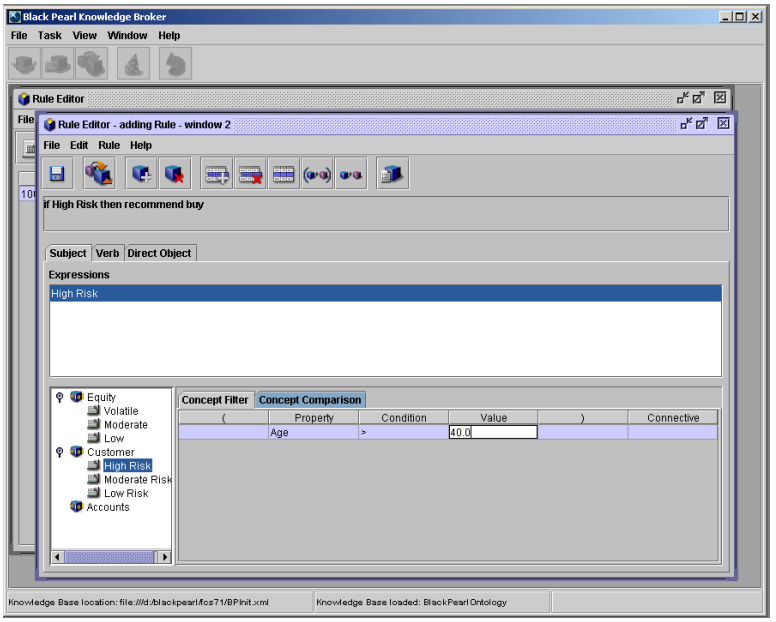
3 Click the Condition field, and select the appropriate Condition operator from the drop-down list.

Figure 6-11. Concept Filter Condition Menu Drop-Down



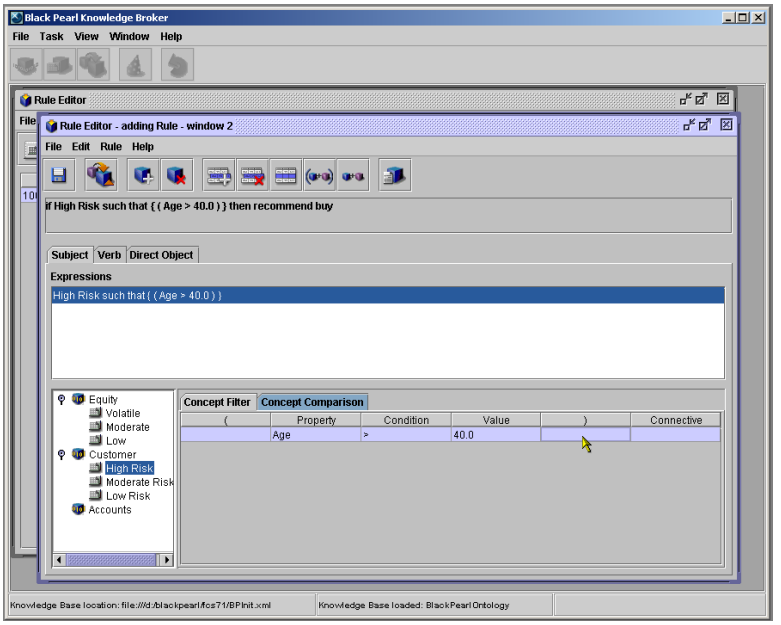
4 Click the Value field, and type the appropriate Value.

Figure 6-12. Concept Filter Value Field



5 Hit **`Enter`** to complete this filter row. The filter row displays in the Expression panel.

Figure 6-13. Concept Filter With Completed Expression



6 (Optional) If you require a more complex filter, select Rule > Add Filter Row and click the Connective field. Currently *and* is the only connective available in the drop-down list. Repeat steps 1, 2, and 3 for each filter that you want associated with the Rule.

The filter-level `and` connective requires that all filter conditions so joined be satisfied for the component to evaluate true during the rule processing phase. Selecting multiple filtered Properties within a Concept will introduce more constraints into that component of the rule.

On the *Subject* side of the rule, this will make the rule less likely to activate. On the *Direct Object* side of the rule, this will reduce the number of Property items onto which the *verb* will be applied. So even if a rule activates, with strict filters on *Direct Objects*, the number of items satisfying the filter conditions may decline to zero and your rule may not return any recommendations.

Adding a Concept Comparison

You can compare Concept Properties in the *Subject* or the *Direct Object* parts of a rule. The format is:

```
Concept_A.Property OPERATOR Concept_B.Property
```

Concept Comparisons are a powerful mechanism with two main functions:

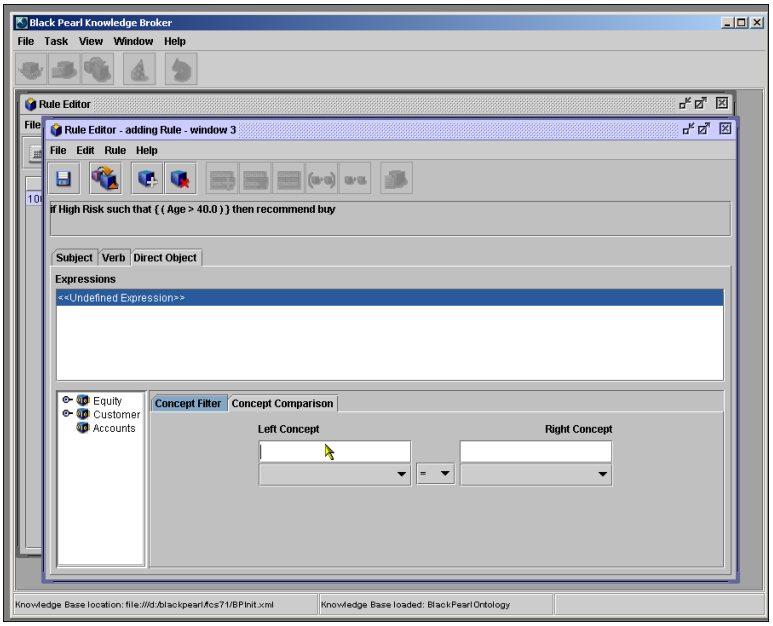
1. In the *Subject* part of a rule, you use them to compare the Properties of Concepts derived from one Data Concept against the Properties of Concepts derived from another Data Concept.
2. In the *Direct Object* part of a rule, you use them to perform the same comparison and then to apply the Verb to those ontology objects selected on by the left Concept of the comparison (if in fact ontology objects exist that satisfy the conditions of the comparison).

For example, to `recommend buy` specifically `Volatile Equities` of companies operating in a `Sector` preferred by your `High Risk Customers`, your *Direct Object* Concept Comparison expression format is:

```
Volatile.Sector = High Risk.Sector Preference
```

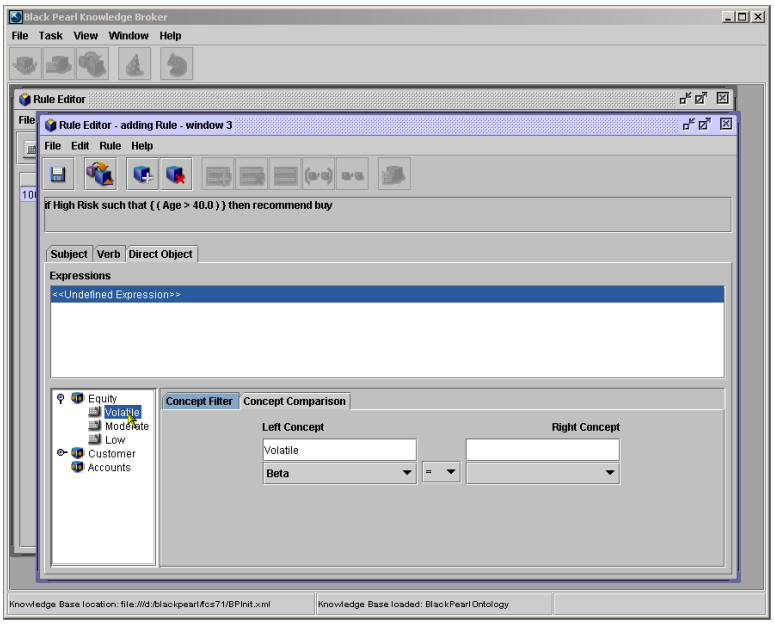
1. Click inside the Left Concept field to begin.

Figure 6-14. Concept Comparison: Left Concept Selected



2 Select a Data Concept or Business Concept in the left-hand panel. Your selected Concept displays in the Left Concept field and the initial Property available for comparison appears underneath this.

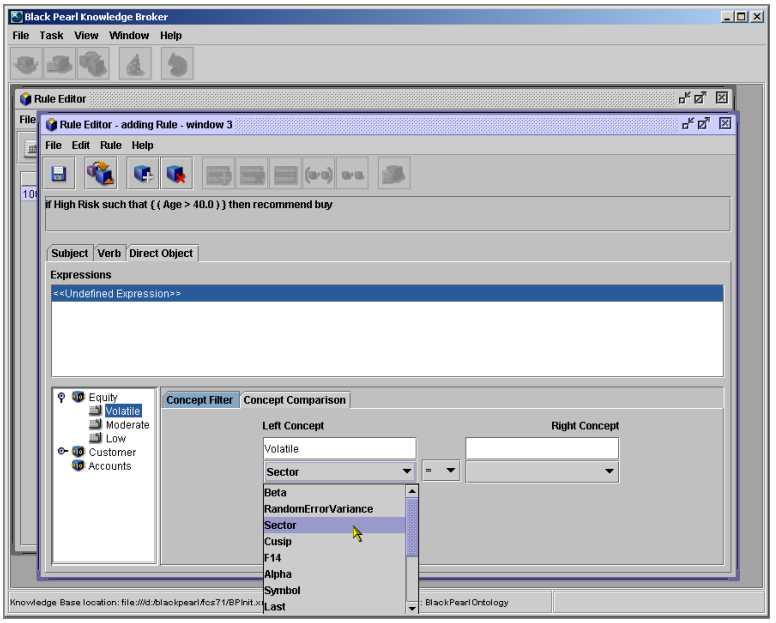
Figure 6-15. Concept Comparison: Business Concept Selected



3 Click the initial Property in the drop-down menu underneath the Left Concept field. A list of all Properties available for comparison appears. these correspond to the Data Concept Properties you defined in the Data Concept Editor (see *Adding a Data Concept Property on page 85*).

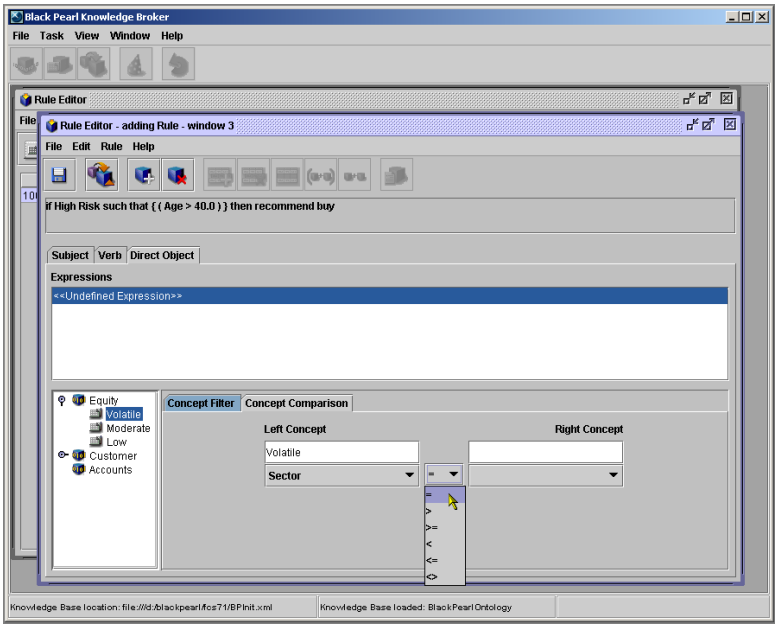
4 Select the Property you wish to compare from the list of available Properties in the drop-down menu.

Figure 6-16. Concept Comparison: Business Concept Property



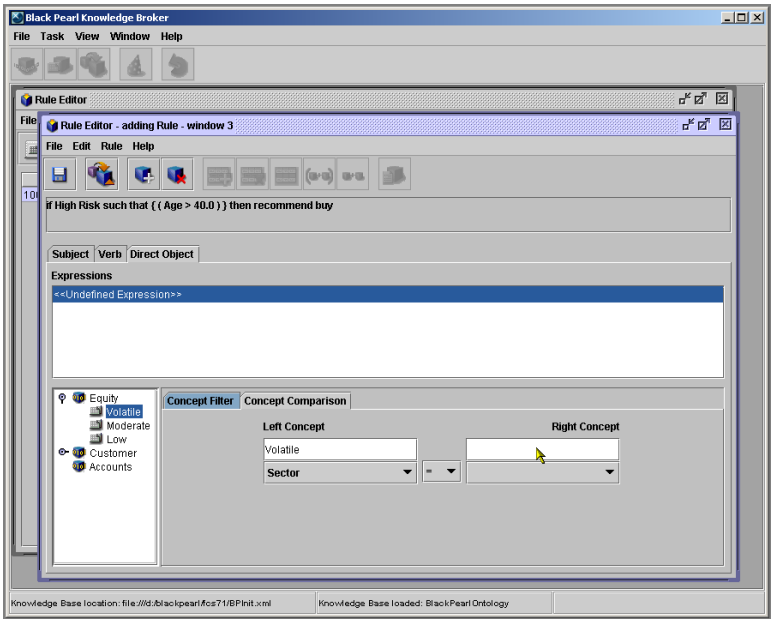
5 Click the Concept Comparison Operator drop-down menu between the Left Concept and Right Concept fields. The drop-down menu displays the available operators. Select your preferred comparison operator.

Figure 6-17. Concept Comparison Operator Drop-Down Menu



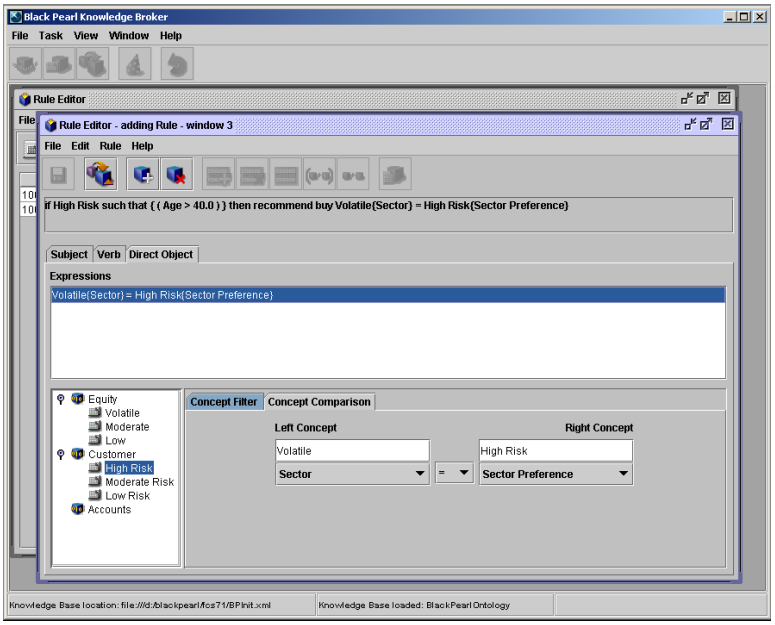
6 Click inside the Right Concept field. You are now ready to select the second Property of your Concept Comparison.

Figure 6-18. Concept Comparison Right Concept Selected



7 Repeat Steps 2, 3, and 4 above to select the Property for the Right Concept field. Your Concept Comparison is now complete.

Figure 6-19. Concept Comparison Completed



8 Select File > Close to finish editing this rule.

9 (Optional) If you require a more complex rule, select Rule > Add Expression or click the Add Expression icon. You can connect multiple Concept Comparisons (or Concept Comparisons and Concept Filters) together within the Subject or Direct Object parts of rule using the Concept-level AND connective.

The same warning applies to multiple Concept Comparisons as to Filter Conditions with multiple filter rows earlier.

Adding more Concept Comparisons to the *Direct Object* part of a rule can mean that even when a rule activates, there may be no ontology objects that satisfy the Concept Comparison conditions. In this case, even though the rule activates, the *Verb* acts on zero ontology objects and so has no effect.

Concept Comparison Operators

The Concept Comparison operators available are described in Table 6-1:

Table 6-1. Concept Comparison Operators

Operator	Description
=	Tests if Concept_A.Property ***is equal to*** Concept_B.Property.
>	Tests if Concept_A.Property ***is greater than*** Concept_B.Property.
>=	Tests if Concept_A.Property ***is greater than or equal to*** Concept_B.Property.
<	Tests if Concept_A.Property ***is less than*** Concept_B.Property.
<=	Tests if Concept_A.Property ***is less than or equal to*** Concept_B.Property.
!=	Tests if Concept_A.Property ***is not equal to*** Concept_B.Property.

Concept Comparison Numeric Operators

If the Properties being compared are numeric then all operators function (using sample Property values of `43.4` and `54.0`) as described in Table 6-2:

Table 6-2. Example of Concept Comparison of Numerical Properties

ConceptA. Property	Operator	ConceptB. Property	Result
43.4	=	54.0	False
43.4	>	54.0	False
43.4	>=	54.0	False
43.4	<	54.0	True
43.4	<=	54.0	True
43.4	!=	54.0	True

Concept Comparison Alphabetic Operators

If the Properties being compared are alphabetic, only the `==` and `!=` operators function. Applying any other operator will result in an undefined result and the Concept Comparison will not perform any operation. This is illustrated in Table 6-3 (using sample Property values of `High Risk` and `Low Risk`):

Table 6-3. Example of Concept Comparison of Alphabetic Properties

ConceptA. Property	Operator	ConceptB. Property	Result
High Risk	=	Low Risk	False
High Risk	>	Low Risk	***Undefined***
High Risk	>=	Low Risk	***Undefined***
High Risk	<	Low Risk	***Undefined***
High Risk	<=	Low Risk	***Undefined***
High Risk	!=	Low Risk	True

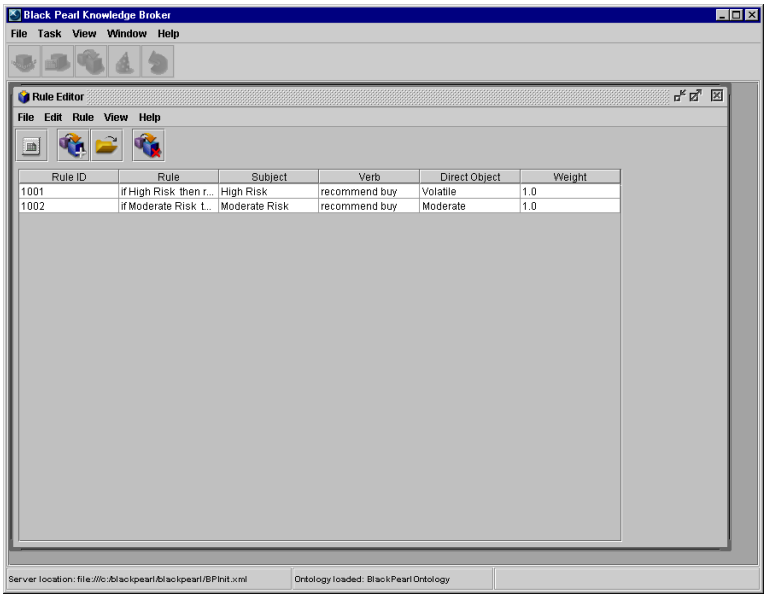
A Concept Comparison using an undefined operation does not produce either a `true` or `false` value. However, other Concept Comparisons or Concept Filters within the same rule can still operate and the rule can still activate (or not activate).

Editing a Rule

You can modify a rule after it is created.

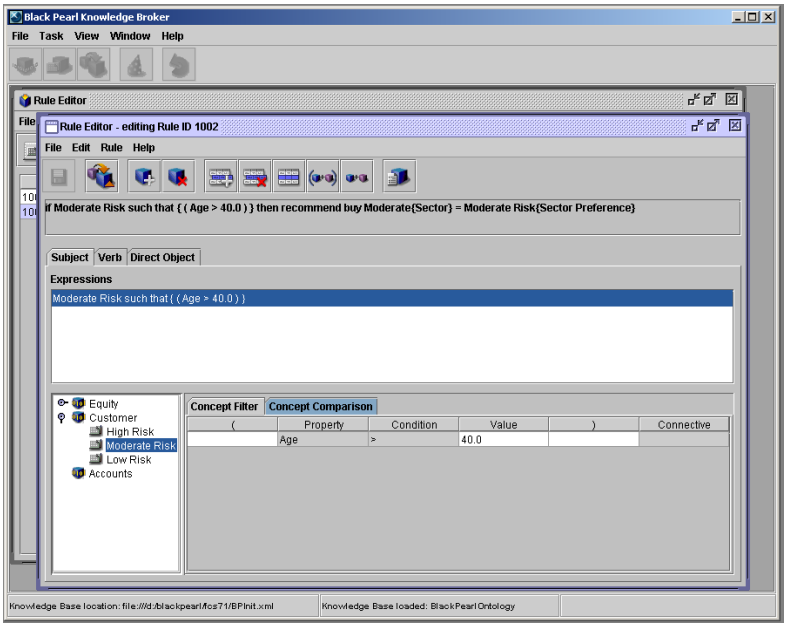
1 Start the Rule Editor, if it is not currently open.

Figure 6-20. Rule Editor



2 Select Rule > Edit Rule, or click Edit Rule on the toolbar. The Rule Definition Editor displays.

Figure 6-21. Edit Rule Editor



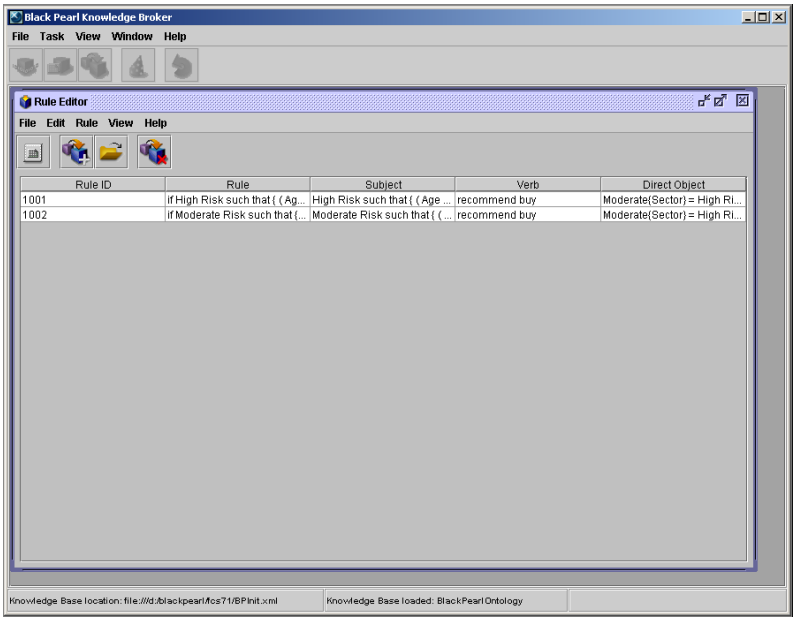
3 Complete the steps detailed in *Defining Rules on page 117*.

Deleting a Rule

You can delete a rule after it is created.

1 Start the Rule Editor, if it is not currently open.

Figure 6-22. Rule Editor: Deleting a Rule



2 Select the rule to be deleted

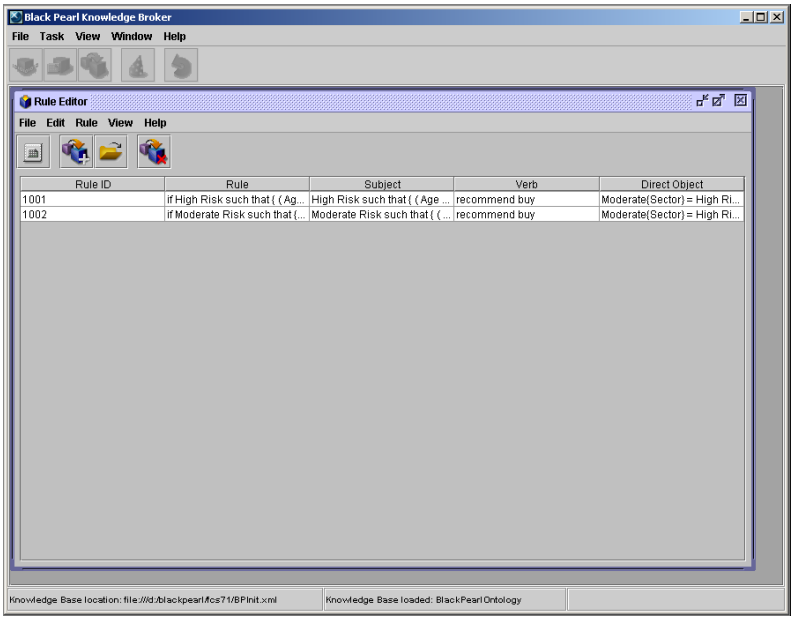
3 Select Rule > Delete Rule, or click Delete Rule on the toolbar. The rule is removed from the Rulebase.

Printing the Rulebase

You can delete a rule after it is created.

1 Start the Rule Editor, if it is not currently open.

Figure 6-23. Rule Editor: Printing a Rule



2 Select File > Print Preview (or click Print Preview on the toolbar).

Editing Rule Editor Columns

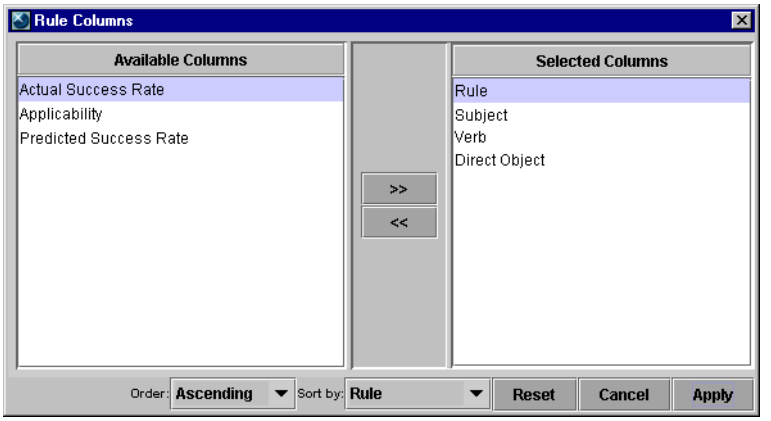
You can modify the number of columns that display in the Rule Editor by using the Edit Columns functionality.

1 Start the Rule Editor, if it is not currently open.

2 Select View > Edit Columns. The Rule Column Editor displays. The Available Columns list displays all currently defined column headings. The Selected Columns list displays all columns that currently display in the Rule Editor. The lower toolbar enables you to define sorting parameters.

You can sort rules in Ascending or Descending order,

Figure 6-24. Rule Columns Editor



3 Select an item from the Available Columns list.

4 Click >> to move the item to the Selected Columns list.

5 (Optional) To change the default Sort Order from Ascending, select Descending from the Order drop-down list.

6 (Optional) To change the default Sort By setting, select a different parameter from the Sort By drop-down list.

7 Click Apply to save the changes to the Rule Editor. The Rule Editor re-displays, with the newly added column displayed as the last column in the Rule Editor.

Clicking Reset will return the sorting parameters to the settings last saved to the Rule Editor.

Changing the Sort Order

You can modify the order in which the Rule Editor displays Rules.

1. Start the Rule Editor, if it is not currently open.
2. Select View > Edit Columns.
3. Complete one or both options:
   a. Select the appropriate item from the Order drop-down list.
   b. Select the appropriate item from the Sort By drop-down list.
4. Click Apply to save the changes to the Rule Editor. The Rule Editor re-displays, with the rules displayed according to the new sorting parameter.

Chapter 7

Defining a Predictive Model

A *predictive model* is a particular type of knowledge discovery process. *Knowledge discovery* is an iterative process that analyzes historical data to determine patterns, relationships, and trends that can be used to predict future values and generate real-time recommendations. The process depends on *models*—an abstract conception of a portion of reality—to uncover these patterns and relationships. A predictive model consists of three objects:

- Input values, known as the *independent* or *predictor variables*, that consist of a sample of your raw data.
- An *algorithm* that is applied to the independent variable to discover patterns and relationships.
- An output value, known as the *dependent* or *target variable*, that consists of predicted values for other data. The output value is the foundation for real-time recommendations.

Predictive modeling is a method for "learning by example" in that existing data is used to determine likely trends that can then be tested against real-world examples.

The Black Pearl Knowledge Broker includes a Knowledge Discovery Center that enables you to select the concepts, attributes, dataset sample size, sample methods, and other settings to be used in the predictive modeling process. Currently, the Black Pearl Knowledge Broker supports two types of predictive models: Decision Tree and Multi-layer Perceptron (MLP).

This chapter provides step-by-step instructions for defining a predictive model.

Why Use a Predictive Model?

The Knowledge Discovery Center helps you to create models of the real world that incorporate relevant Data and Business Concepts stored in your specific ontologies into Datasets. These Datasets concern customer demographics and histories, marketing and demographic information, corporate transactions and databases—for example, credit bureau information. The purpose building a model is to discover patterns and relationships in your data and codify this knowledge into descriptions that can classify new data advantageously.

Classification is the essence of prediction. Consider a brokerage firm that deals in financial securities and wants to predict if a particular bond would be a good or bad investment. There is already a mass of information collected regarding previous bond performance. Somewhere within this data there should be attributes or characteristics that identify which bonds are good or bad investments. These attributes are known as *predictor* or *predictive* attributes.

The predictive attributes available in the Knowledge Discovery Center will help you build models that classify your data into mutually exclusive groups. In the bonds performance example, one of these groups should contain as many as possible of the bonds that turned out to be good investments while the other should contain those which proved to be bad investments.

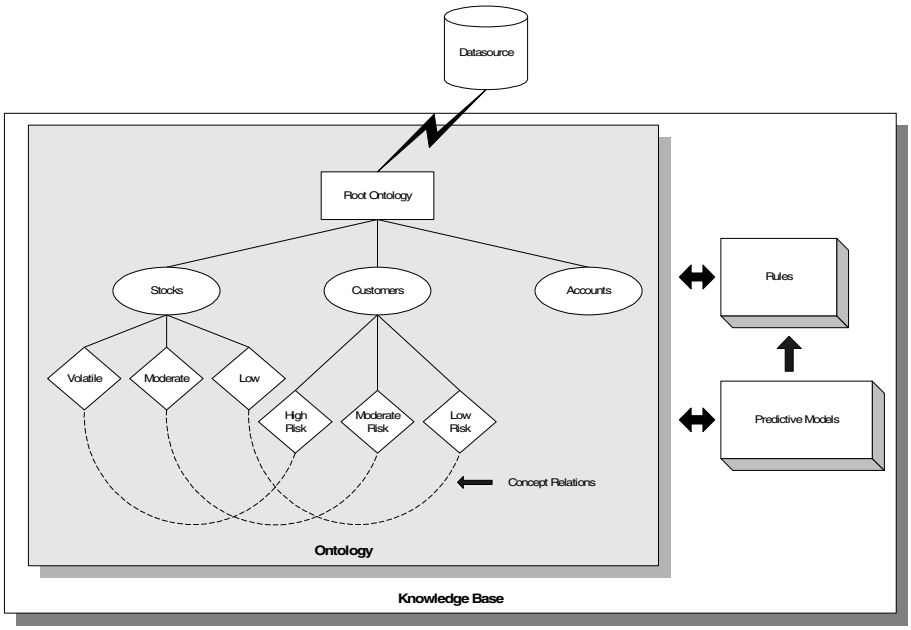
The Knowledge Discovery Center predictors discover internal rules (similar in principle to those you defined manually in the Rule Editor) to perform this classification. Predicting the value of potential bonds simply involves using these classification rules to decide in which group a potential bonds belongs. The Knowledge Discovery Center predictive models learn by example, using previous events and data to guide their predictions and are capable of producing new knowledge and recommendations that you can implement.

Defining a Predictive Model Dataset

Defining a predictive model is a multi-step process:

- *Defining Selected Properties*
- *Defining Input Attributes*
- *Creating a Dataset*
- *Updating the Dataset*
- *Updating Data Visualization*
- *Defining Settings*

Figure 7-1. Predictive Model Knowledge Base



Defining Selected Properties

The optimal way to discover knowledge within your ontology is to approach the Knowledge Discovery Center with a specific idea of the relation between your business objectives and the different concepts available within your ontology. You should then identify those concepts within your ontology that are relevant to the kind of predictions you want to make and incorporate them within your model.

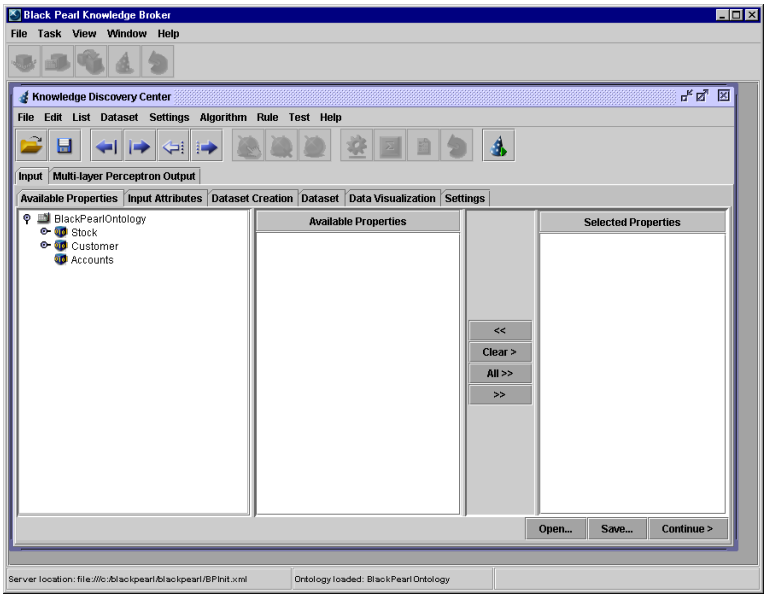
For example, a brokerage firm may want to screen new customers to determine their future investment profile on a scale of conservative, moderate, active, and venturesome. Based on their historical investment activities, the brokerage firm can assign its existing customers to one of these four profile categories.

This data is calculated and stored as the Like Investor Pool Property for the Customer Data Concept in the DemoParker ontology.

Alternatively, brokerage firms may want to offer additional tax consulting services. A reasonable assumption would be that the net tax bracket of customers can be predicted by their annual incomes. In that case, it is reasonable to create a predictive model that incorporates only those characteristics (represented in the ontology as Data Concept Properties) that help answer your formulated question, while temporarily disregarding other characteristics as non-relevant. The Data Concept Properties selected are known as the Selected Properties, and server as Input Attributes for the predictive models.

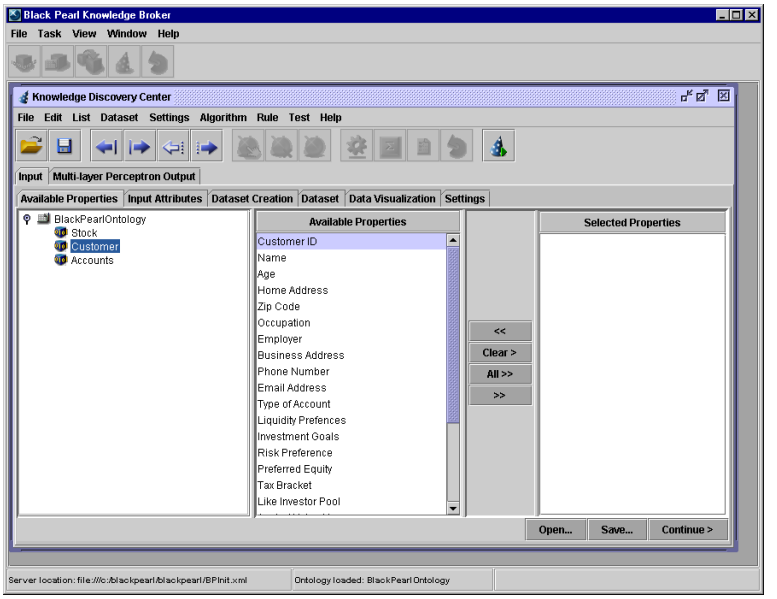
1 Start the Knowledge Discovery Center, if it is not currently open. The Data and Business Concepts currently defined within your ontology displays as a hierarchical tree in the left-hand panel.

Figure 7-2. Knowledge Discovery Center



2 Select a concept from the ontology list. A list of Properties assigned to that concept displays in the Available Properties panel.

Figure 7-3. Available Properties Editor

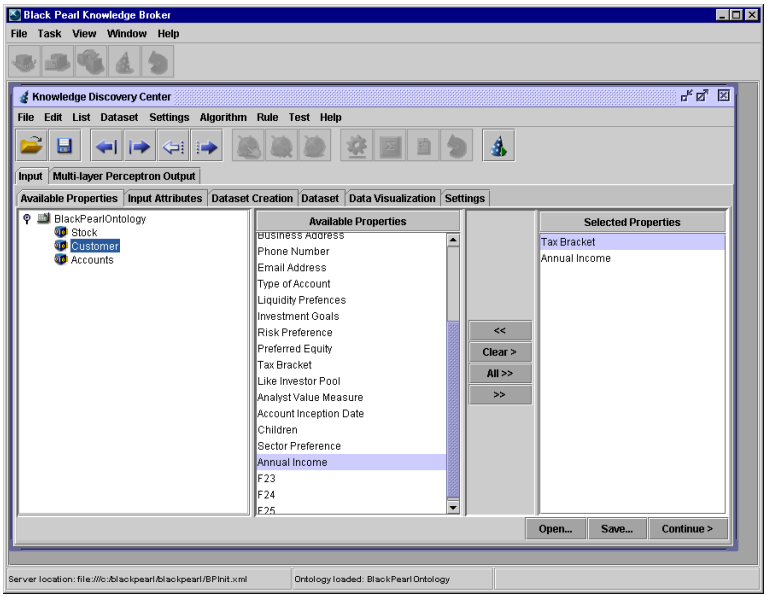


3 Complete one of the following:

- **a** Select an item from the Available Properties list. Click >> to move the item to the Selected Properties list. Repeat for each attribute that you want to move to the Selected Properties list. (You may select List **>** Move Right, or click the Move Item To Right button on the Toolbar.)
- **b** Press CTRL and select one or more non-consecutive items from the Available Properties list. Click >> to move the selected items to the Selected Properties list.
- **c** Press SHIFT and select one or more consecutive items from the Available Properties list. Click >> to move the selected items to the Selected Properties list.
- **d** Click All >> to move all Available Properties to the Selected Properties list. (You may select List **>** Move All Right, or click the Move All Items To Right button on the Toolbar.)

If you want to remove an item from the Selected Properties list, click <<, select List > Move Left, or click the Move Item to Left button on the Toolbar. If you want to remove all items from the Selected Properties list, click the Clear, select List > Clear All Choices, or click the Clear All button on the Toolbar.

Figure 7-4. Properties Selected



4 (Optional) Click Save to save the Selected Properties to the Knowledge Discovery Center.

Defining Input Attributes

The Selected Properties become attributes of the model, but not all attributes are created equal. The best attributes to define as predictors within the model (or Input Attributes) are known as independent attributes. These are characteristics of the real world (included within the ontology as Data Concept Properties) that vary independently of each other. The sum of their interactions creates and predicts the behavior of the customer or financial security or object. Some attributes are really independent attributes in disguise and you should try to avoid selecting them as Input Attributes to avoid clutter.

For example, suppose a broker wants to predict a potential customer's income. A reasonable assumption is that this can be predicted from their ZIP code, because certain locations tend to attract more affluent residents than others. Is it worth also including their land-based phone number to see if this improves the accuracy or reach of the predictive model? Probably not, because phone numbers tend to be assigned on a geographic basis because of the way the phone network is built.

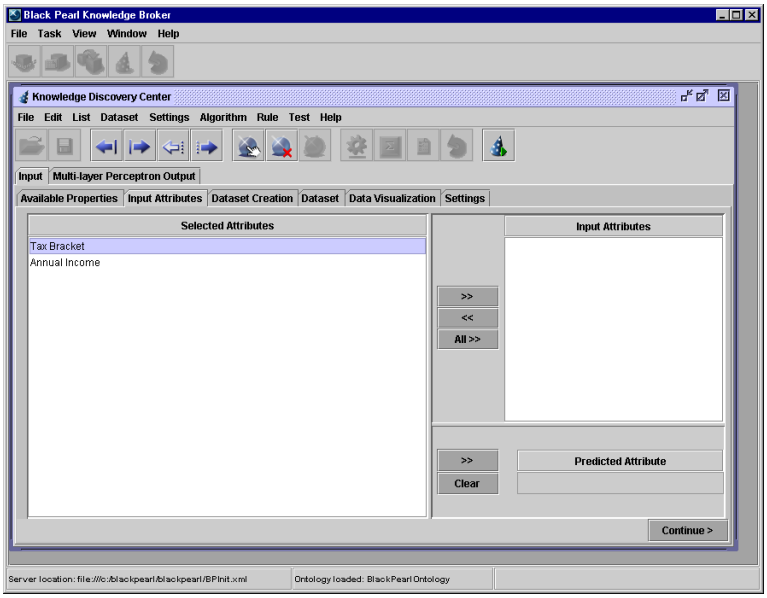
So phone numbers are dependent on geographic location, just like ZIP codes. Including either ZIP codes or telephone numbers within the model would be sufficient to include geographic location within the predictive model as an independent attribute. Avoiding these unnecessary, disguised attributes requires maintaining a clear understanding of the source of the Data Concepts and how they inter-relate with each other.

The aim of all this activity is to predict some behavior or characteristic from the Input Attributes. That is, to discover if and by how much a Predicted Attribute is dependent on the selected Input Attributes. For example, is the Like Investor Pool behavior dependent on Income and Risk Preference?

Selecting Income and Risk Preference as Input Attributes, and Like Investor Pool as Predicted Attribute will identify if Like Investor Pool is dependent on the two inputs and therefore allow it to be predicted. Similarly, selecting Annual Income as the Input Attribute and Tax Bracket as the Predicted Attribute will show exactly how customers' yearly incomes influences the tax bracket they find themselves in.

1 Click Continue (or click the Input Attributes tab). The Input Attributes Editor displays. You use this Editor to select the Input Attributes and Predicted Attribute, which are subsets of the Selected Properties.

Figure 7-5. Input Attributes Editor



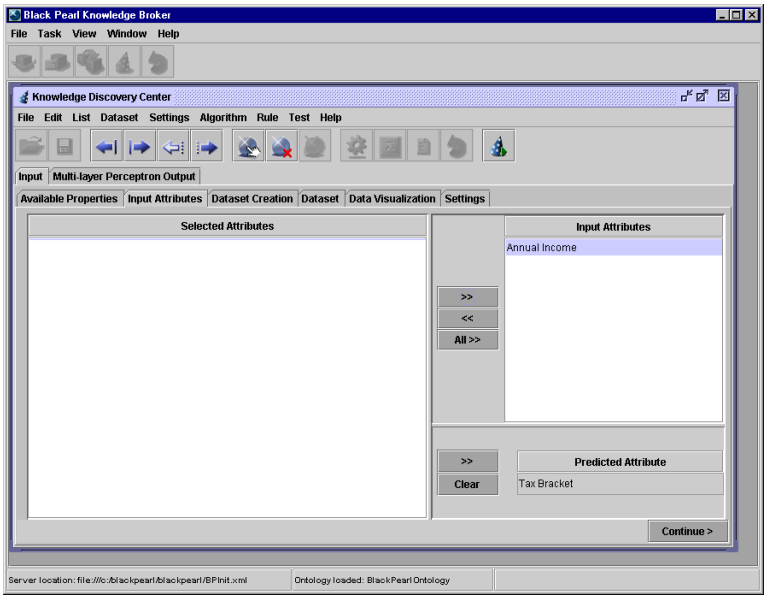
2 Select an item from the Selected Properties list. Click the Predicted Attribute >> to move this item to the Predicted Attribute section.

You must select a Predicted Attribute before proceeding to the Dataset Creation Editor.

3 Complete one of the following options:

- **a** Select an item from the Selected Properties list. Click Input Attributes >> (or select List **>** Move Right, or click the Move Item To Right toolbar icon) to move the item to the Input Attributes list. Repeat for each attribute that you want to move to the Input Attribute list.
- **b** Press CTRL and select one or more non-consecutive items from the Selected Properties list. Click >> to move the selected items to the Input Attributes list.
- **c** Press SHIFT and select one or more consecutive items from the Selected Properties list. Click >> to move the selected items to the Input Attributes list.
- **d** Click All >> (or select List **>** Move All Right, or click the Move All Items To Right toolbar icon) to move all Available Attributes to the Selected Properties list.

Figure 7-6. Input Attributes Selected

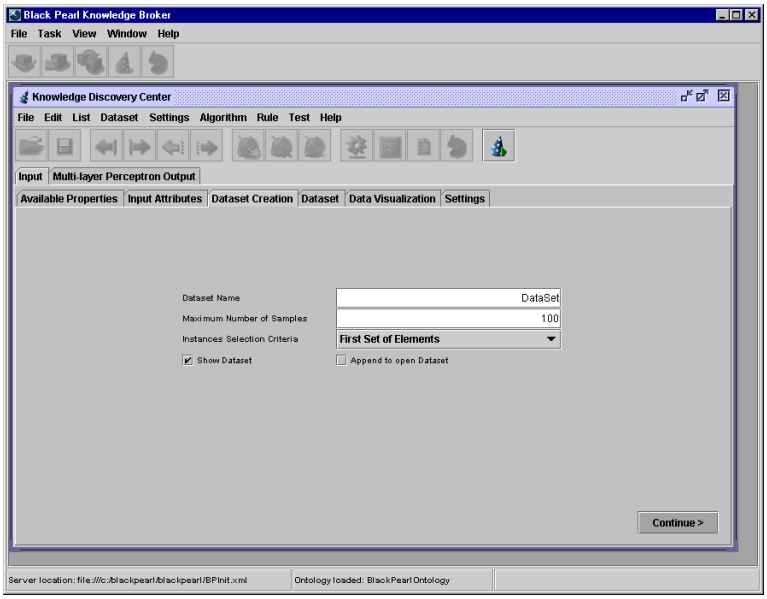


Creating a Dataset

Having selected your independent, Input Attributes and the Predicted Attribute that you hope will emerge as dependent on these inputs, you must create your Dataset, that is, the collection of values that will be used by the predictive model as examples to try to forecast the behavior of the Predicted Attribute.

1 Click Continue (or click the Dataset Creation tab). The Dataset Creation Editor displays. You use this Editor to generate a new dataset from the Selected Properties. A *dataset* is a set of instances matching the selecting attributes.

Figure 7-7. Dataset Creation Editor



2 Type the dataset name. By default, the dataset is named DataSet. Press Tab.

3 Type a number to set the Maximum Number of Samples. A *sample* is a set of Instances to be queried by the Black Pearl Knowledge Broker. The Maximum Number of Samples limits the maximum number of Instances available for the Input Attributes.

Because the amount of data to be analyzed can sometimes be very large, you can specify that the model selects and works with only a smaller, representative sample from the data. The number of samples selected influences the efficiency and the amount of time required for the predictive models to learn how to forecast successfully from the data.

4 Click the Instances Selection Criteria drop-down list and select a criteria.

- *First Set of Elements*: The Black Pearl Knowledge Broker will query the first set of Instances that conform to your designated Maximum Number of Samples. For example, if your Maximum Number of Samples is 100, the Black Pearl Knowledge Broker will query the first 100 Instances in your ontology.
- *Elements after Instance X*: The Black Pearl Knowledge Broker will ignore the first Instance in the ontology. For example, if your Maximum Number of Samples is 100, the Black Pearl Knowledge Broker will query Instances 2 through 101.

  Selecting either of the first two options when dealing with a large pool of potential data means is the simplest way to sample your data but may not produce optimal results.
- *Uniform Probability Distribution*: The Black Pearl Knowledge Broker will select a pool of samples using this statistical function, which from a pool of *N* data elements, selects a random sample with a probability of *1/N*. A single dice roll, for example, has a uniform probability distribution. The pool of potential data here would be {1,2,3,4,5,6} and the probability of any one number being selected is therefore 1/6 or 0.16667. If you think that the data you are sampling is best described by a uniform distribution, then you should select this option to produce an optimal selection.
- *Normal Probability Distribution*: This is a more complicated statistical function that is commonly known as a bell curve distribution. The exact shape of the bell curve can be described exactly using two statistical values known as the mean and the variance of your data. Selecting the correct values here will ensure that your sample selection more closely reflects the actual distribution of data. The formula itself is quite complex but importantly, many aspects of behavior in the real world that depend on the sum of the interactions of other, uniformly distributed independent, simpler behaviors are described by a bell curve. For example, rolling two dice is a more complex event than a single die roll and instead forms a normal distribution. The possible values are (1-6), (2-5), (3-4), (4-3), (5-2), and (6-1). The probability of rolling seven is therefore much higher than rolling three (possible values (1-2), (2,1) only).

You may have to construct various predictive models, using a variety of sampling methods, to find the most optimal one for your problem.

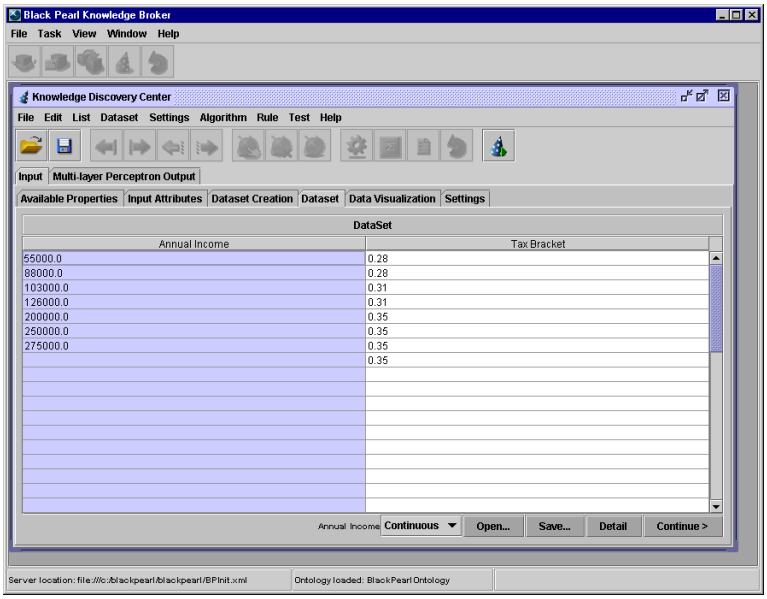
5 (Optional) Click the Show Dataset checkbox to *prevent* the display of the Dataset.

6 (Optional) Click the Append to open Dataset checkbox to add the current Dataset to a previously defined Dataset.

Updating the Dataset

1 Click Continue (or click the Dataset tab). The Dataset displays.

Figure 7-8. Dataset Editor



The predictive model can work with either discrete or continuous data. *Discrete* data can be categorized into only a few values. For example, Risk Preference can be described as discrete, since it can be categorized as Low, Moderate, or High. *Continuous* data consists of a range of values. For example, Annual Income contains data consisting of a range of numbers. Since the Decision Tree predictive model can only use discrete data in its calculations, you must use the Attribute Name drop-down list to break continuous data values into segments.

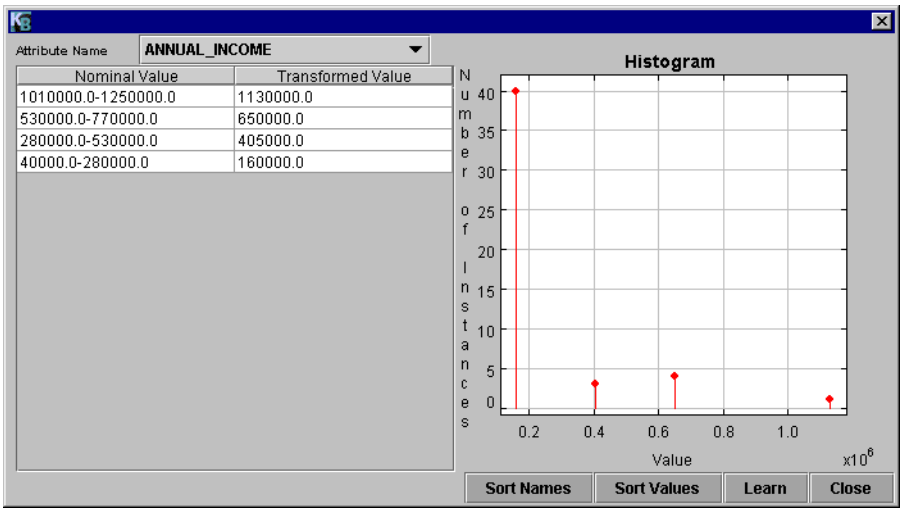
For example, to use Annual Income and Risk Preference to predict Like Investor Pool, you would transform the Annual Income into three segments or ranges. These would then represent, in your model, Lower Income, "Middle Income, and Higher Income customers.

2 (Optional) If you want to convert continuous data into discrete data, select a cell within a Dataset column. Select an appropriate value from the Attribute Name drop-down list (located in the lower portion of the screen).

3 (Optional) Click Save to save the Dataset to the Knowledge Discovery Center.

4 (Optional) Click Detail to view the Detail Editor. Complete one or more of the following options:

Figure 7-9. Dataset Details Editor



a Select an item from the Attribute Name drop-down list. The Histogram changes to visually reflect the relationship between the Instances and Values in the dataset. Each point in the Histogram represents an Instance that matches a particular Value.

b Click Sort Names to sort the Name column into Ascending order. (If you click Sort Names again, the column sorts into Descending order.)

c Click Sort Values to sort the Value column into Ascending order. (If you click Sort Values again, the column sorts into Descending order.)

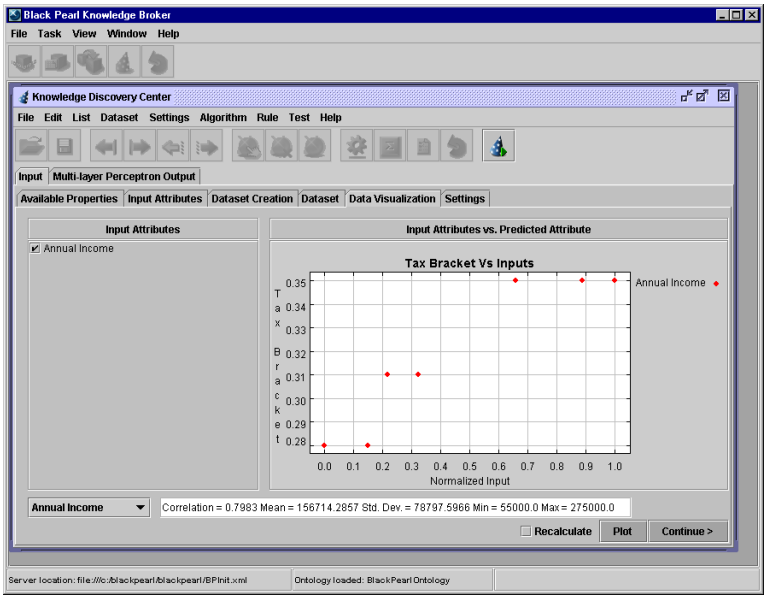
d Select an Input Attribute Instance and type a number in the Transformed Value cell. Click Learn. Unlike the Decision Tree, the MLP Input Attributes must be Continuous. The Knowledge Discovery Center transforms Discrete Input Attributes used in the MLP into Continuous Input Attributes. The results of the transformation are displayed in the Transformed Value cells.

Updating Data Visualization

Visual inspection of the Dataset can often yield valuable insights into “hidden” relations, trends, and biases within the data that you have been unaware of. Clusters and correlations of Input Attribute data can sometimes become visible and prompt you to go back and revise your input selection or the discretization methods you have selected.

1 Click the Data Visualization tab. The Data Visualization Editor displays. You use this Editor to view a visual representation of the correlation between each Input Attribute and the Predicted Attribute.

Figure 7-10. Data Visualization Editor



2 Complete one or more of the following options:

   a De-select one or more items in the Input Attributes list. Click Plot to update the graph. The graph updates the correlation between Input and Predicted Attributes, displaying values for only those items selected in the Input Attributes list.

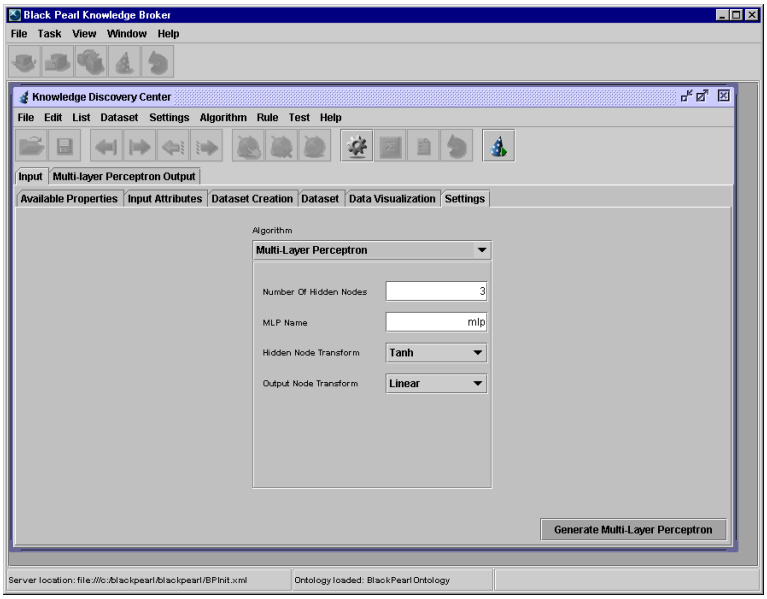
   b Click the Recalculate checkbox and click Plot to restore the default settings.

   c Select an item from the drop-down list (located in the lower right-hand corner) to view the Correlation, Mean, Standard Deviation, Minimum, and Maximum values in numerical terms. Correlation measures the relation between two or more variables and ranges from -1.00 to +1.00, with -1.00 indicating a complete negative correlation and +1.00 indicating a complete positive correlation. Obtaining 0.00 (or close to it) indicates a lack of correlation. A high magnitude of correlation is a strong indicator of a causal relationship between Attributes, but it is a relatively simple calculation and does not guarantee that a relationship exists. To establish this more fully, you need to create and train a predictive model.

Defining Settings

1 Click Continue (or click the Settings tab). The Settings Editor displays.

Figure 7-11. Settings Editor



2 Complete one of the following options:

a Select Decision Tree from the Algorithm drop-down list. Complete the procedures for *Generating and Updating a Decision Tree on page 161*. The Input Attributes for a Decision Tree are always Discrete and the Prediction Attribute is represented as a boolean (true or false) question. For example, if you have discretized, say, Annual Income and combined it with Risk Preference to see if you can predict Like Investor Pool, then you should use a Decision Tree to ask pose the question of whether the Like Investor Pool classification is *Moderate*, *True* or *False*? Similarly, you can pose the same question about *Active*, *Conservative*, or *Venturesome*.

The main advantages of the Decision Tree model is that it is very efficient, making its predictions quickly, and that its decisions can be expressed in simple IF-THEN rules in a sentence-like format similar to those Rules you defined in the Rules Editor. When you have to make categorical classifications (for example, "*this* customer likes *that* product and *that* product, will they like *this* product?") Decision Trees are the best model to choose.

b Select Multi-layer Perceptron from the Algorithm drop-down list. Complete the procedures for *Creating and Training a Multi-Layer Perceptron on page 167*. The MLP is based on a model of how the human brain learns knowledge using a great number of individual, simple cells known as neurons. The MLP uses these as nodes to map out patterns in the connections (if any exist) between the Input Attributes and the Predicted Attribute. Because of the way it functions, the MLP works best on continuous data. If, for example, you wanted to predict tax brackets of customers with different incomes, you would generate an MLP predictive model.

The main advantages of the MLP is that it copes well with "noisy" or hazy data and is a "black box" or hidden system (based on the presence of a huge number of internal interconnections) that can automatically form a connection between the Input Attributes and the Predicted Attribute. The internal model it has created can then be fed new values of Input Attributes and will output Predicted Attribute values to match. The MLP experiences a "graceful degradation" as the number of nodes is diminished. This allows its models to scale well across different datasets and different system configurations.

Its black box nature means that, unlike the Decision Tree, the MLP can't explain the exact nature of the connection and as a result can't produce easy-to-read rules. The MLP must trained—through an iterative process— to produce correct Predicted Attribute values.

Generating and Updating a Decision Tree

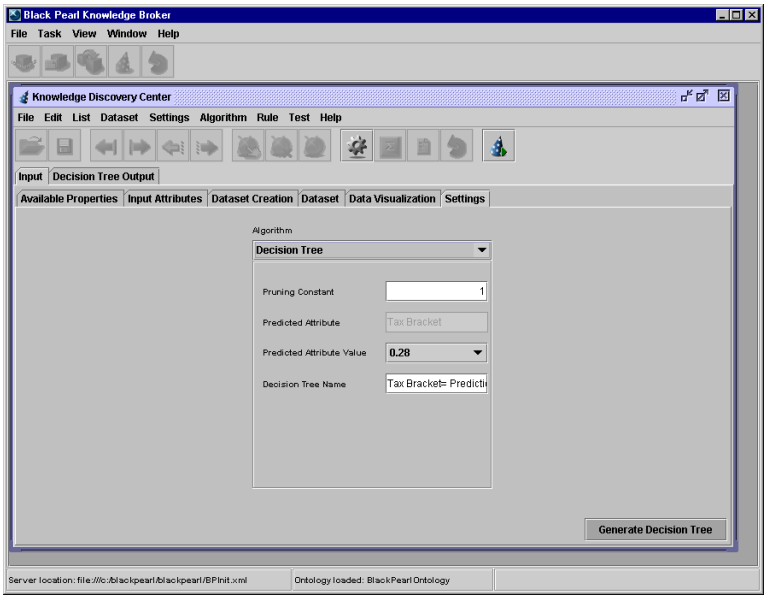
If you selected the Decision Tree algorithm in the Settings Editor, you must complete the following:

- *Setting Decision Tree Parameters*
- *Pruning a Decision Tree*
- *Updating Decision Tree Production Rules*
- *Testing the Decision Tree Predictive Model*

Setting Decision Tree Parameters

Click the Settings tab if the Settings Editor is not currently open.

Figure 7-12. Decision Tree Parameters



1. Type the Pruning Constant (value - 0 or 1). Low values reduce pruning severity, while values nearer to and including 1 increase the pruning severity. Press Tab.
2. Select an item from the Predicted Attribute Value drop-down list. Press Tab.
3. Type a name in the Decision Tree Name text field. By default, the Decision Tree Name is identical to the Predictive Attribute Value Click Generate Decision Tree.

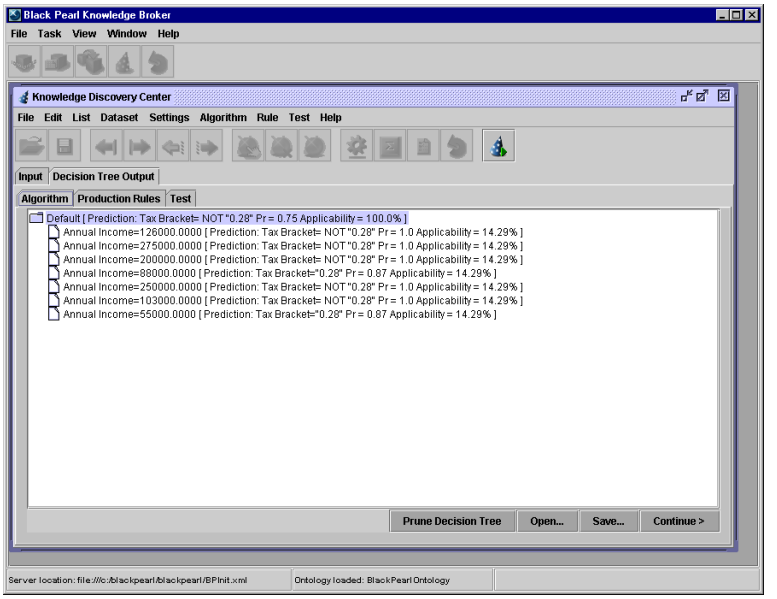
4 Click Generate Decision Tree >. The Algorithm Editor displays. You use this Editor to view and prune the Decision Tree.

Pruning a Decision Tree

A Decision Tree contains numerous branches that represent choices and decisions, some of which may be based on noisy data. For example, if you had 100 customers who lived in ZIP 12345 and who liked risky investments and a single customer who lived in ZIP 12345 but liked conservative investments, you could cut down on the size of your Decision Tree by omitting this single customer from the data. Or there may not be enough statistical correlation between certain sets of data to justify the additional complexity or "bushiness" in the Decision Tree.

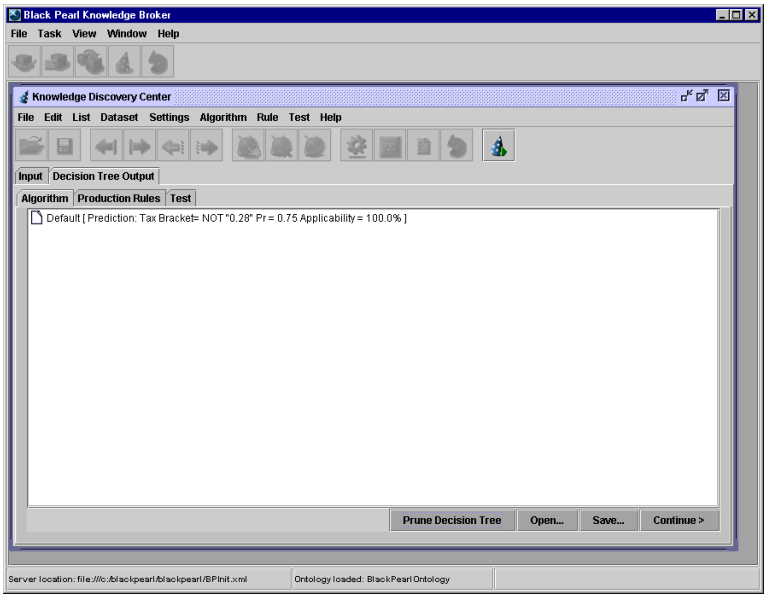
Selecting Pruning enables an internal statistical test that measures the significance of the data and simplifies the Decision Tree by removing the least important branches.

Figure 7-13. Decision Tree Algorithm Editor



1 Click Prune Decision Tree. The KDC removes all redundant rules to improve clarity.

Figure 7-14. Pruned Decision Tree

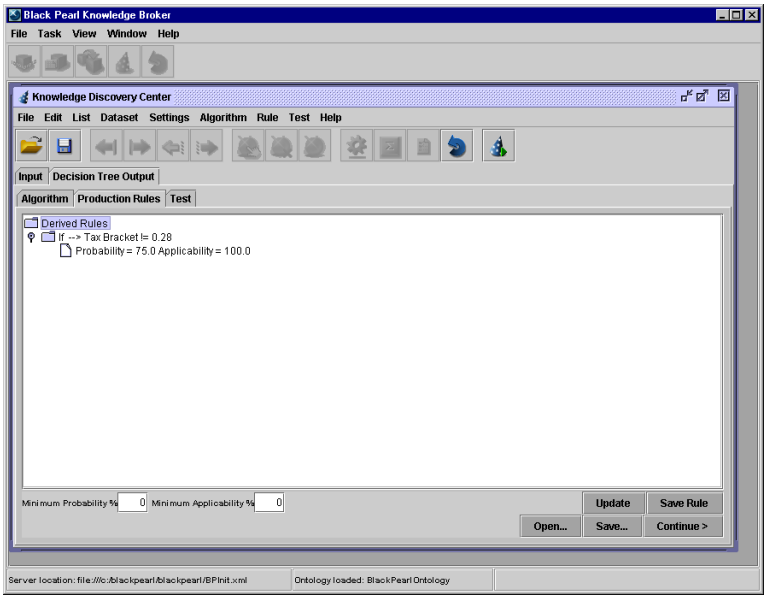


c Click Save to save the decision tree within an XML-format file. This decision tree can be embedded within the ontology as a Calculated Property (see *DecisionTreeCalculator on page 87*).

Updating Decision Tree Production Rules

1 Click Continue (or the Production Rules tab). The Decision Tree Production Rules Editor displays the rules that have been generated by the Decision Tree. These rules are now expressed as in an *if-then* format, suitable for incorporation within the knowledge base.

Figure 7-15. Decision Tree Production Rule Editor



2 Complete one or more of the following options:

a Type a number in the Minimum Probability % text field. Click Update to view the results. This will remove from display all rules that do not meet this minimum probability criteria. A probability of 1 means that a rule will always be true, while values nearer 0 indicate rules that are increasingly unlikely.

b Type a number in the Minimum Applicability % text field. Click Update to view the results. This will remove from display all rules that do not meet this minimum probability criteria. This measures to what extent of the data a rule applies. For example, if you select 80% here, then only rules that apply to more than 80% of the Dataset will be displayed.

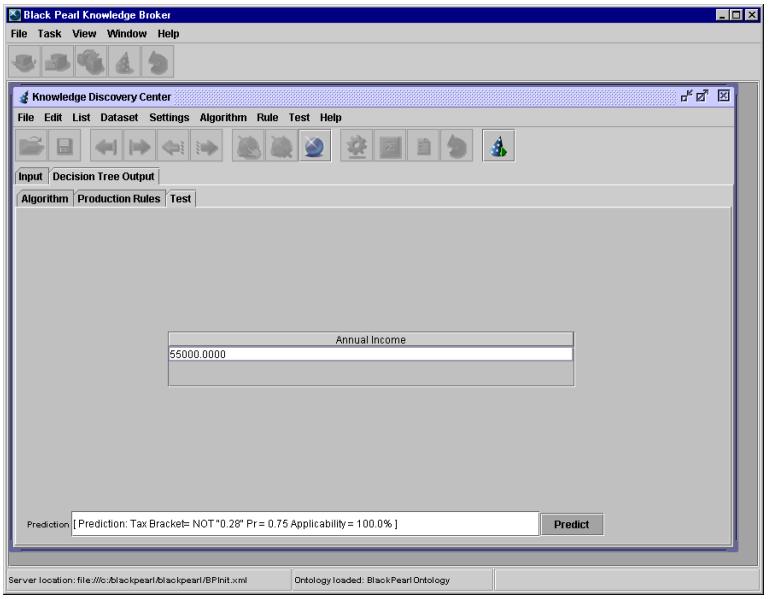
c Select an individual rule from the display list. Click Save Rule to save the item to the current knowledge base.

d Click Save to save the derived if-then rules within an XML-format file. This set of rules can be embedded within the ontology as a Calculated Property (see *RuleSetCalculator on page 87*).

Testing the Decision Tree Predictive Model

1 Click Continue (or click the Test tab). The Test Editor displays.

Figure 7-16. Decision Tree Test Editor



2 Click Predict. The results of the Decision Tree predictive model display.

3 Complete one or more of the following options:

   a Re-define your Decision Tree predictive model by repeating the steps in *Defining a Predictive Model Dataset on page 145* and *Generating and Updating a Decision Tree on page 161*.

   b Select File **>** Save in the Black Pearl Knowledge Broker Launcher to save your changes to the knowledge base.

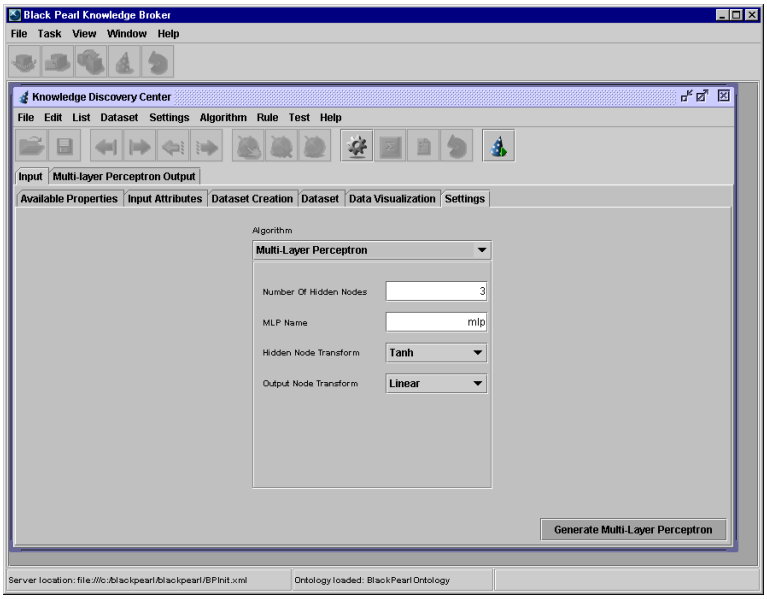
Creating and Training a Multi-Layer Perceptron

If you selected the MLP algorithm in the Settings Editor, you must complete the following:

- *Setting MLP Parameters*
- *Updating the MLP Algorithm*
- *Managing the MLP Test*
- *Managing MLP Output Variance*
- *Managing MLP Sensitivity*

Setting MLP Parameters

Figure 7-17. MLP Settings Editor



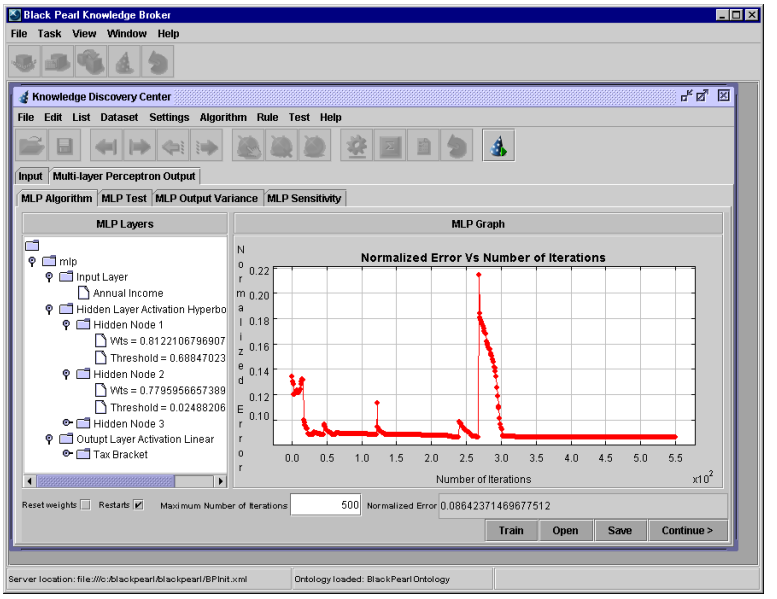
The MLP requires a number of parameters to be fine-tuned for maximum performance. This is mostly a process of trial-and-error, guided by your experiences of the MLPs performance in the past or with similar Datasets. The aim is to get the MLP to train successfully. That is, to reduce the Normalized Error between its forecasts for the Predicted Attribute values and actual values recorded in the Dataset to a reasonably low value. A good value to aim for is 0.08-0.1.

1. Open the Settings Editor, if it is not currently open.
2. Select Multi-layer Perceptron from the Algorithm drop-down list. Press Tab.
3. Type a value in the Number of Hidden Nodes field. Press Tab.

   Selecting a high value here may improve the accuracy of the MLP (because more nodes means more interconnections that lead to a more refined model of the relationship between the Input and Predicted Attributes). However, more nodes can also increase the time required for training, possibly making it impossible to train successfully or to reduce the error to a low-enough value. And with too many nodes, if the MLP does train successfully, the model it produces may be too "fragile" and work very well with the values in the Dataset but when you input additional real-world values the Predicted Attribute values will not be very accurate.
4. Type a name for the Multi-Layer Perceptron (MLP).
5. Click Generate Multi-Layer Perceptron.

Updating the MLP Algorithm

Figure 7-18. MLP Algorithm Editor



1 Open the MLP Algorithm Editor, if it is not currently open.

2 Complete one or more of the following options:

   a Click the Reset Weights checkbox to change the Hidden Node and Output Layer weights. You should do this is, by looking at the graph, you see that the error is not decreasing. One of the features of the MLP is that it can sometimes fail to train successfully and needs to be "rebooted" to try again.

   b Type a value in the Maximum Number of Iteration field to change the default value. You should avoid "overtraining" the MLP. This occurs when you reduce the error too much and the MLP learns the predictive pattern between the Input Attributes and the Predicted Attribute very well indeed. Sometimes too well, in fact, and in tests with additional real-world Input Attribute values that were not in your Dataset, the Predicted Attribute values may not be very accurate.

The value set in the Maximum Number of Iteration field does not indicate the maximum number of times the MLP will train, but the "do not exceed" value. Sometime the MLP will stop training—it won't be able to complete the number of requested iterations. This happens when further training will not reduce the error rate.

   c Click Train to enable the Black Pearl Knowledge Broker to "learn by example" and reduce the Normalized Error.

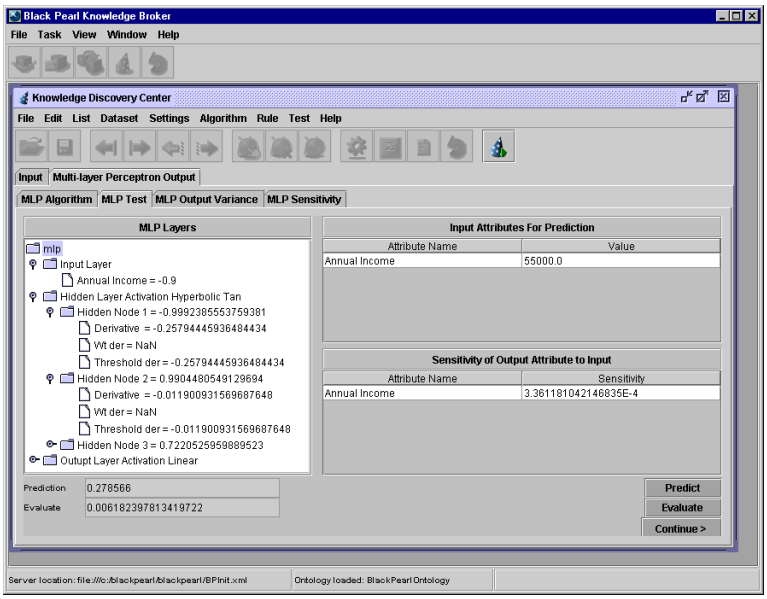
   d Click Save to save the MLP within an XML-format file. This pre-calculated MLP can be embedded within the ontology as a Calculated Property (see *MLPCalculator on page 87*).

   e Click Continue (or click the MLP Test tab).

Managing the MLP Test

The MLP Test Editor displays an exact number (as opposed to Normalized Error, which displays a number between 0 and 1). The number provides you with an average Root Mean Square between the Input and Predicted Attributes. You can use the MLP Test Editor to test the accuracy of the MLP predictive model.

Figure 7-19. MLP Test Editor



1 Open the MLP Test Editor, if it is not currently open.

2 Complete one or more of the following options:

   a Click Predict to change the Input Attributes and view the MLPs forecast for the value of the Predicted Attribute.

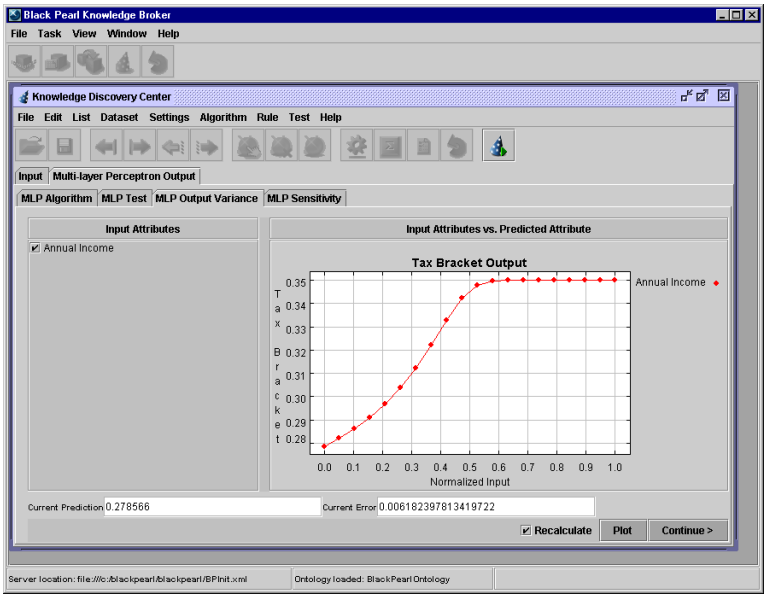
   b Click Evaluate to view the results of the MLP. This displays the Root Mean Square (RMS) error between the predicted value and the actual value for the sample dataset. To determine the efficacy of the MLP against a different dataset, click the Dataset Creation tab and create a new dataset. (See *Creating a Dataset on page 153* for detailed instructions.) Then return to the MLP Test Editor and click Evaluate to determine the RMS for the new dataset.

3 Click Continue (or the MLP Output Variance tab).

Managing MLP Output Variance

The MLP Output Variance Editor enables you to view the sensitivity of the Predicted Attribute to Input Attribute values by varying each Input Attribute around a point you specify in the MLP Test Editor. If the Predicted Attribute varies by a proportionately large amount, this indicates that the Input Attribute plays a significant role in the prediction and is a strong influence. On the other hand, if the Predicted Attribute does not vary significantly, this indicates that the Input Attribute plays an insignificant role in the prediction and potentially should not be used in the predictive model.

Figure 7-20. MLP Output Variance



1 Open the MLP Output Variance Editor, if it is not currently open.

2 Complete one or more of the following options:

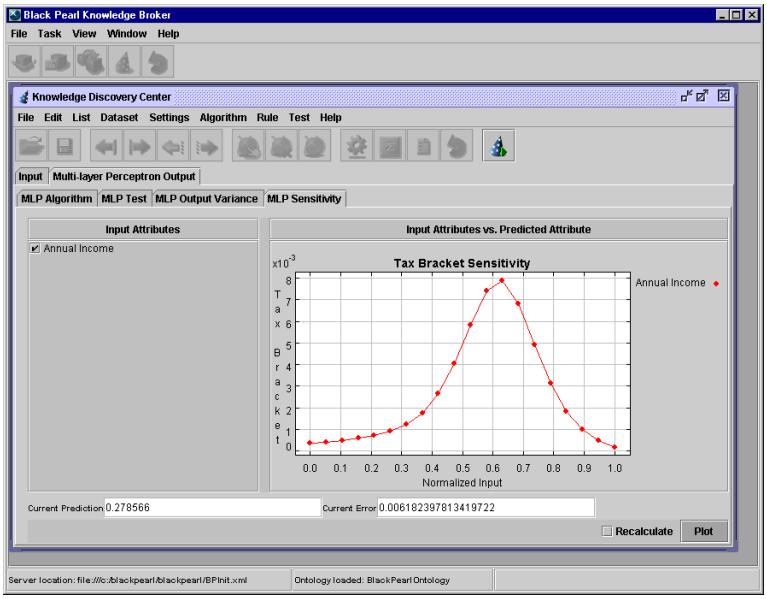
   a Click the Recalculate checkbox to recalculate the plot. This is useful if there is a change in the Input Attributes in the MLP Test tab.

   b Click Plot to refresh the plot.

3 Click Continue (or click the MLP Sensitivity tab).

Managing MLP Sensitivity

Figure 7-21. MLP Sensitivity



1 Open the MLP Sensitivity Editor, if it is not currently open.

2 Complete one or more of the following options:

   a Click the Recalculate checkbox to recalculate the plot. This is useful if there is a change in the Input Attributes in the MLP Test tab.

   b Click Plot to refresh the plot.

Creating a Knowledge Discovery Script

Creating and refining predictive models can be cumbersome and time-consuming, especially if you require more than a handful of predictive models. Scripting enables you automate the steps used in creating your predictive models and can deliver considerable time savings. You also gain the ability to customize the knowledge discovery process to your own preferences.

The process is similar to making a CD recording: Play, Stop, and Record, and Forward/Back buttons are used to control the process. You can record multiple tracks, and then run one or many tracks, as desired. Each track consists of a block of instructions that tells the Knowledge Discovery Center how to create or make changes to predictive models, or to make changes to the ontology.

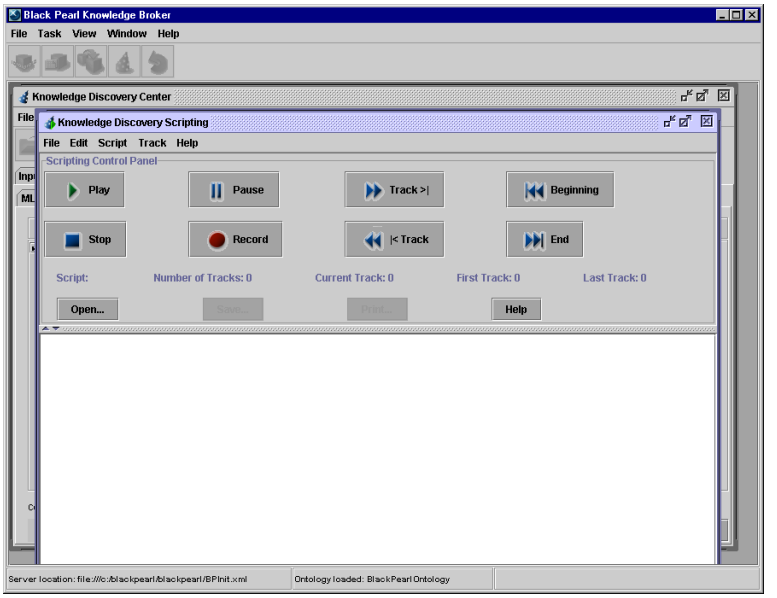
The tracks are stored in files in the `\blackpearl\data\kd` directory with filenames *`[Name]`*`_script.xml`. This directory contains a sample script with several tracks called `demo_script.xml`.

Starting Scripting

1 Start the Knowledge Discovery Center, if it is not currently open. This is a requirement for the Knowledge Discovery Scripting functions to operate.

2 Start the Knowledge Discovery Scripting by clicking on the appropriate icon in the Knowledge Discovery Center toolbar. You may also activate Scripting by selecting Test **>** Scripting from the Knowledge Discovery Center menu bar.The Knowledge Discovery Scripting window substantially replicates the appearance of a CD.

Figure 7-22. Knowledge Discovery Scripting Window



Loading a Pre-Recorded Script

The Black Pearl-supplied **demo_script.xml** contains five tracks. Advanced Knowledge Broker users can use an XML editor to alter these script files directly to match custom ontologies and produce desired results. The individual tracks are described below:

Table 7-1. Demo Script Tracks

Track	Description
CreateDatasetMLP	Creates a three-node, 1000-pass MLP with Annual Income as the Input Attribute and Tax Bracket as the Predicted Attribute.
CreateDatasetDecisionTree	Creates an unpruned tree, with Investment Goals as the Input Attribute and Risk Preference as the Predicted Attribute, with a test for Risk Preference as High.
RetrainMLP	Retrains the MLP created in Track 1 (CreateDatasetMLP), using 1000 passes.
CreateDataset	Creates a Dataset with Liquidity Preferences and Investments Goals as the Selected Properties.
AddNewConcept	Adds a new Business Concept called High Risk Customer to the Customer concept, selecting as a filter only those Customers whose Risk Preference is High.

These scripts are designed for a specific ontology and installation configuration and may not function on your installation without modification. KDC script users can download the free XML Notepad from this URL: http://msdn.microsoft.com/xml/notepad/ and modify these files as desired.

Part

IV

Tutorial

The Tutorial provides detailed lessons for using the Black Pearl Knowledge Broker to create an ontology, business rules, and predictive model suitable for an investment house.

Chapter

8

Example: Defining an Ontology

This chapter walks you through the basic steps you need to use the Black Pearl Knowledge Broker. It uses sample data for a fictitious company, ABC Financial, to demonstrate how to connect to a datasource and define an ontology.

The steps you will follow in this chapter are:

1. Connect to the sample datasource
2. Create Data
3. Organize your conceptual view of the data
4. Create Business Concepts and filters

Before You Begin

1. Install the Black Pearl Knowledge Broker.
2. Follow the steps in *Chapter 3, Configuring ODBC Data Source Drivers on page 52* to configure your JDBC/ODBC drivers.

Ontology Overview

The Black Pearl Knowledge Broker ontology consists of five components:

- *Datasource Connections,* which are URLs, strings, or pathnames to disparate data sources
- *Datasources,* which are the disparate Web, relational database, and flat files containing data pertinent to your business
- *Data Concepts*, which are the vocabulary terms that map directly to one or more of your datasources
- *Business Concepts*, which are the vocabulary terms defined by you that are relevant to your industry or company and map to other concepts in the ontology
- *Concept Relations*, which define the connections between concepts

Connecting to the Sample Datasource

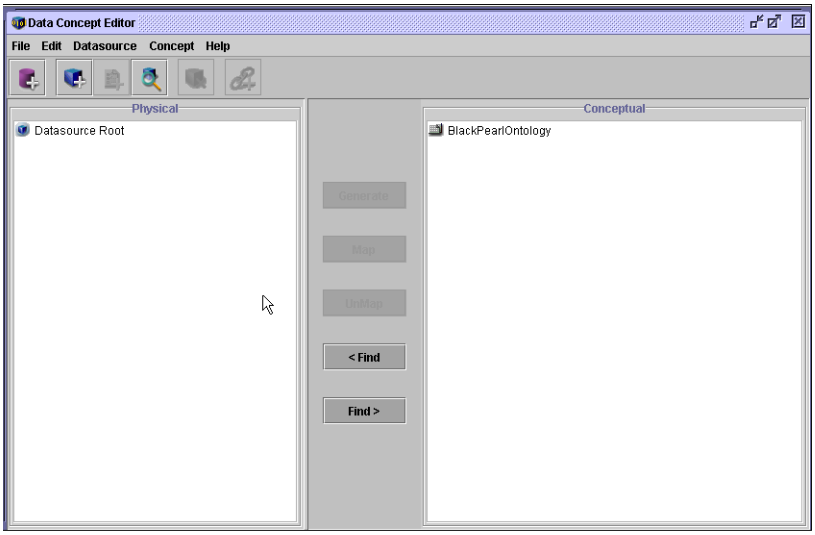
To perform its role, the Black Pearl Knowledge Broker must have access to one or more datasources. A *datasource* is a repository that contains your data. You will use this data to create *concepts*, the building blocks of your ontology. An *ontology* is the classification system that organizes concepts in a knowledge representation system.

In this tutorial you will connect to a sample datasource that contains customer, product, and account records for ABC Financial.

1 Select Start **>** Programs **>** BlackPearl **>** Knowledge Broker to launch the Black Pearl Knowledge Broker.

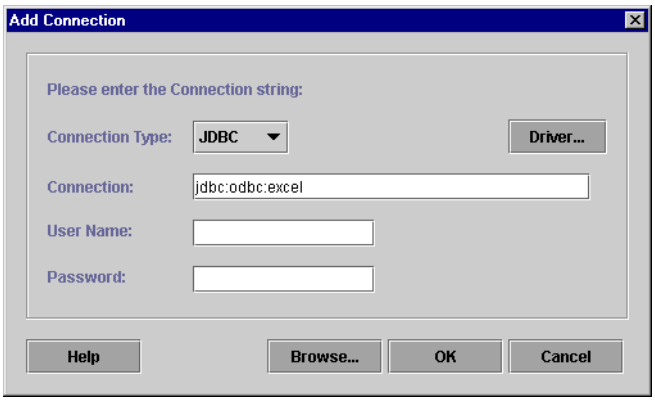
2 Click the Data Concept Editor button on the toolbar, or select Task **>** Data Concept Editor to open the Data Concept Editor.

Figure 8-1. Data Concept Editor



3 Click the Add Connection button on the toolbar, or select Datasource > Add Connection to open the Connection dialog.

Figure 8-2. Connection Dialog



The Connection dialog enables you to connect to datasources. Here, you will connect to the sample datasource. (For this tutorial, you will not need a User Name and Password.)

4 Make sure that JDBC is selected in the Connection Type drop-down list.

5 Enter jdbc:odbc:excel in the Connection field.

6 Click OK to connect to the datasource.

a If the datasource connection is successful, you will be notified with a success message. Click OK to exit the message.

Figure 8-3. Success Message

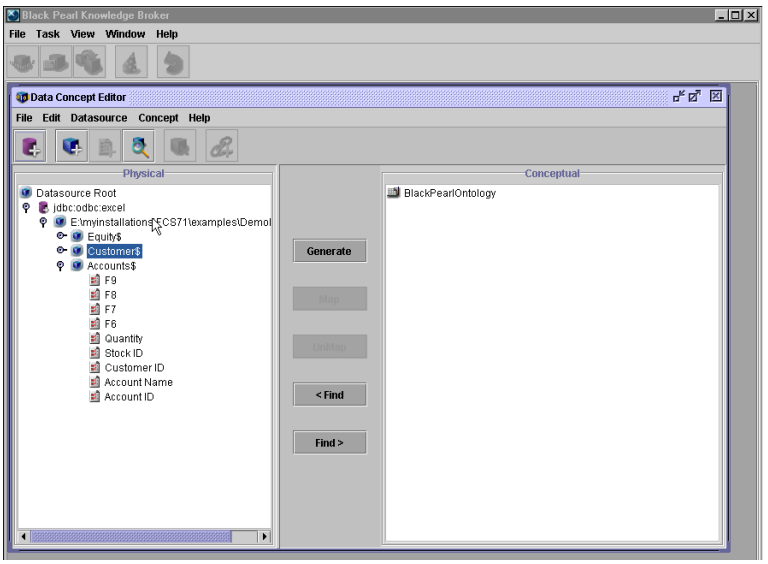
b If an error exists, or the datasource does not exist, you will be notified with an error message. If this occurs, click OK, then try re-entering the datasource, or contact your database administrator.

c If you have already gone through the tutorial, you will receive the following message: "Connection already added. Delete the connection before attempting to add the connection again." Click OK and Cancel to exit the dialogs, and delete the connection.

After you connect to a datasource, the left-hand Physical panel displays a Datasource List of all datasource connections.

7 Click the Expand icon in the Physical list to view the datasource components.

Figure 8-4. Expanded Datasource List



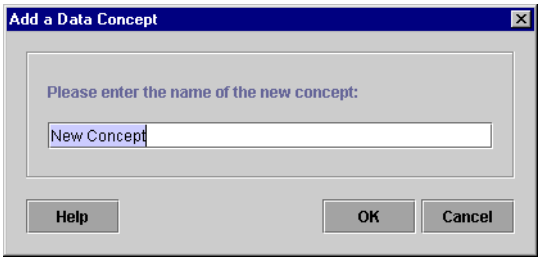
You are now ready to create the ontology. As mentioned earlier, it is the classification system that organizes your concepts. It is a repository of the data and business concepts which you create, and the relationships between them.

Creating Basic Data Representations

After you connect to a datasource, you must give it meaning by creating data concepts and business concepts. You accomplish this in the Data Concept Editor.

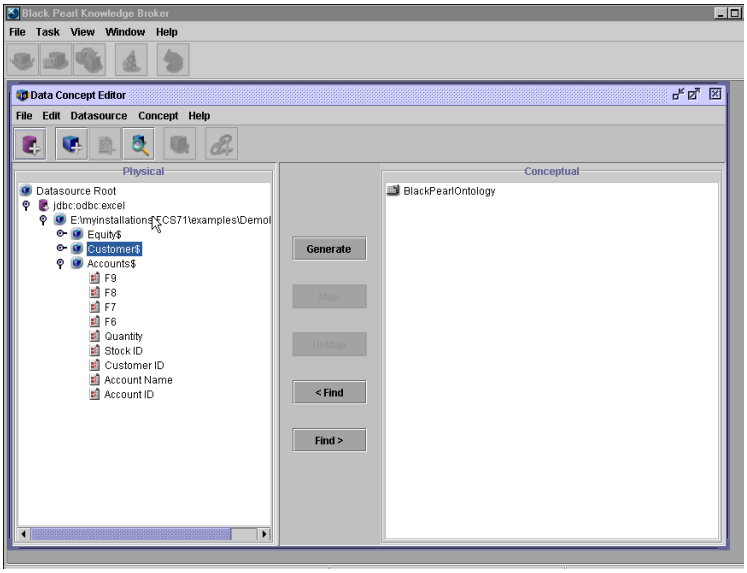
1. In the Data Concept Editor, click the Add Concept button, or choose Concept **>** Add Concept to display the Add a Data Concept dialog.
2. Enter Customer, the name of your new concept.

Figure 8-5. Add a Data Concept Dialog



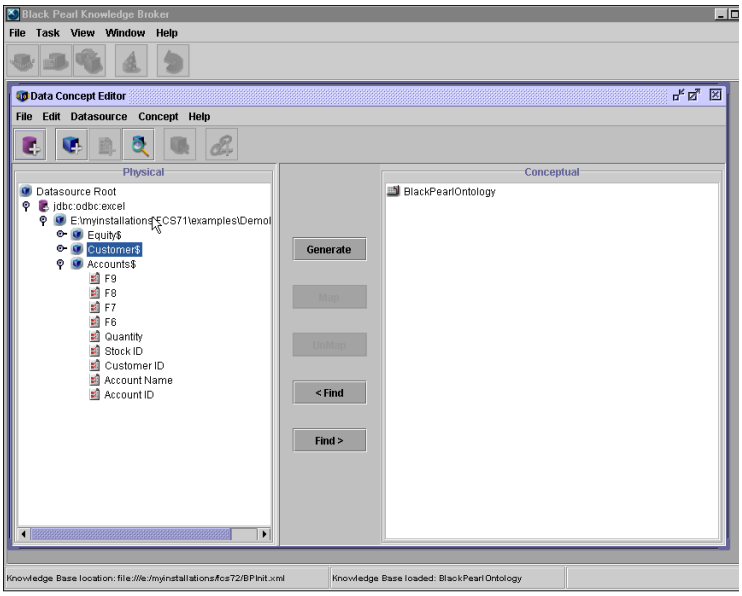
3. Click OK to exit the Add a Data Concept dialog. The Customer concept appears in the Data Concepts list under the root ontology, *Black Pearl Ontology*.

Figure 8-6. Data Concepts List



Next, you will populate the Customer concept with data from your datasource. You will do this by linking the data from the Customer spreadsheet to the Customer concept using the Map button.

Figure 8-7. Mapping a Datasource to a Data Concept



4 Select the Customer$ spreadsheet from the Physical list, and the Customer concept from the Conceptual list in the right-hand window. Click the Map button to link them together. A success message appears.

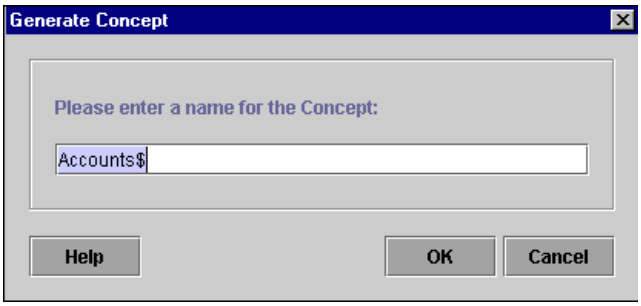
Figure 8-8. Success Message

5 Click OK to close the success message.

Now you will use the Generate button to add two more concepts to your ontology, and populate them with data in just one step.

6 Select the Accounts$ spreadsheet from the Datasource list, and click the Generate button. The Generate Concept dialog appears, displaying the name of the selected datasource the Name field.

Figure 8-9. Generate Concept

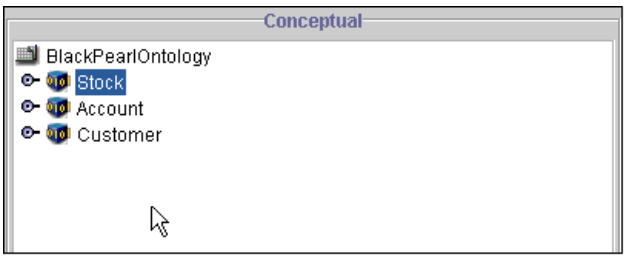


7 Change the Name to "Account," and click OK. This automatically links the "Account" concept to the data in the Accounts$ spreadsheet, and displays it in the list of Data Concepts.

8 Repeat the above steps to generate a new concept called "Stock," and link it to the Equities$ spreadsheet.

You have now created three new data concepts, and populated them with data.

Figure 8-10. Data Concepts List



9 Select File > Close in the Data Concept Editor to close the Editor.

10 Select File > Save Knowledge Base from the Launcher menu to save the concepts and mappings to the ontology.

11 Select File > Close in the Data Concept Editor to close the Editor

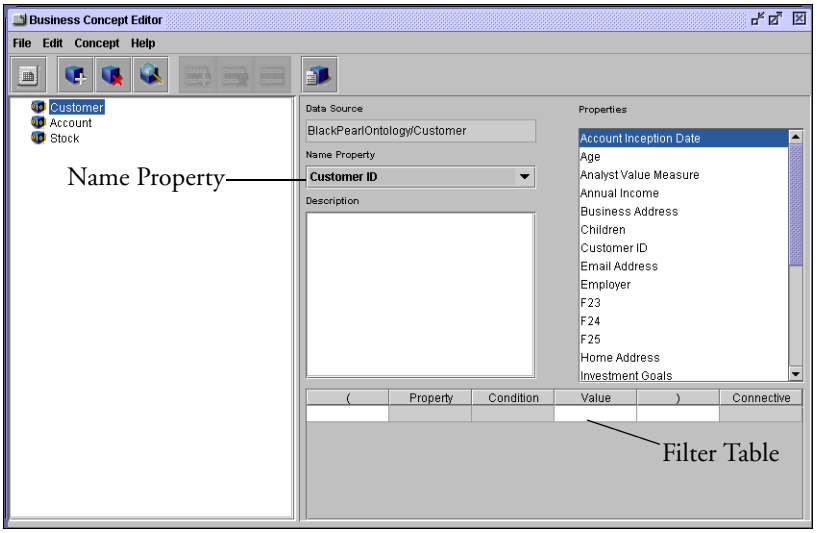
So far, you have given meaning to your data, enabling the Black Pearl Knowledge Broker to conceptualize the sample datasource as a list of customer, account, and stock concepts. Next, you will learn how to organize your view of data.

Organizing Your View of the Data

You can use the Business Concept Editor to define your view of properties of concepts (for example, according to Product ID, Name, or Brand).

1 Click the Business Concept Editor button on the toolbar, or select Task **>** Business Concept Editor to open the Business Concept Editor. The concepts you created earlier (Customer, Account, and Stock) appear on the left-hand side of the dialog.

Figure 8-11. Business Concept Editor



2 Select the Customer concept to display its properties on the right side of the dialog, including its datasource, optional description, and properties.

The *Name Property* is the 'key' you will use to order the list of properties in a view. For example, you can list properties of the Customer concept by Age, Name, or Customer ID. It is recommended that you choose a distinct attribute that is specific for each customer (such as Customer ID).

3 Choose Customer ID from the Name Property drop-down list, and click the Show Instances button, or select Concept **>** Show Instances. This displays all of the properties of the Customer concept, organized by Customer ID. Select another instance from the list of instances to display its information on the right-hand side of the dialog. Repeat the process and select different properties in the Name Property box to familiarize yourself with this feature. When you are done, click OK to exit the Concept Instances dialog.

4 Repeat Step 3 to display instances of the Account and Stock concepts. Select Account ID as the Name Property for the Account concept, and Symbol as the Name Property for the Stock concept.

Figure 8-12. Account Concept Instances

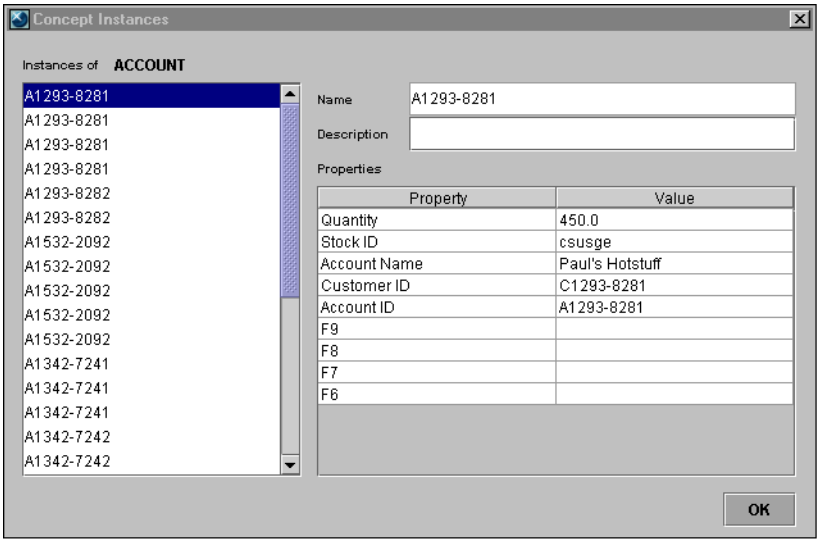
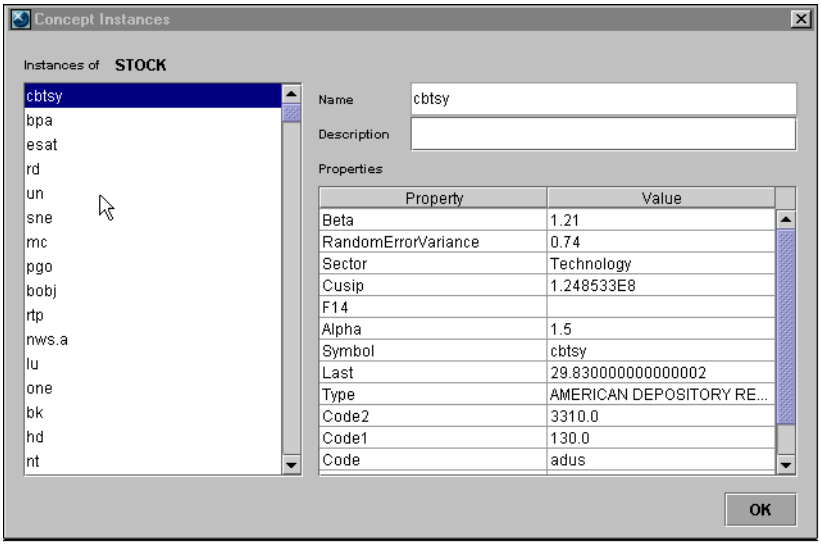


Figure 8-13. Stock Concept Instances



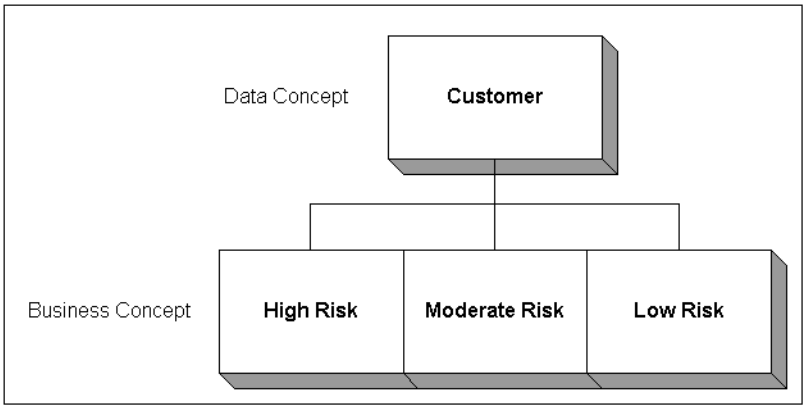
5 Click OK to exit the Concept Instances dialog.

So far, you have ordered your view of data by selecting data concepts, and viewing their properties and instances. Now that you understand how to order your view of data, you will organize data concepts into business concepts. Later in the tutorial, you will learn how this helps you, especially when creating marketing objectives.

Creating Business Concepts and Filters

You can add business concepts to organize the concepts in your ontology. For example, a Customer concept may represent several kinds of customers, such as Low, Moderate, and High Risk customers, for which you will need to create business concepts.

Figure 8-14. Customer Concept

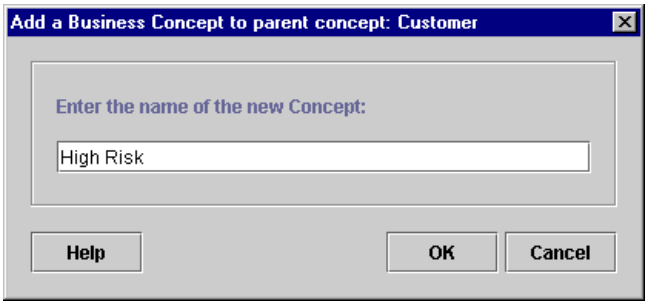


Business concepts can be derived from both data concepts, and other business concepts.

You can also create *filters* for concepts that qualify concepts when they are used in a rule. For example, in the rule "If customer such that age is less than 35, the recommend buy Internet stocks," the condition, *age is less than 35,* is the filter. The Black Pearl Knowledge Broker lets you create a constraint filter, described in *Chapter 6, Defining Rules on page 113*.

1 In the Business Concept Editor, select the Customer concept from the left-hand side of the dialog. Click the Add Concept button, or select Concept **>** Add Concept to add a Business Concept to the Customer concept. The Add Concept dialog appears, where you will enter a name for the new concept.

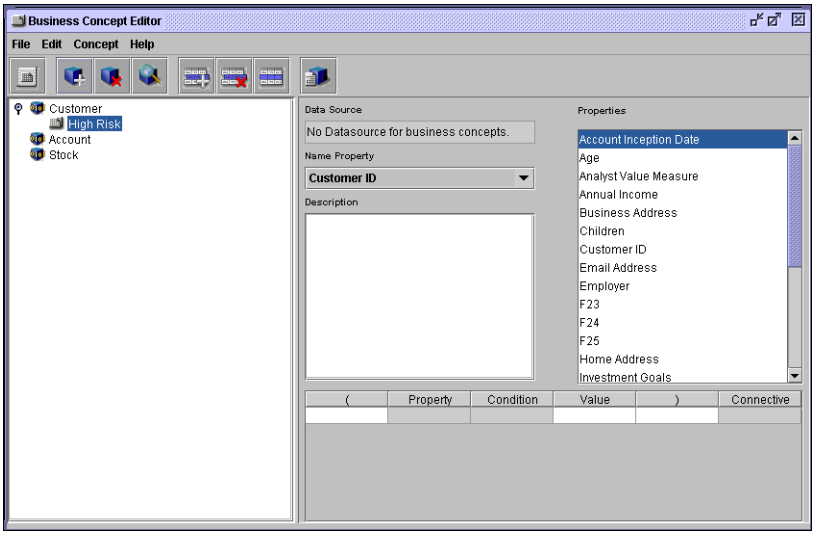
Figure 8-15. Add Business Concept



2 Enter High Risk into the Concept Name field. (You will notice that "Customer" appears in the title bar as the parent concept of the new High Risk concept.)

3 Click OK to exit the Add Concept dialog.

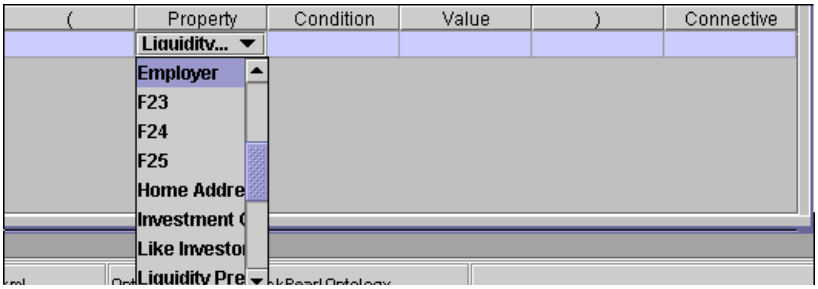
Figure 8-16. Business Concept Added



The new concept, "High Risk," appears in the list under its parent concept, Customer. Now you will add a filter that selects only those customers whose liquidity preference is high.

4 In the filter table (located on the lower right portion of the dialog), click on Property and select Liquidity Preferences from the drop-down list that appears.

Figure 8-17. Business Concept Properties List



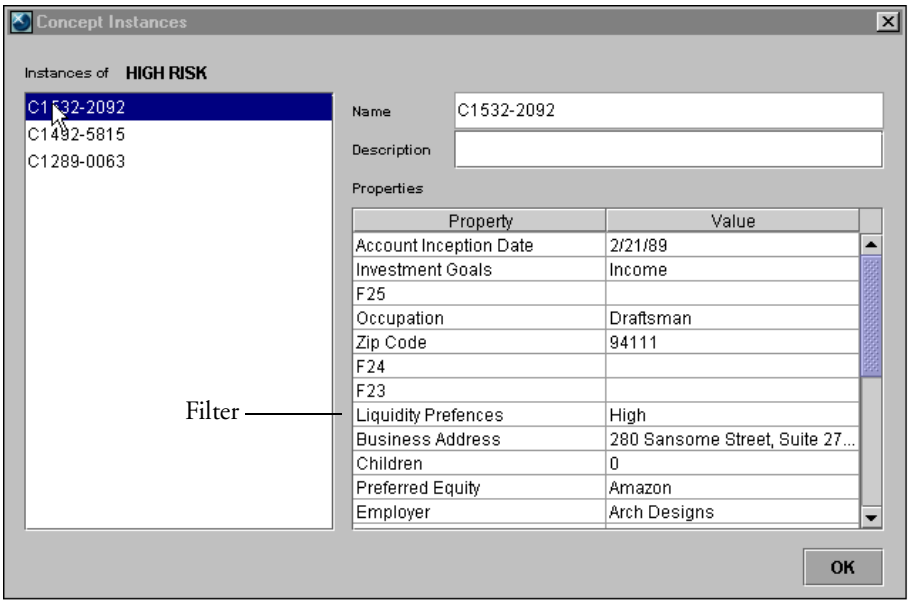
5 Select "=" from the Condition drop-down list, and enter "High" in the Value field. Click outside the cell to make sure that each entry is highlighted in blue, assuring that your input is read. The final filter should look like the one in Figure 8-18.

Figure 8-18. Filter Added

(	Property	Condition	Value	)	Connective
	Liquidity Pref...	=	High		

6 To view the new set of instances that the filtered High Risk concept contains, click the Show Instances button, or select Concept **>** Show Instances. This displays the list of all Customer concepts that match the filter "*Liquidity Preferences* = *High*." Click on each instance and view the value of the Liquidity Preferences field in the Properties table to ensure that it matches your filter and equals "High".

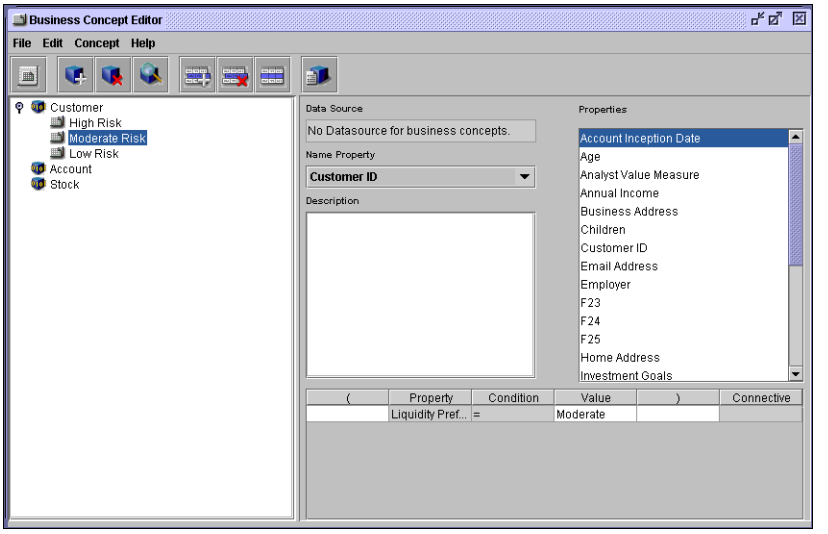
Figure 8-19. Customer Concept Instances



7 Click OK to exit the Concept Instances dialog.

8 Repeat the steps you learned above to create two more Customer concepts: "Moderate Risk," and "Low Risk," with Liquidity Preferences equal to Moderate and Low respectively.

Figure 8-20. Multiple Business Concepts Added

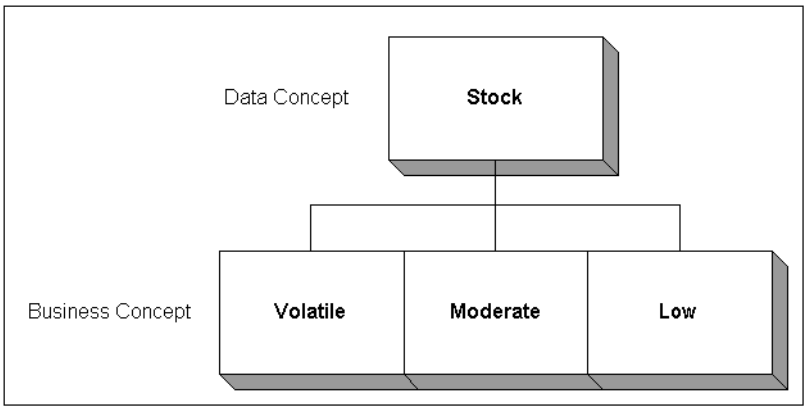


Make sure that Customer is always the parent concept.

Next, you will create three new business concepts for the Stock concept: Volatile, Moderate, and Low. You will filter them according to their Beta values.

Beta is a measure of the volatility of a stock. The higher the Beta value, the more volatile the stock.

Figure 8-21. Stock Concept



9 Select the Stock concept from the left side of the Business Concept Editor. Choose Concept **>** Add Concept, or click the Add Concept button on the toolbar. The Add Concept dialog displays.

10 Enter "Volatile" in the Concept Name field, and click OK.

11 In the filter table, select Beta from the Property drop-down list, select ">" from the Condition drop-down list, and enter "5" for the Value.

12 In the Connective drop-down list, select "and" to add a second row to the filter table.

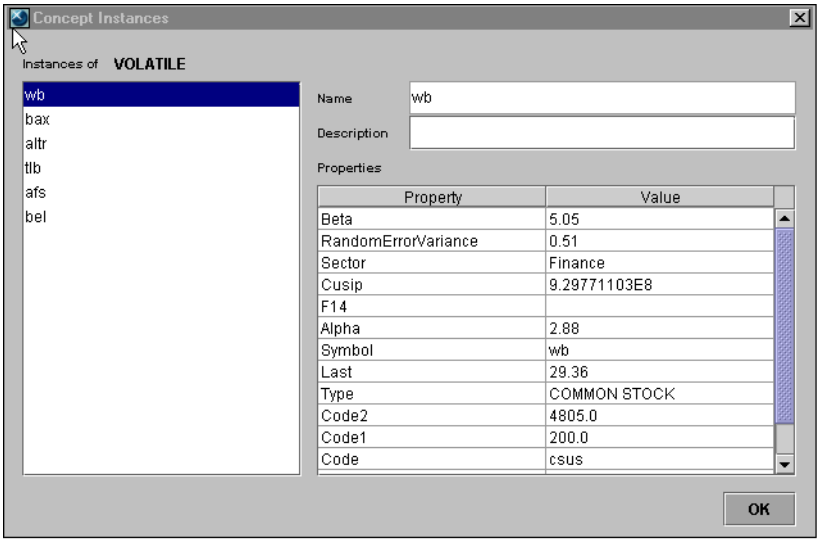
13 Click in the second row, and select the Beta property from the Property drop-down list, select "<" from the Condition drop-down list, and enter "5.1" for the Value, as shown in Figure 8-22.

Figure 8-22. Multiple Filter Rows Added

(	Property	Condition	Value	)	Connective
	BETA	>	5.0		and
	BETA	<	5.1		

You have created a business concept, "Volatile," and filtered it to include only those instances of the Stock concept that contain a Beta value greater than 5, and less then 5.1. To see the fruits of your hard work, with "Volatile" highlighted in the concept list, click the Show Instances button to display a list of stocks in the Volatile concept. View the Beta values to check them against your filter. They should be between "5" and "5.1".

Figure 8-23. Volatile Concept Instances



14 Click OK to exit the Concept Instances dialog.

15 Now use the above steps to create the following business concepts under the Stock concept, and their corresponding filters:

- **a** Moderate, Beta Value > 2 and < 2.1.
- **b** Low, Beta Value > 0 and < 1.1

16 Close the Business Concept Editor.

17 Select File **>** Save from the Launcher menu to save new concepts and filters to the ontology.

At this point, you have built the foundation for creating rules, giving the Black Pearl Knowledge Broker its knowledge of the world by connecting to a datasource, and creating data concepts. You have derived business concepts, and applied filters that reduce the set of data before rules processing occurs. The ontology holds this critical information and will establish context during rules evaluation.

You are now ready to create rules.

Chapter

9

Example: Defining Rules

This part of the tutorial shows you how to define and manage rules. It should be used after completing *Chapter 8, Example: Defining an Ontology on page 179*.

Creating Rules to Establish Marketing Objectives

To make relevant recommendations, the Black Pearl Knowledge Broker must know what rules to follow. *Rules* allow the Black Pearl Knowledge Broker to make recommendations that adhere to your marketing objectives. Although the Black Pearl Knowledge Broker supports the creation of extremely complex rules, to get you up to speed quickly the rules you will create in this tutorial are simple "if then else" rules.

Each rule is of the form:

```
if <SUBJECT> then <VERB > to <DIRECT OBJECT>
```

or, simply, "if X is true, then perform action Y on Z".

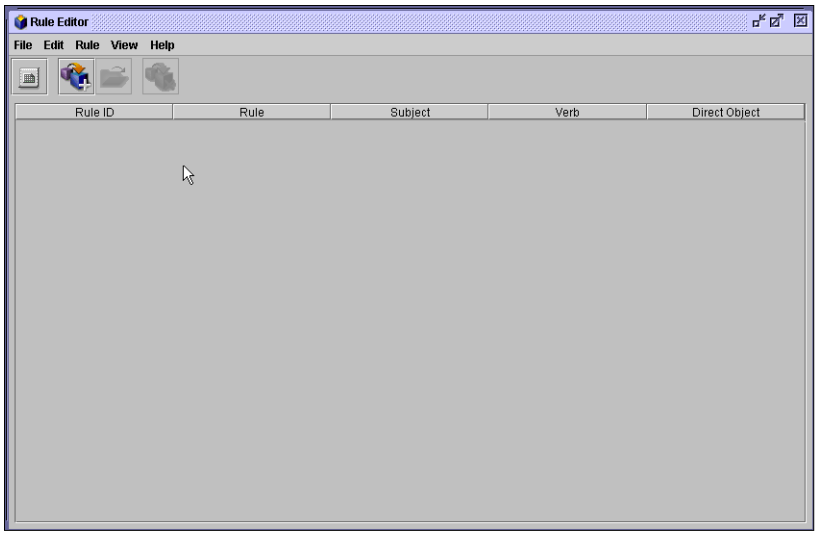
This part of the tutorial will show you how to create rules to meet the marketing objectives of the ABC Financial company, listed in Table 9-1.

Table 9-1. Marketing Objectives

Customer	Marketing Objective
High Risk	Recommend Volatile risk stocks
Moderate Risk	Recommend Moderate risk stocks
Low Risk	Recommend Low risk stocks

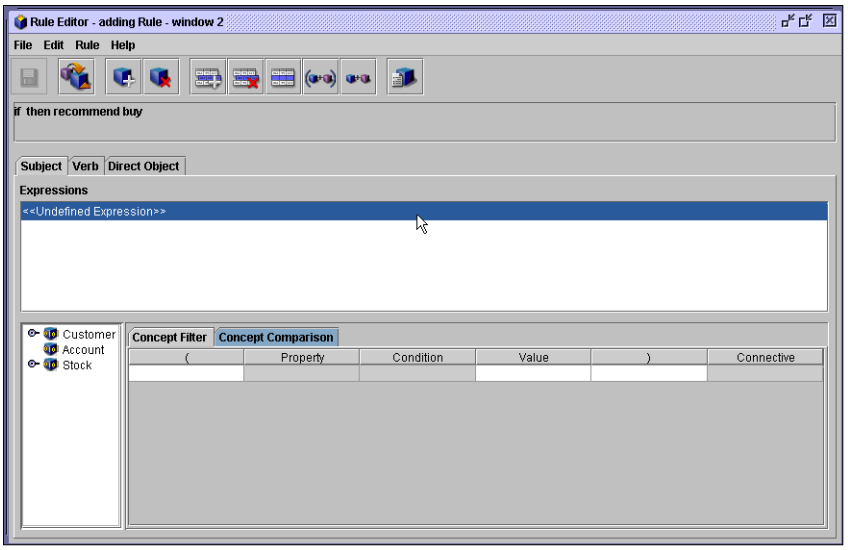
1 Click the Rule Editor button on the toolbar, or select Task **>** Rule Editor to open the Rule Editor. If this is the first time you have used the Black Pearl Knowledge Broker to create rules, you will see an empty window, indicating that no rules have been created.

Figure 9-1. Rule Editor



2 Click the Add Rule button, or select Rule > Add Rule to open the secondary Rule Editor, where you create and manage rules.

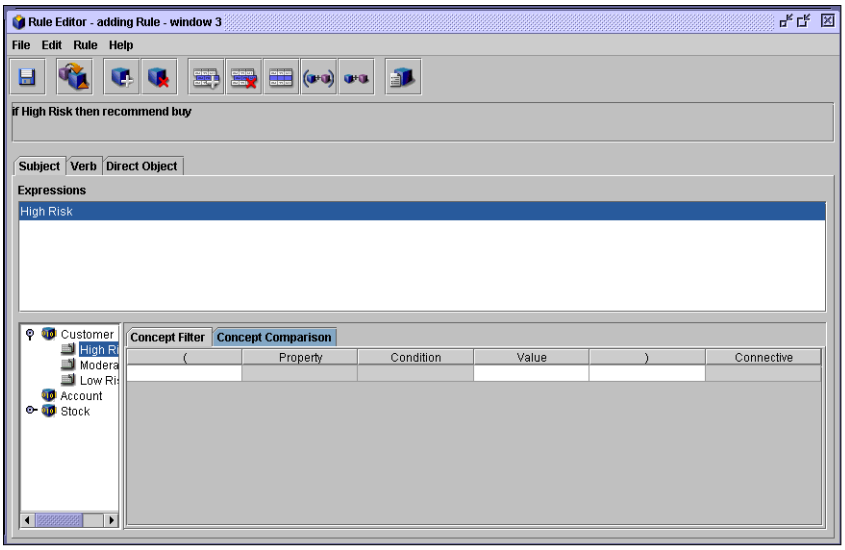
Figure 9-2. Secondary Rule Editor



The building blocks for the rule you will create appear in the top blue portion of the dialog. You can use the tabs to insert each part of the rule (Subject, Verb, Direct Object), to create filters for rules, and to do concept comparisons.

3 First, you will insert the subject into the rule. Click the Subject tab and expand the list of customers in the lower left portion of the dialog, and select High Risk to place it in the rule at the top of the dialog.

Figure 9-3. Expanded Subject Tab



The *Subject* is the qualifying condition that must be met before the rule can be run.

4 Click the Verb tab, and select recommend buy to add it to the rule.

Figure 9-4. Subject and Verb Added to Rule

The *Verb* is the action portion of a rule, and indicates what will occur when the rule is processed.

5 Click the Direct Object tab, expand the Customer concept, and select the Volatile concept to add it to the rule.

The *Direct Object* is the recipient of the action.

Congratulations! You have created your first rule. It says:

```
If a customer is high risk, then recommend volatile stock.
```

6 Select File **>** Save Rule, or click the Save button in the Rule Editor to save the rule.

7 Close the secondary Rule Editor. The new rule appears in the list of rules displayed in the main Rule Editor. It has been assigned a Rule ID.

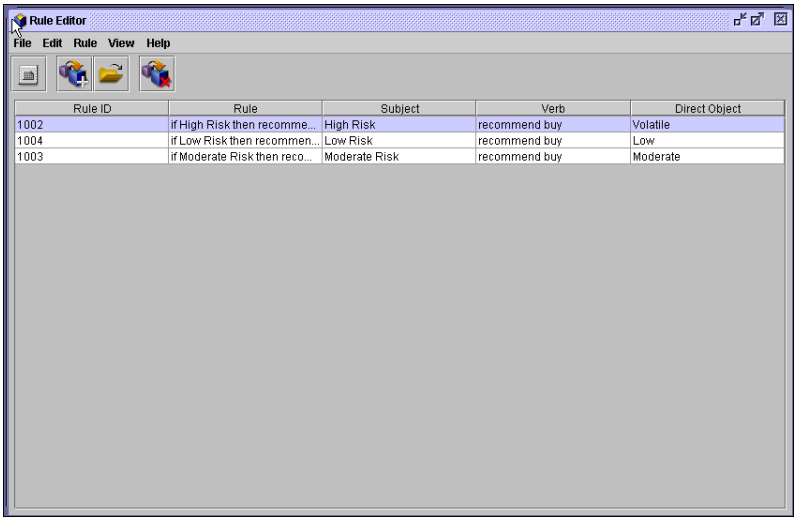
8 Repeat steps 1 through 7 to create the following two rules:

   a if Moderate Risk then recommend buy Moderate

   b if Low Risk then recommend buy Low

9 Close the secondary Rule Editor to view the list of rules that you have created. Each rule in the list has an associated ID number.

Figure 9-5. Multiple Rules Listed



10 Close the Rule Editor.

11 From the main Launcher window, select File **>** Save Knowledge Base.

At this point, you have given the Black Pearl Knowledge Broker the knowledge to recommend the purchase of volatile stocks to high-risk customers, moderate stocks to moderate-risk customers, and low stocks to low-risk customers.

"OK, great, but how does this help me?" you might ask. Applications that access the Black Pearl Knowledge Broker receive relevant recommendations and suggestions for their queries. For example, suppose that you are a stock broker with the ABC Financial company, responsible for managing customer financial accounts. A customer calls and asks, "What should I buy today?" Unless you are very experienced in the world of finance, and have a strong relationship with the customer, you may not have an easy time answering their question. However, via your broker application, you can query the Black Pearl Knowledge Broker for suggestions on what to do for a particular customer. If the customer profile matches your Moderate Risk concept, the purchase of Moderate stocks may be recommended. If the customer profile matches the Low Risk concept, the purchase of Low Risk concept stocks may be recommended, and so on.

Although this tutorial provides a very simple example of how to create concepts and rules, imagine the power the Black Pearl Knowledge Broker can provide after you create more complex concepts and rules for your specific business situation.

The next part of the tutorial shows you how to define a predictive model.

Chapter

10

Example: Defining a Predictive Model

This part of the tutorial shows you how to define a predictive model.

Analyzing Your Data

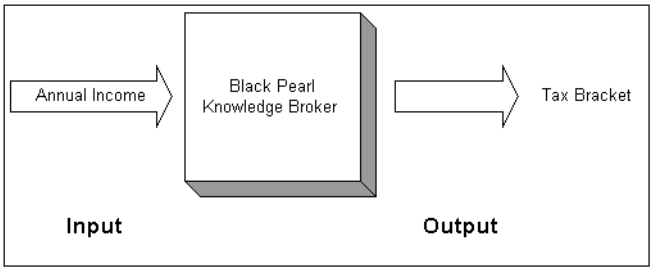
A *predictive model* is a model that you create and use to analyze your data and predict an outcome. It uses data with known results to develop a model that can be used to explicitly predict values for different data. You can use a predictive model to determine a customer's reaction to offers and recommendations, or particular characteristics such as age and risk.

For example, you can create a model using the payment history of customers who apply for loans to help identify those customers who are likely to default on loans, and those who are likely to repay them. For this tutorial, you are going to predict a customer's tax bracket based on their annual income.

To accomplish this, you must provide the following information in the Knowledge Discovery Center:

- One or more attributes to look at (*input)*, such as annual income.
- The attribute you want to predict (*output)*, such as the tax bracket.

Figure 10-1. Predictive Model



Before you define your predictive model, ask yourself the following questions:

1. What do you want to predict? (For example, a customer's tax bracket? their risk level?)
2. What attributes influence what you are predicting? (For example, does annual income influence tax bracket? will marital status affect risk level?)

Overview of Steps

The steps you will follow to define the sample predictive model are:

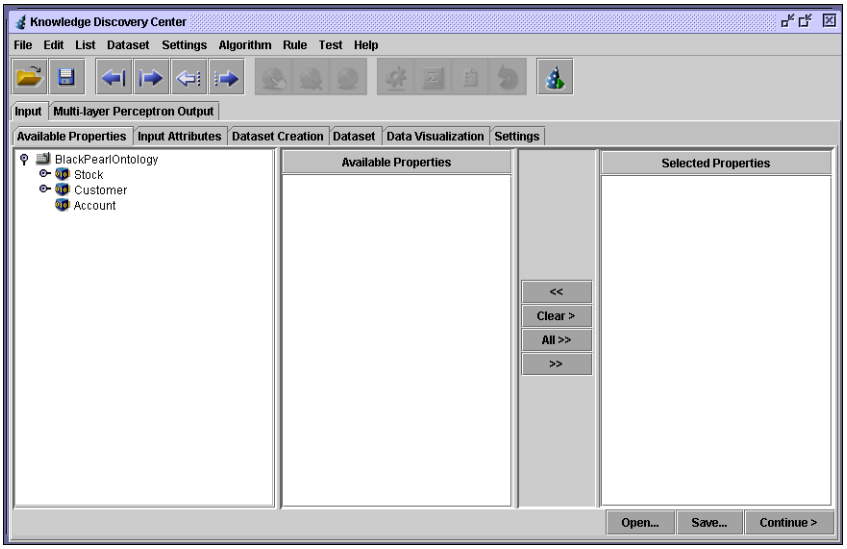
1. Choose what you want to predict (the *output attribute*), and what data elements will influence that prediction (the *input attributes*).
2. Define the dataset by assigning a name to the model, and choosing the number and type of sample data elements to analyze.
3. Visualize the data in a graphical chart.
4. Choose a type of algorithm you want use (Decision Tree, or Multi-Layer Perceptron), and adjust its settings.
5. Train the predictive model.
6. Test the accuracy of the predictive model using the MLP Test function.

Choosing Input and Output Attributes

This part of the tutorial shows you how to choose the item you want to predict and select the items that will influence it.

1 Click the Knowledge Discovery Center button on the Toolbar, or select Task > Knowledge Discovery Center to open the Knowledge Discovery Center.

Figure 10-2. Knowledge Discovery Center



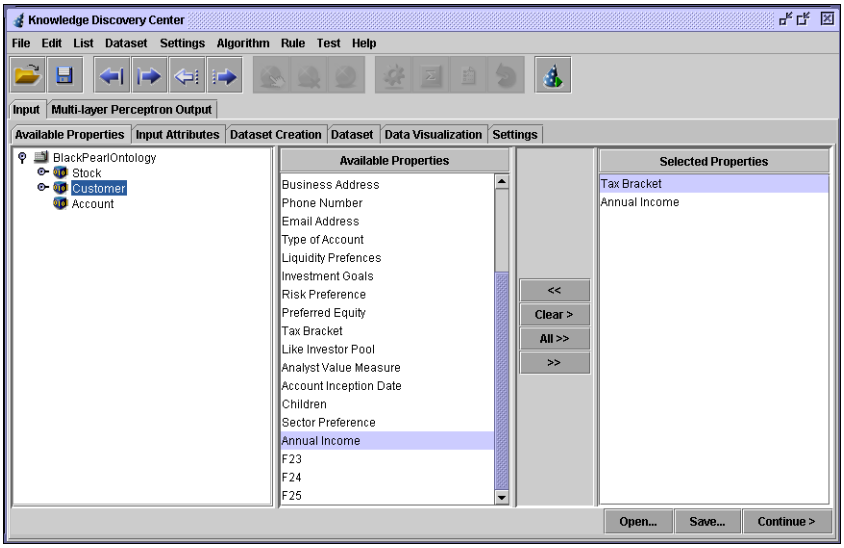
The Business Concepts that you created earlier appear in the Available Properties panel. These will provide the input for your predictive model.

2 Choose the Customer concept from the Available Properties tab. This populates the Available Properties list with the available properties for that concept, and allows you to define only the properties of interest for this prediction.

3 Select the following attributes from the Available Properties list, and use the arrow button to place them into the Selected Properties list:

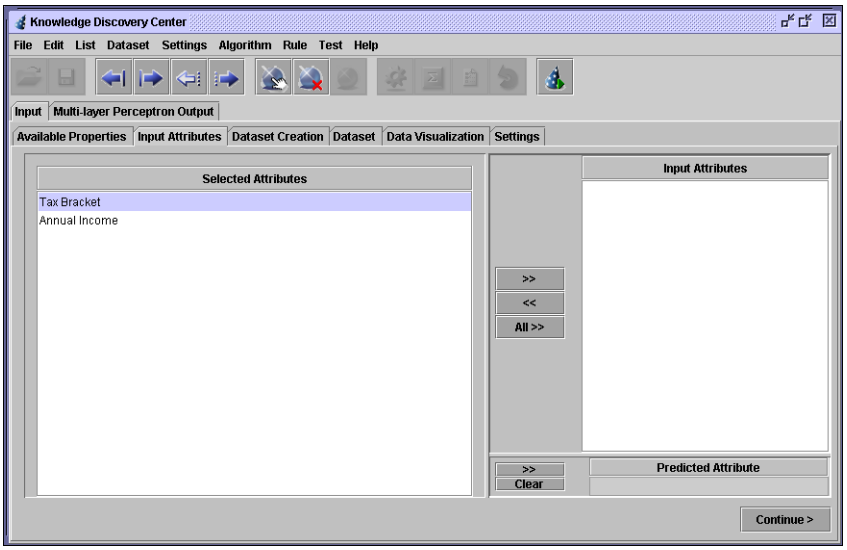
- Tax Bracket
- Annual Income

Figure 10-3. Selected Properties



4 Click Continue to open the Input Attributes tab. This tab enables you to define which attribute you want to predict, and which attributes will influence that prediction.

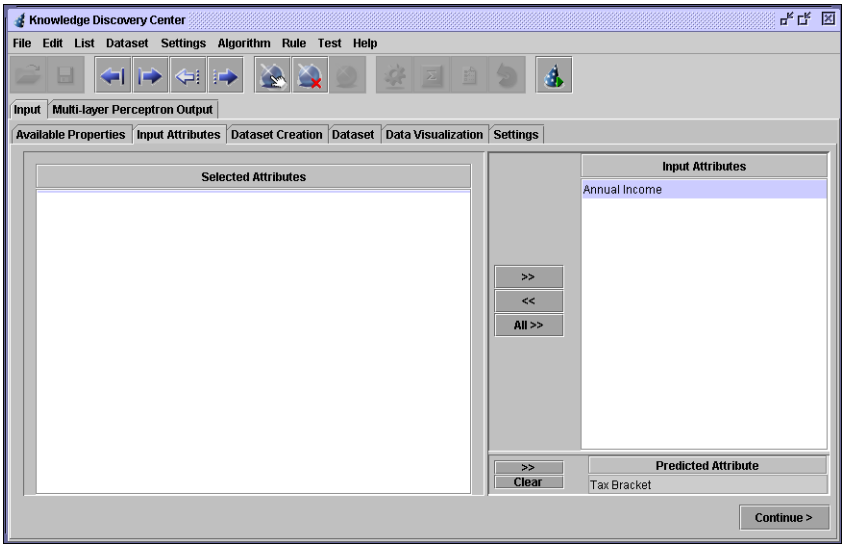
Figure 10-4. Selected Attributes



5 Select Annual Income from the Selected Attributes panel, and use the top arrow button to place it into the Input Attributes panel.

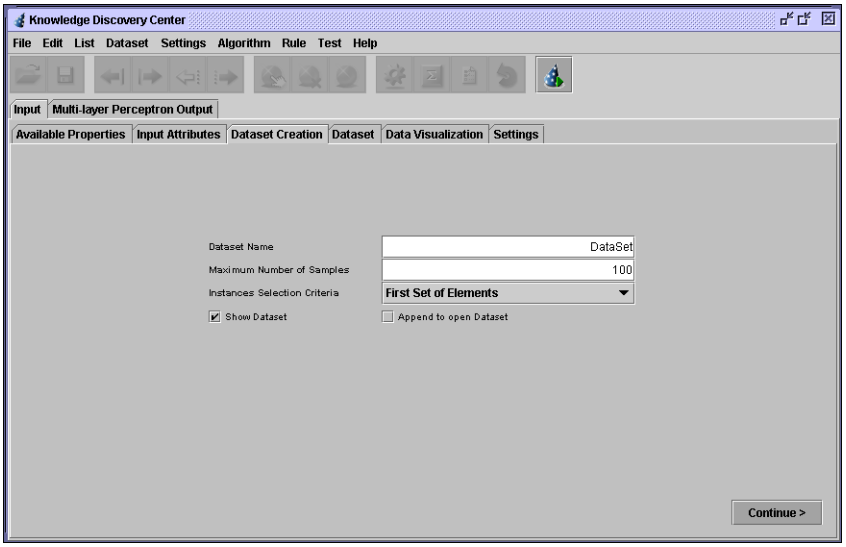
6 Select Tax Bracket from the Selected Attributes list, and use the lower arrow button to place it into the Predicted Attribute panel.

Figure 10-5. Input Attributes



7 Click Continue to open the Dataset Creation tab.

Figure 10-6. Dataset Creation Tab



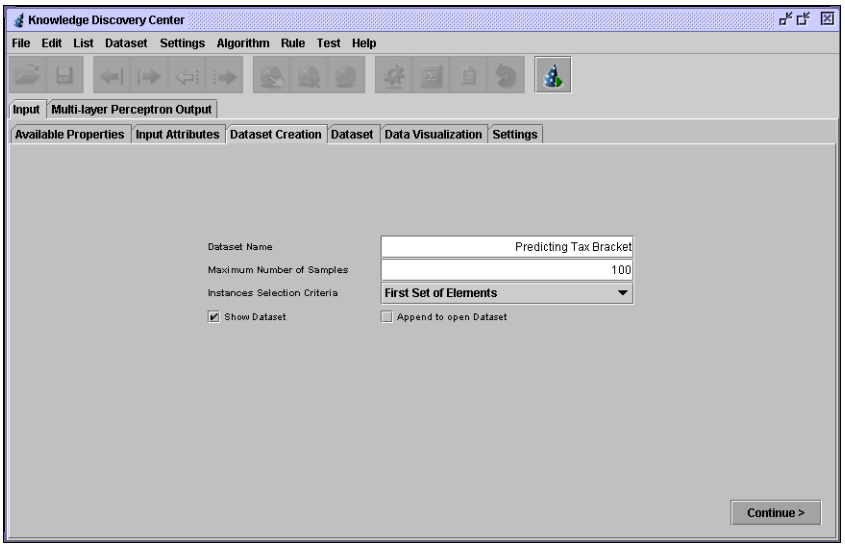
Defining the Dataset

The Dataset Creation tab enables you to name the predictive model and determine the number of sample data elements you want to analyze.

Selecting many samples allows for better predictive modeling, but can be time-consuming. You may wait up to a couple of seconds for a predictive model to be generated if the sample is large enough. However, choosing too few samples is not recommended.

1. In the Dataset Creation tab, type Predicting Tax Bracket in the Dataset Name field. This is the name of your dataset.
2. In the Maximum Number of Samples field, type 100. This tells the predictive model to use 100 data elements from the total number of elements in the datasource as the sample for the predictive model.
3. Select First Set of Elements from the Instances Selection Criteria drop-down list. This tells the predictive model to use the *first* set of elements in the dataset, in this case, elements 1 through 100.
4. Select the Show Dataset option so you can view the dataset you are creating.
5. Make sure that the Append to Open Dataset option is not selected.

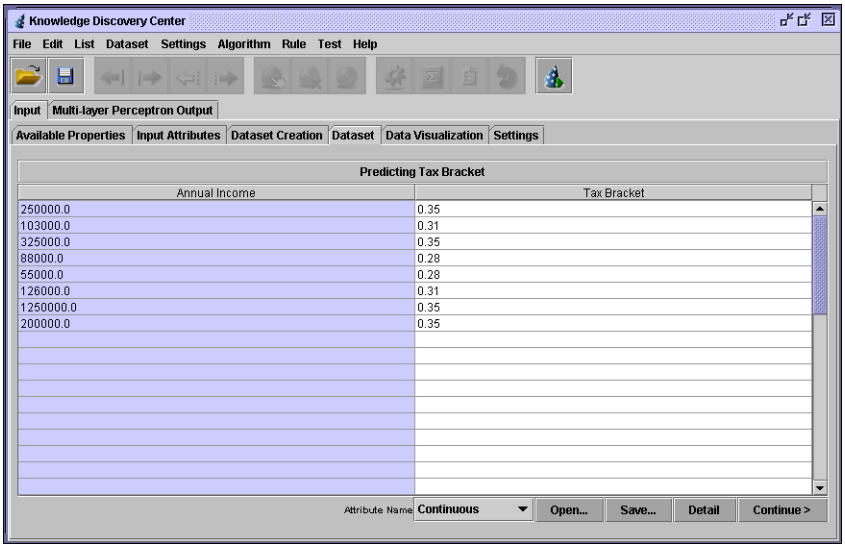
Figure 10-7. Dataset Creation Tab



6. Click Continue to open the Dataset tab.

The Dataset tab displays a view of instances that reflects the selections you made in the previous dialogs. At the top of the dialog, you will see the name you assigned to your predictive model. Each column heading displays one of the attributes you specified.

Figure 10-8. Dataset Tab

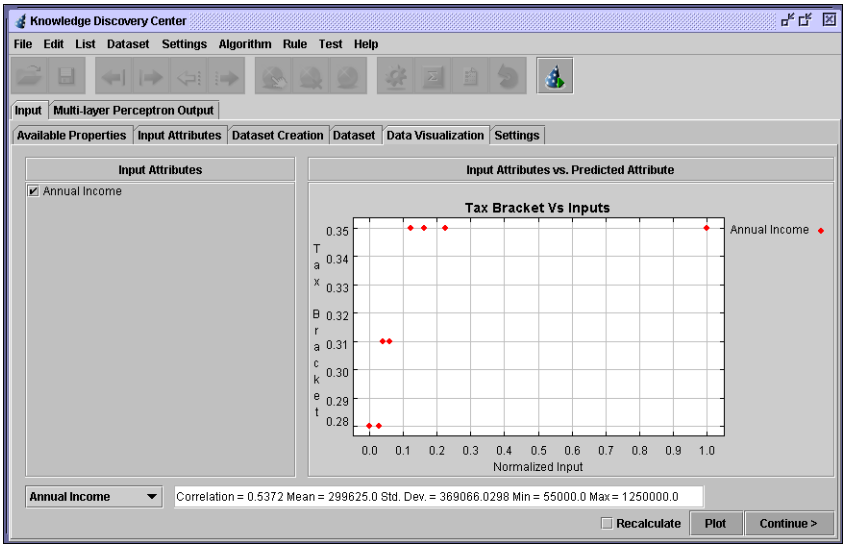


7 Click Continue to open the Data Visualization tab.

Visualizing Your Data

The Data Visualization tab enables you to view the data in graphical form. Specifically, it displays the plotted data in a graph to help you visualize the data to see if there is any correlation between an input and the output. You can also view the correlation, mean, standard deviation, minimum, and maximum values for each of the inputs.

Figure 10-9. Data Visualization Tab



For more information about Data Visualization, see *Visualizing Your Data on page 214*.

1 View the data, and click Continue to open the Settings tab.

Choosing an Algorithm

The Settings tab enables you to choose an algorithm, and adjust the performance of the predictive model by adjusting the settings for that algorithm. An *algorithm* is the actual computational process that takes the input values and produces an output value, the prediction. The Black Pearl Knowledge Broker provides two types of algorithms: Decision Tree, and Multi-Layer Perceptron. In this tutorial, you will generate a Multi-Layer Perceptron.

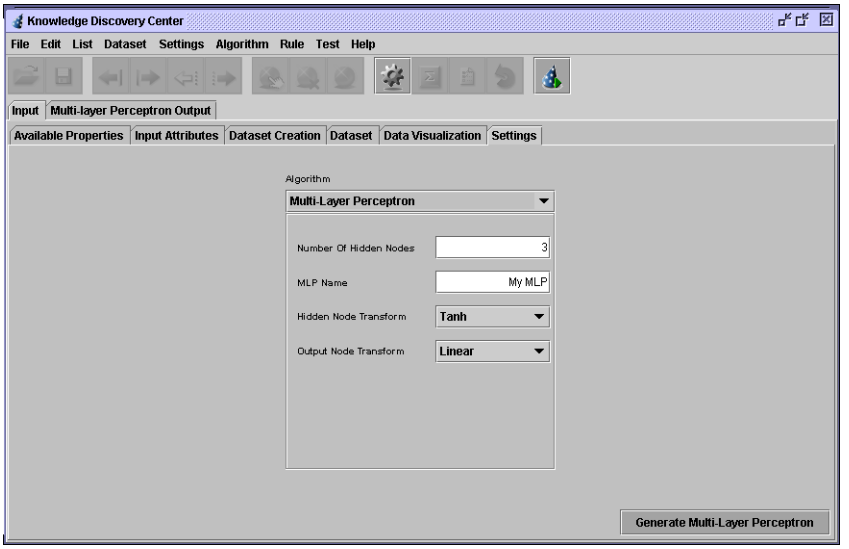
1 Choose Multi-Layer Perceptron from the Algorithm drop-down list.

2 Type 3 in the Number of Hidden Nodes field.

Hidden nodes are objects that receive the input you specify, and produce output. You may guide the predictive model to become more accurate by adding or removing these nodes. There should be a balance of hidden nodes–too many or too few will not result in an optimal predictive model. Furthermore, by default, values over 50 will set the number or hidden nodes to 50.

3 Type My MLP in the MLP Name field.

Figure 10-10. Settings Tab



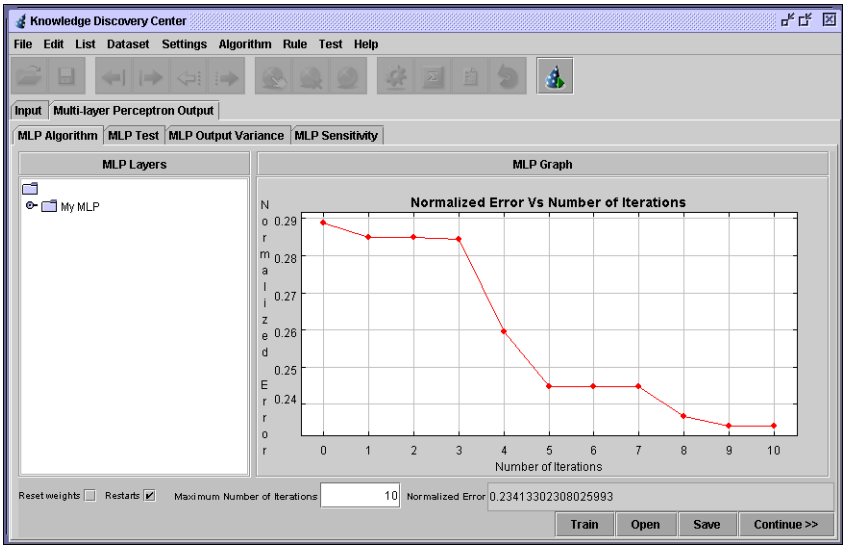
4 Select *tanh* from the Hidden Node Transform drop-down list. The *tanh* function is a special mathematical operation that allows hidden nodes to learn the relationship or pattern between the Annual Income and Tax Bracket.

5 Select Linear from the Output Node Transform drop-down list. This ensures that the output nodes in the MLP you are creating will produce a suitable solution.

At this point, you have set everything you need to build a predictive model--one that can predict a customer's tax bracket based on their annual income.

6 Click the Generate Multi-Layer Perceptron button. This opens the Multi-Layer Perceptron Output tab, generates a perceptron, and displays a graph that lets you view the initial creation of the predictive model.

Figure 10-11. MLP Algorithm Tab



The graph displays the normalized error versus the number of iterations. You can determine whether or not the predictive model is ready to make the prediction based on the Normalized Error value (located on the lower portion of the dialog). A good predictive model will have a Normalized Error value close to zero. The closer this value is to 0, the better the predictive model. A predictive model sometimes reiterates through sets of data many times before it is able to make valid predictions.

Training the Predictive Model

The predictive model must be "trained" to make the most accurate predictions. Therefore, you will need to click the Train button a number of times before your model is ready to make predictions.

Your goal is to move the Normalized Error to a number as close to zero as possible. The normalized error is a measure of the difference between the results your MLP is producing and the results it should be producing. The normalized error range lies between `0` and `1`, with `0` being a "perfect" score, or complete agreement between expected and actual results. Currently the Normalized Error number displays as `0.23`. Let's see if we can improve it.

1. In the MLP Algorithm tab, increase the Maximum Number of Iterations to 100 by typing `100` in the Maximum Number of Iterations field.
2. Click the Train button. After it has gone through 100 iterations, it returns a message informing you how many cycles were trained, the error reduction rate, and a plotted graph of the data.

Figure 10-12. Training Alert

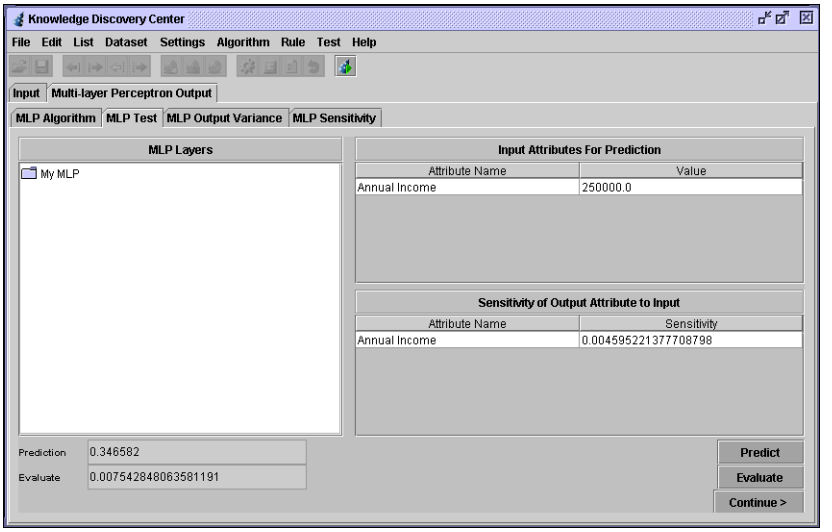
Because an element of randomness always exists in training data, the numbers you see on your screen will not display exactly as shown in this tutorial. They will, however be near the numbers in this tutorial. This does not affect accuracy.

3. Click OK to exit the Information message.
4. Repeat the Train process by clicking the button several more times to "teach" the predictive model until the Normalized Error is closer to `0.10`.

   You will notice that each time you train the model, you receive an alert indicating the error reduction rate, the graph changes slightly, and the number in the Normalized Error field changes. After the value is close to zero (such as `0.10`, for example) you are ready to test your predictive model.
5. Click the Continue button to open the MLP Test tab, where you will test the accuracy of your predictive model.

Testing the Accuracy of Your Predictive Model

Figure 10-13. MLP Test Tab



The MLP Test tab enables you to enter values that let you test the accuracy of the predictive model. While the Normalized Error number gives you a number between 0 and 1, this tab gives you an exact number. You can test the model against what you know about tax brackets and income from the sample data in the example file:

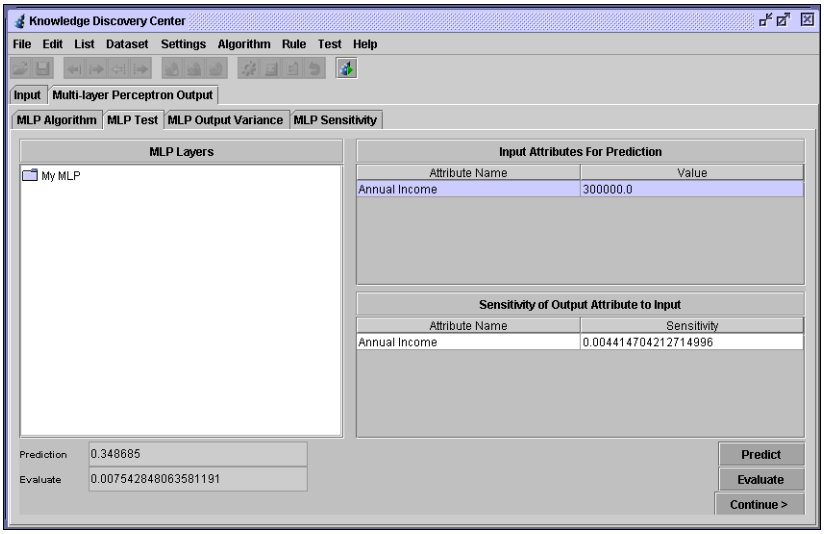
Table 10-1. Sample Data

Annual Income	Tax Bracket
< $50,000	25%
$50,000 - $100,000	28%
$100,000 - $200,000	31%
> $200,000	35%

1 In the Input Attributes for Prediction panel, click to select the Value field. Type 300000 to indicate an annual income of $300,000. (From the sample data, we know customers with an annual income of $300,000 should fall into a tax bracket of 35%.)

2 Click Predict. The Prediction field at the bottom of the dialog indicates a number close to 0.350, or 35%, indicating a tax bracket of 35%.

Figure 10-14. MLP Test Tab



3 Try entering different values in the Annual Income field and using the Predict button to compare the prediction against the sample data in Table 10-1.

Congratulations! You have tested the predictive model and reviewed the results to ensure that your model is accurate, and you have completed the tutorial.

Part

V

Appendices

The Appendices provide a tutorial and reference material, and glossary for the Black Pearl Knowledge Broker.

Appendix

A

Navigating the Black Pearl Knowledge Broker Interfaces

The Black Pearl Knowledge Broker consists of five main user interfaces (UIs), including the:

- Launcher
- Data Concept Editor
- Business Concept Editor
- Rule Editor
- Knowledge Discovery Center

Using these interfaces enables you to identify Datasource Connections, define an ontology, rules, and predictive models relevant to your business, and set the foundation for generating real-time recommendations.

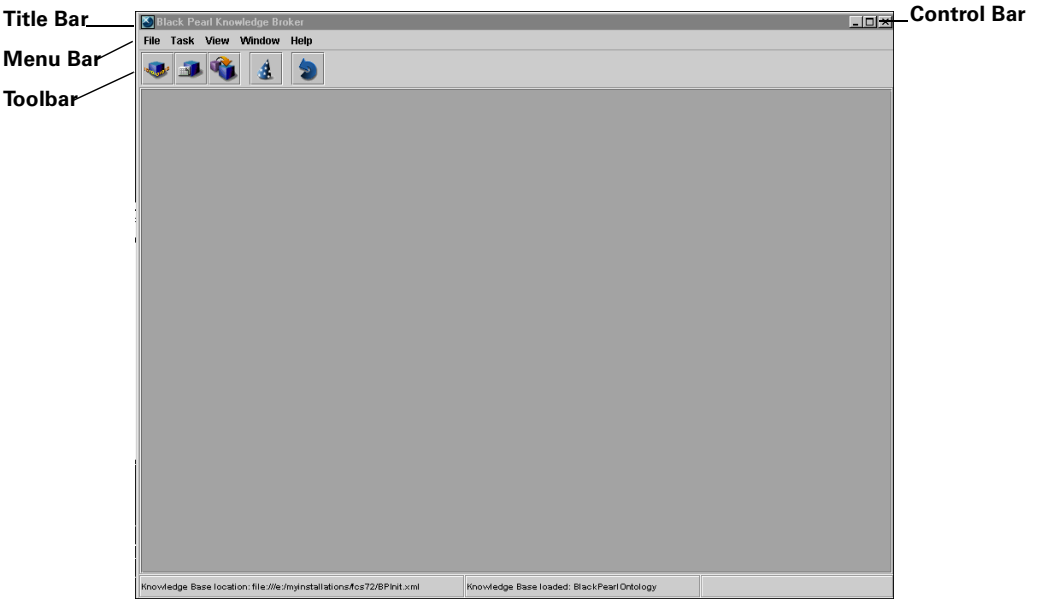
This chapter provides an overview of the Black Pearl Knowledge Broker UIs, menus, toolbars, and command buttons.

Launcher

The *Launcher* enables you to access the various Task Editors, Knowledge Discovery Center, and online Help. It also lets you save your changes to a knowledge base and set viewing parameters. The Launcher displays when you:

- Select Start **>** Programs **>** Black Pearl **>** Knowledge Broker.

Figure A-1. Launcher



The Launcher contains the following elements and functionality:

1 *Title Bar*—identifies the active application.

2 *Control Bar*—minimizes, maximizes, and closes the active application.

3 *Menu Bar*—accesses the following menus:

- *File*—saves your work or exits the application.

You must select File **>** Save on the Black Pearl Knowledge Broker Launcher to record changes to any of the Task Editors.

- *Task*—accesses the Data Concept, Business Concept, Concept Relation, and Rule Editors, plus the Knowledge Discovery Center. Only one Task Editor may be active at any one time; however, multiple windows within a specific Task Editor can be open at the same time.
- View—determines whether to large toolbar icons and/or toolbar labels will display.
- *Window*—identifies which Task Editor windows are currently open, and enables you to tile or cascade open windows.
- *Help*—accesses the Help Index and version and copyright information.

4 *Toolbar*—accesses the Data Concept, Business Concept, Concept Relation, and Rule Editors, plus the Knowledge Discovery Center.

Figure A-2. Launcher Toolbar

Data Concept Editor—add Data Items and manage Concept Items

Rule Editor—define rules

Business Concept Editor—define Business Concepts and filters

Knowledge Discovery Center—define predictive modeling settings

Concept Relation Editor—assign multiple meanings to a single Business Concept

Refresh Ontology—update ontology

Each icon on the Toolbar includes a text-based Tool Tip that enables you to identify the icon's purpose whenever you move your cursor over the icon. Information about the features and functionality of each Editor and the Knowledge Discovery Center is detailed in this chapter.

5 *Workspace*—displays the active Task Editor. Only a single Task Editor may be active at one time; however, multiple windows within an active Task Editor may be open at the same time.

6 *Status Bar*—identifies the server location and active ontology.

Data Concept Editor

The *Data Concept Editor* enables you to add Data Items and manage Concept Items.

Data Items include:

- *Datasource Connections* are URLs, strings, or pathnames to your datasource.
- *Datasources* are the disparate Web, relational database, and flat files containing data pertinent to your business.
- *Datasource Component*s, which are database tables or XML files; for example, Equity. There can be multiple Datasource Components within a single Datasource.
- *Datasource Properties*, which are database columns or XML pages; for example, Beta, Industry, Symbol. There can be multiple Datasource Properties within a single Datasource Component.

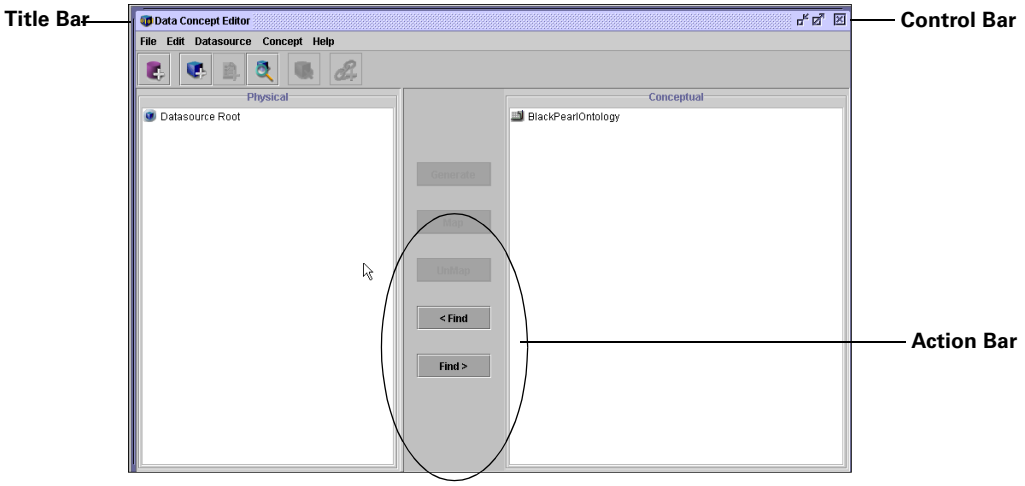
Concept Items include:

- *Data Concepts, which* are vocabulary terms that map directly to one or more of your datasources. For example, *Customers*, *Accounts*, and *Products* can be defined as Data Concepts.
- *Data Concept Properties* are vocabulary terms that function as attributes or defining characteristics for a Data Concept. For example, *Beta*, *Industry*, and *Symbol* can be defined as characteristics of a Stock.

The Data Concept Editor displays when you either:

- Select Task **>** Data Concept Editor in the Launcher.
-  Click the Data Concept Editor icon on the Launcher Toolbar.

Figure A-3. Data Concept Editor



The Data Concept Editor contains the following elements and functionality:

1. *Title Bar*—identifies the active Task Editor.
2. *Control Bar*—minimizes, maximizes, and closes the active Task Editor.
3. *Menu Bar*—accesses the following menus:
   - *File*—closes the Data Concept Editor.
   - Edit—not active.
   - *Datasource*—adds, deletes, finds, and shows details for Data Items.
   - *Concept*—generates, adds, finds, deletes, shows details for, renames, maps, and unmaps Data Concepts and Data Concept Properties; and adds, modifies, or deletes Calculated Properties.
   - *Help*—accesses the online Help Index.
4. *Toolbar*—adds Datasource Connection, Data Concept, or Data Concept Property; finds or deletes Concept Items; and maps Data Items to Concept Items.

Figure A-4. Data Concept Editor Toolbar

Add Connection—define path to a Datasource

Find Data Concept—search for a specific Data Concept or Property

Add Data Concept—create concept that maps to a Datasource

Delete Data Concept—remove a specific Data Concept or Property

Add Property—create attribute that maps to a Data Concept

Add Mapping—define relationship between physical Datasources and abstract Data Concepts and Properties

Each icon on the Toolbar includes a text-based Tool Tip that enables you to identify the icon's purpose whenever you move your cursor over the icon.

5. *Action Bar*—finds Data Items; and generates, maps, unmaps, and finds Concept Items.

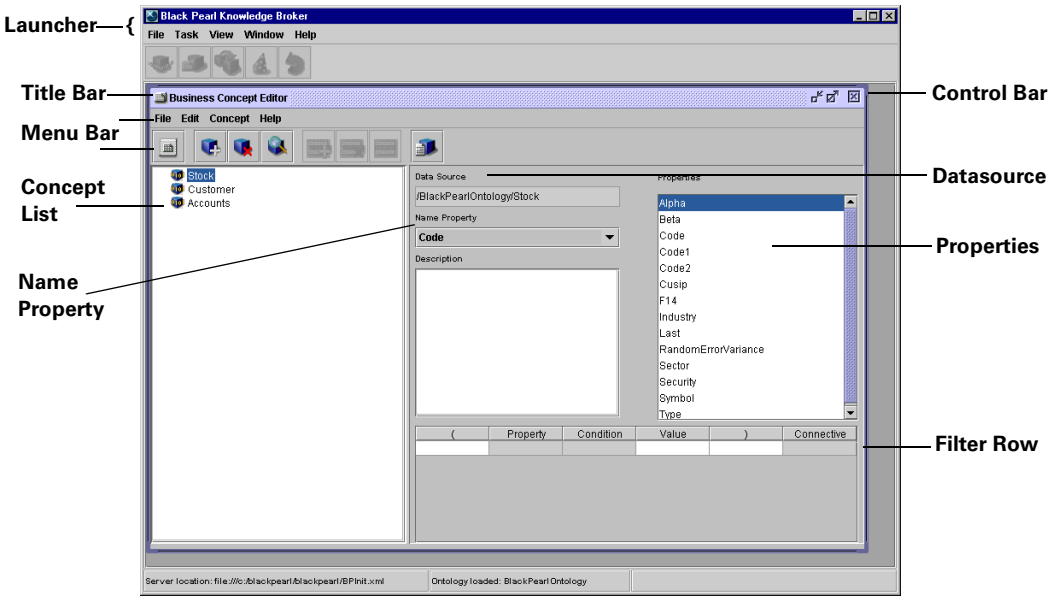
Business Concept Editor

The *Business Concept Editor* enables you to define your Business Concepts and Concept Filters. *Business Concepts,* which are vocabulary terms relevant to your business, function as sub-categories of a Data Concept or another Business Concept. Each Business Concept is differentiated from its Parent Concept by filters. A *filter* specifies the parameters that define the Business Concept. For example, if you add a Volatile Business Concept that is a sub-category of the Stock Data Concept, you must specify the criteria that differentiate Volatile Stock from all other Stock.

The Business Concept Editor displays when you either:

- Select Task **>** Business Concept Editor in the Launcher.
- Click the Business Concept Editor icon on the Launcher Toolbar.

Figure A-5. Business Concept Editor



The Business Concept Editor contains the following elements and functionality:

1. *Title Bar*—identifies the active Task Editor.
2. *Control Bar*—minimizes, maximizes, and closes the active Task Editor.
3. *Menu Bar*—accesses the following menus:
   - *File*—previews and prints a report of all Business Concepts and closes the Business Concept Editor.
   - Edit—not active.
   - *Concept*—adds, deletes, renames, and finds Business Concepts; adds, deletes or clears Business Concept filters; and finds Instances for a Business Concept.
   - *Help*—accesses the online Help Index and further information.
4. *Toolbar*—accesses a print preview; adds, deletes, and finds Business Concepts; adds, deletes, or clears filters; and displays Instances for a Business Concept.

Figure A-6. Business Concept Editor Toolbar


Print Preview—review a report before printing


Add Concept—define a new Business Concept


Delete Concept—remove a specific Data or Business Concept


Find Concept—search for a specific Data or Business Concept


Add Filter Row—define parameters for a specific Business Concept


Delete Filter Row—remove a filter from a specific Business Concept


Clear Filter Row—remove current parameters and enables modifications


Show Instances—display individual records for a specific Business Concept


Each icon on the Toolbar includes a text-based Tool Tip that enables you to identify the icon's purpose whenever you move your cursor over the icon.

5. *Datasource*—identifies the Datasource for a Business Concept, and add a description for each Property associated with a Business Concept.
6. Name Property—selects the sorting option for the Business Concept.
7. Description—enables you to type keywords describing the Business Concept.
8. *Properties*—displays a list of all Properties associated with a specific Business Concept.
9. *Filter Rows*—enable you to add filters for a specific Business Concept.

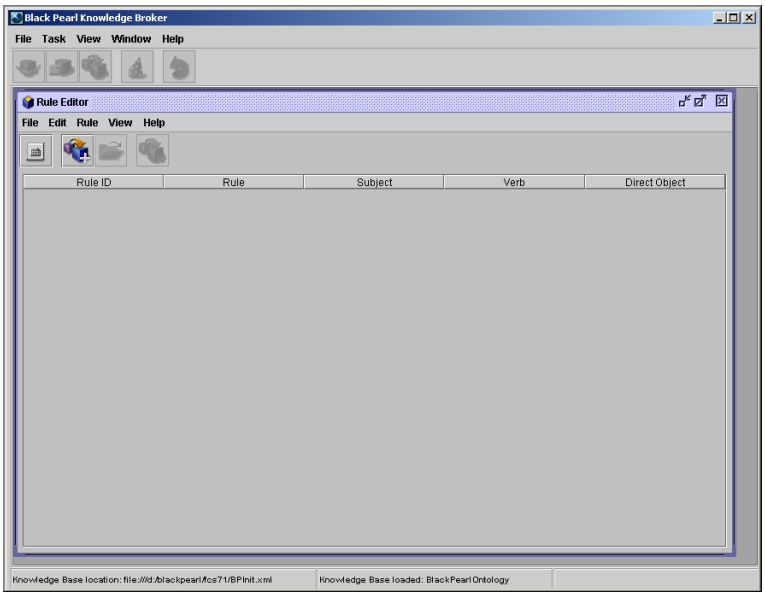
Rule Editor

The *Rule Editor* enables you to define rules. A *rule* is a logical statement that describes two things: how your business typically responds to individual transactions and what situations will require different responses in seemingly similar situations. Rules are the means for "teaching" the computer how to respond to situations relevant to your business requirements, and are the basis for generating real-time recommendations.

The Rule Editor displays when you either:

- Select Task **>** Rule Editor in the Launcher.
- Click the Rule Editor icon on the Launcher Toolbar.

Figure A-7. Rule Editor



The Rule Editor contains the following elements and functionality:

1. *Title Bar*—identifies the active Task Editor.
2. *Control Bar*—minimizes, maximizes, and closes the active Task Editor.
3. *Menu Bar*—accesses the following menus:
   - *File*—previews and prints a report of all rules and closes the Rule Editor.
   - Edit—not active.
   - *Rule*—adds, edits, or deletes a rule.
   - View—enables you to select which columns are viewable in the Rule Editor.
   - *Help*—accesses the online Help Index and information about the Rule Editor.
4. *Toolbar*—previews and prints a report, and adds, edits, or deletes a rule.

Figure A-8. Rule Editor Toolbar


Print Preview—review a report before printing


Edit Rule—open the Edit Rule Editor


Add Rule—open the Add New Rule Editor


Delete Rule—remove a specific rule from the Rulebase


Each icon on the Toolbar includes a text-based Tool Tip that enables you to identify the icon's purpose whenever you move your cursor over the icon.

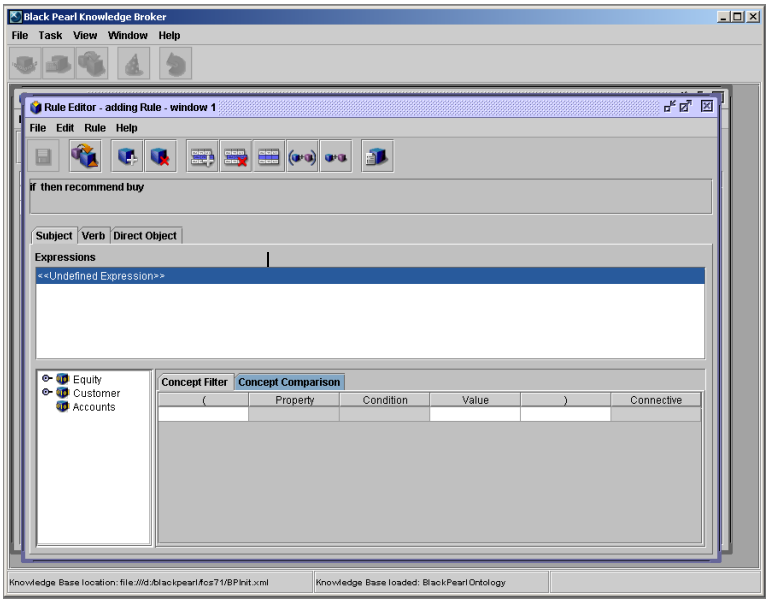
5. *Rule Rows*—displays each rule currently saved in the knowledge base.

New Rule Editor

The *New Rule Editor*, which is a part of the Rule Editor, enables you to add a new rule. It displays when you either:

- Select Rule **>** Add Rule in the Rule Editor.
- Click the Add Rule icon on the Rule Editor Toolbar.

Figure A-9. New Rule Editor



The New Rule Editor contains the following elements and functionality:

1. *Title Bar*—identifies the New Rule Editor, and Rule ID.
2. *Control Bar*—minimizes, maximizes, and closes the active New Rule Editor.
3. *Menu Bar*—accesses the following menus:
   - *File*—saves a rule and closes the Rule Editor 1.
   - Edit—not active.
   - *Rule*—clears a rule; adds, deletes, clears, groups, or ungroups a filter; displays Rule Instances.
   - *Help*—accesses the online Help Index and information about the Rule Editor.
4. Toolbar—saves or clears a rule; adds, deletes, or clears a filter; groups or ungroups set of filters; and displays Rule Instances.

Each icon on the Toolbar includes a text-based Tool Tip that enables you to identify the icon's purpose whenever you move your cursor over the icon.

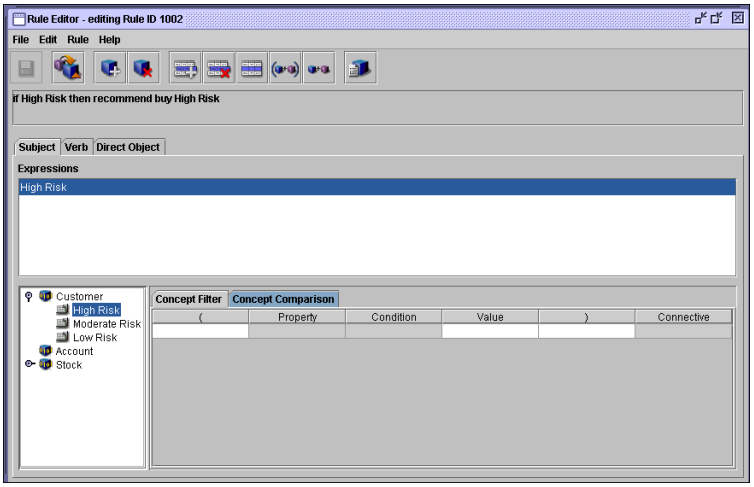
5. Rule Panel—displays the rule.
6. *Concept Filter*—enables you to create and edit filters for the rule.
7. *Concept Comparison*—allows you to directly compare two properties of two different concepts.

Change Rule Editor

The *Change Rule Editor*, which is a part of the Rule Editor, enables you to edit an existing rule. The Existing Rule Editor displays when you either:

- Select Rule **>** Edit Rule in the Rule Editor.
- Click the Edit Rule icon on the Rule Editor Toolbar.

Figure A-10. Change Rule Editor



The Change Rule Editor contains the following elements and functionality:

1. *Title Bar*—identifies the active Task Editor.
2. *Control Bar*—minimizes, maximizes, and closes the active Task Editor.
3. *Menu Bar*—accesses the following menus:
   - *File*—saves a rule and closes the Rule Editor 1.
   - Edit—not active.
   - *Rule*—clears a rule; adds, deletes, clears, groups, or ungroups a filter; displays Rule Instances.
   - *Help*—accesses the online Help Index and information about the Rule Editor.
4. Toolbar—saves or clears a rule; adds, deletes, or clears a filter; groups or ungroups set of filters; and displays Rule Instances.

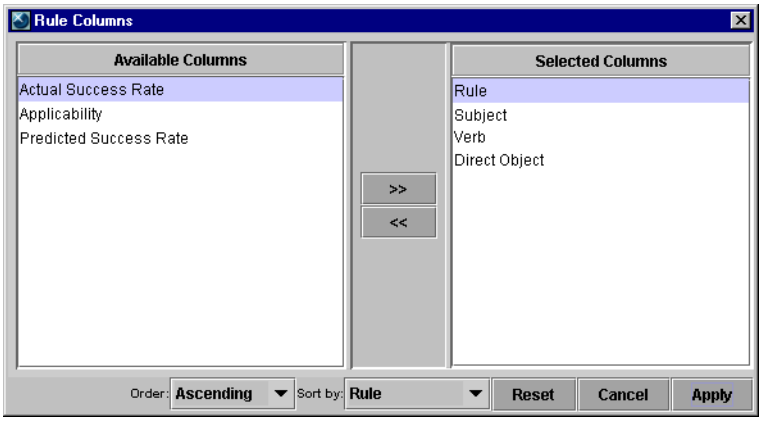
Each icon on the Toolbar includes a text-based Tool Tip that enables you to identify the icon's purpose whenever you move your cursor over the icon.

5. Rule Panel—displays the rule.
6. *Tabs*—access options for each part of a rule (Subject, Verb, and Direct Object).
7. *Concept Filter*—enables you to create and edit filters for the rule.
8. *Concept Comparison*—allows you to directly compare two properties of two different concepts.

Edit Rule Columns

The Edit Rule Columns enables you to select which items display in the Rule Editor. The Edit Rule Columns displays when you select View **>** Edit Columns.

Figure A-11. Edit Rule Columns



The Edit Rule Columns contains the following elements and functionality:

1. *Available Columns*—lists information columns available but not displayed in the Rule Editor.
2. *Selected Columns*—lists the information columns currently displayed in the Rule Editor.
3. >>—moves an Available Column to the Selected Column.
4. <<—moves a Selected Column to the Available Column.
5. *Order*—determines whether rules will be listed in Ascending or Descending order.
6. *Sort By*—determines whether the rules will be sorted by Subject, Verb, or Direct Object.
7. *Action Bar*—resets the Rule Editor to the last saved settings; or cancels or applies the changes made in the Edit Rule Column window.

Knowledge Discovery Center

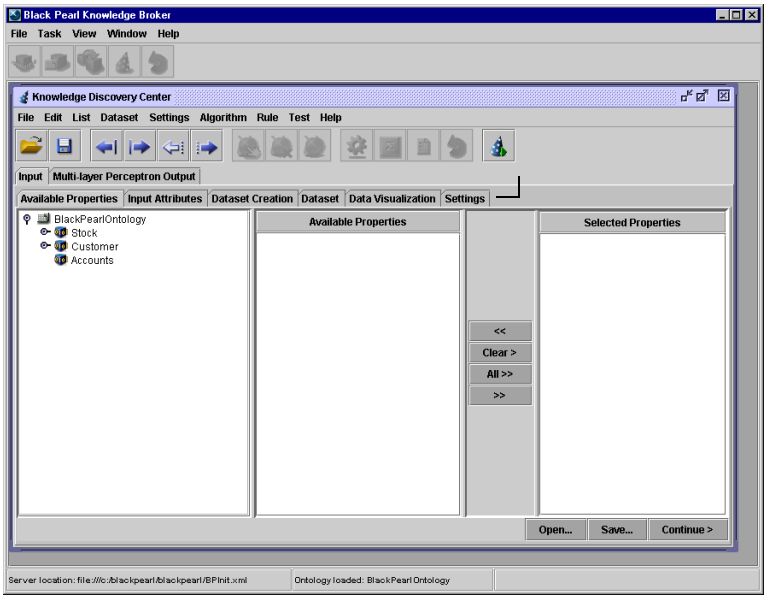
The *Knowledge Discovery Center* enables you to select the concepts, attributes, dataset sample size, sample methods, and other settings used in predictive modeling. A *predictive model* is a data-mining process whereby a mathematical function is used to analyze your data to discover metadata and make predictions about actions, behavior, or requirements. It is a method for "learning by example" and consists of three objects:

- An input value, known as the *independent* or *predictor variable*, that consists of a representative sample of your raw data.
- A mathematical function, known as an *algorithm*, that is applied to the independent variable to discover patterns and relationships.
- An output value, known as the *dependent* or *target variable*, which consists of predicted values for other data. The output value is the actionable recommendation.

The Knowledge Discovery Center displays when you either:

- Select Task **>** Knowledge Discovery Center in the Launcher.
- Click the Knowledge Discovery Center icon on the Launcher Toolbar.

Figure A-12. Knowledge Discovery Center



The Knowledge Discovery Center consists of the following elements and functionality:

1. *Title Bar*—identifies the active Task Editor window.
2. *Control Bar*—minimizes, maximizes, and close the active Task Editor window.
3. *Menu Bar*—accesses the following menus:
   - *File*—opens or saves an Attribute file, and closes the Knowledge Discovery Center.
   - Edit—not active.
   - *List*—moves a list item left or right, clears all choices, clears a choice or moves it to the right, and sets a prediction Attribute.
   - Dataset—displays or appends a Dataset.
   - Settings—generates a Decision Tree predictive model.
   - Algorithm—sets advanced parameters and properties.
   - Rule—prunes, updates, or saves a derived rule.
   - Test—tests a predictive model, or create a script to generate a predictive model.
   - *Help*—accesses the online Help Index and information about the Knowledge Discovery Center.
4. Toolbar—accesses
5. *Tabs*—access Input and Output Editors.
   - *Input Editors* include: Available Properties, Input Attributes, Dataset Creation, Dataset, Data Visualization, and Settings.
   - *Output Editors* include: MLP Algorithm, MLP Test, MLP Output Variance, and MLP Sensitivity, or Decision Tree Algorithm, Production Rules, and Test.

The functionality for each of the Input and Output tabs is described in the next sections.

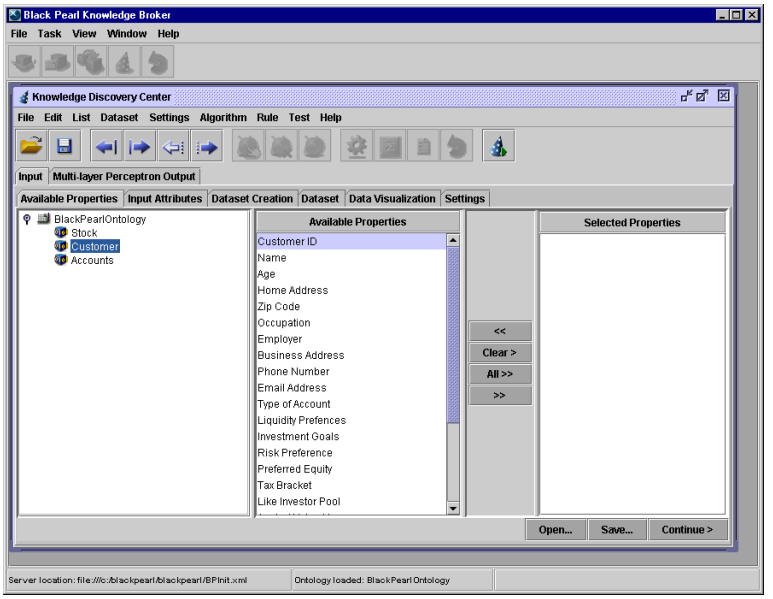
Available Properties Editor

The *Available Properties Editor* enables you to view the Data or Business Concepts properties that can be used in the predictive modeling process.

The Available Properties Editor displays when you either:

- Select Task **>** Knowledge Discovery Center in the Launcher.
- Click the Available Properties tab in the Knowledge Discovery Center.

Figure A-13. Available Properties Editor



The Available Properties Editor consists of the following elements and functionality:

1. *Ontology*—displays all Data and Business Concepts currently defined within your ontology.
2. *Available Properties*—displays a list of all properties associated with a specific Data or Business Concept.
3. *Selected Properties*—displays a list of all properties selected for use in the predictive modeling process.
4. <<—moves selected items from the Selected Properties to Available Properties.
5. >>—moves selected items from Available Properties to Selected Properties.
6. *Clear* >—moves all items from Selected Properties to Available Properties.
7. *All* >>—moves all items from Available Properties to Selected Properties.
8. *Open*—accesses a previously saved Attribute Dataset.
9. *Save*—saves the current Selected Properties to an Attribute Dataset.
10. *Continue*—accesses the Input Attributes Editor and temporarily stores your selections in the Knowledge Discovery Center.
11. *List*—accesses options for moving items between Available and Selected Properties.

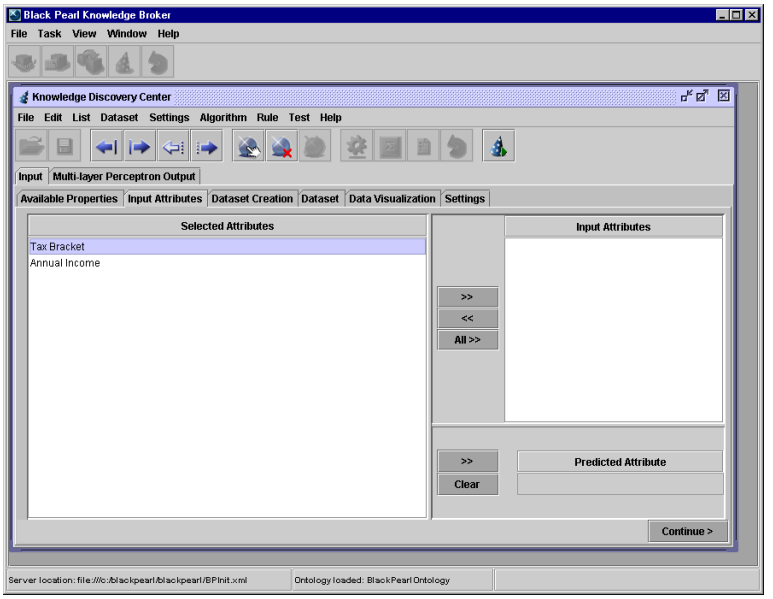
Input Attributes Editor

The *Input Attributes Editor* enables you to select the attributes used in the predictive modeling process.

The Input Attributes Editor displays when you either:

- Click Continue on the Available Properties Editor.
- Click the Input Attributes tab in the Knowledge Discovery Center.

Figure A-14. Input Attributes Editor



The Input Attributes Editor consists of the following:

1. Selected Attributes—displays a list of all Attributes currently selected for use in the predictive model.
2. Input Attributes—displays a list of what values you want the Black Pearl Knowledge Broker to learn.
3. Predictive Attribute—displays the value to be predicted.
4. >>—moves selected items from Selected Attributes to Input Attributes.
5. <<—moves selected items from Input Attributes to Selected Attributes.
6. All >>—moves all items from Selected Attributes to Input Attributes.
7. >>—moves a selected item from Selected Attributes to Predictive Attributes.
8. Clear—moves the Predicted Attribute to Selected Attributes.
9. *Continue*—accesses the Dataset Creation Editor and temporarily stores your selections in the Knowledge Discovery Center.
10. *List* menu—accesses options that enables moves items between panels.

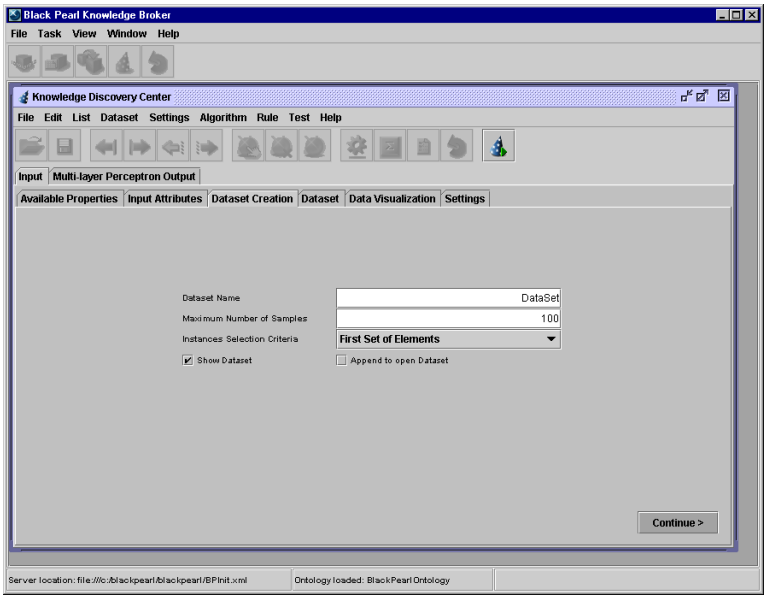
Dataset Creation Editor

The *Dataset Creation Editor* enables you to select the criteria for how the "learn by example" dataset will be created.

The Dataset Creation Editor displays when you either:

- Click Continue in the Input Attributes Editor.
- Click the Dataset Creation tab in the Knowledge Discovery Center.

Figure A-15. Dataset Creation Editor



The Dataset Creation Editor consists of the following elements and functionality:

1. *Dataset Name*—displays the name for the dataset. The default name is Dataset, which you can modify by typing a new name in the Dataset name field.
2. *Maximum Number of Samples*—specifies the number of samples to be used in the training set. The default sample size is 100, which you can modify by typing a new number in the Samples field.
3. *Instances Selection Criteria*—specifies which set of data will be used for the training set. There are four options within this drop-down list:
   - *First Set of Samples*—uses datasource Instances 1 through *n* (where *n* equals the number specified in the Maximum Number of Samples field). For example, if you specified 100 samples, then the first 100 Instances will be queried.
   - *Elements After Instance n*—uses a variable set by the Black Pearl Knowledge Broker based upon the total number of datasource Instances that are already in the knowledge base. For example, if you specified 100 samples, then Instances 2 through 102 will be queried.

Selecting either of these two criteria is a simple way to deal with a large datasource. However, optimal results may be produced.

   - *Uniform Probability Distribution*—uses a statistical parameter of *1/n* to create a small, representative dataset from within a larger dataset. Each sample in the Datasource is selected for inclusion in the Dataset, based on the specified statistic.
   - *Normal Probability Distribution*—uses a bell-curve statistical parameter to create a dataset. The exact shape of the bell-curve is determined by the mean and variance of your data. *Mean* is the representative average for the entire dataset. *Variance* is a measure of the spread within a dataset. For example, if you selected Age as an Input Attribute and you want to make a prediction concerning people *around* age 40, then you can set a Mean of 40 and a Variance of 2. The training set will include Instances that are normally distributed, with a Mean = 40, and a Variance = 2.
4. *Show Dataset* —specifies whether or not the dataset displays in the Dataset Editor.
5. *Append to Open Dataset*—specifies whether or not the current dataset is added to a previously saved dataset.
6. *Continue*—accesses the Dataset Editor and temporarily stores your selections in the Knowledge Discovery Center.
7. *Dataset*—accesses Show Dataset and Append to Dataset functionality.

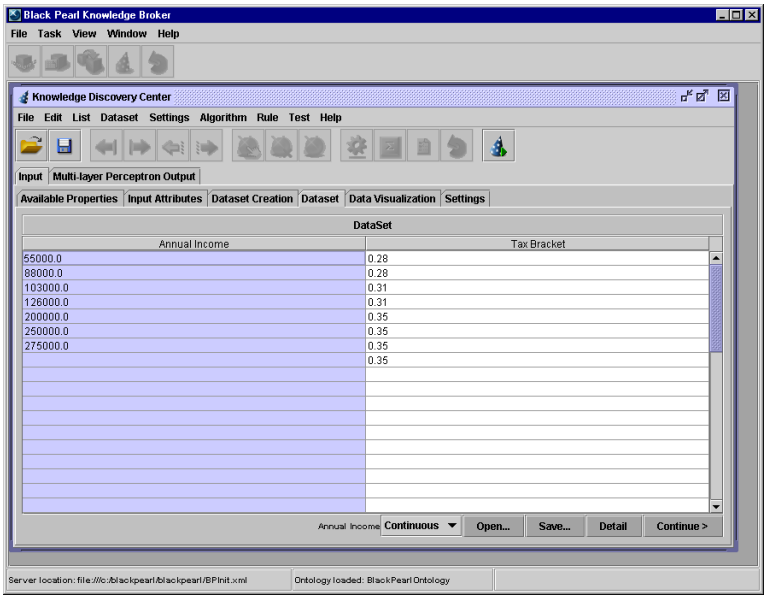
Dataset Editor

The *Dataset Editor* enables you to view the Instances that will be used in the predictive modeling process, and change Attributes from Continuous to Discrete. *Continuous* is any value within a finite or infinite interval; for example, age, weight, height. *Discrete* is a value that is distinct and countable; for example, the number of months in a calendar year, the number of employees in a company, the number of scenes in a movie.

The Dataset Editor displays when you either:

- Click Continue in the Dataset Creation Editor.
- Click the Dataset tab in the Knowledge Discovery Center.

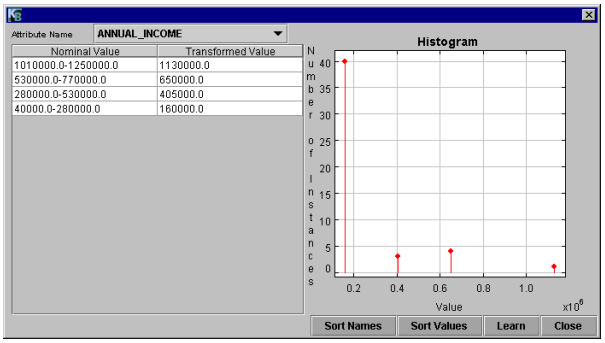
Figure A-16. Dataset Editor



The Dataset Editor consists of the following:

1. *Dataset*—displays all Instances being used in the predictive modeling process.
2. *Attribute Name*—enables the changing of an Attribute from Continuous to Discrete. MLP algorithms only work with Continuous Attributes.
3. *Open*—accesses a previously saved dataset.
4. *Save*—save the current dataset.
5. *Detail*—displays a Histogram of the data. A *histogram* groups the datasource Instances into bins, plotting the number of members in each bin versus the bin number. For example, if you had a dataset with 100 datasource Instances with a Uniform Distribution from 0 and 10, a histogram would create 10 bins, with bin one containing datasource Instances 0-1, bin two containing datasource Instances 1-2, bin three containing datasource Instances 2-3, etc. The histogram changes to visually reflect changes in the relationships between Instances and Values in the dataset.

Figure A-17. Dataset Editor



6. *Continue*—accesses the Data Visualization Editor and temporarily stores your selections in the Knowledge Discovery Center.

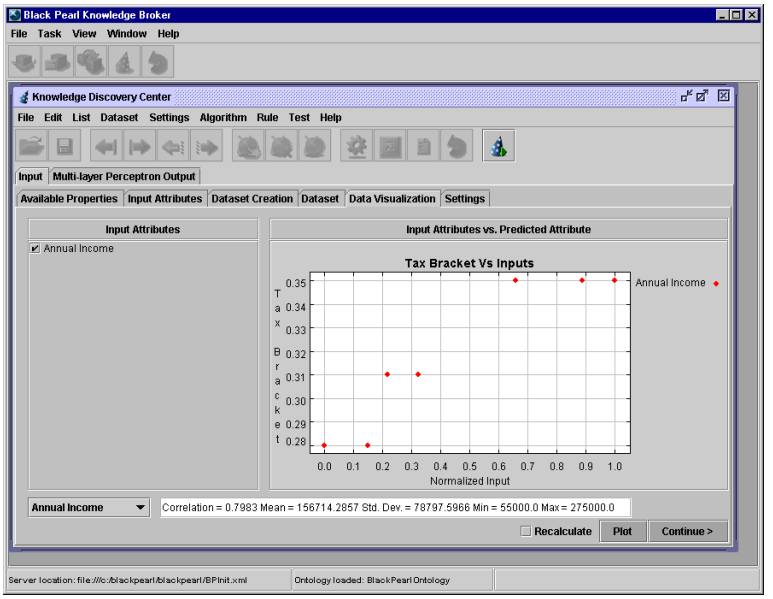
Data Visualization Editor

The *Data Visualization Editor* enables you to view the distribution of data used in the predictive modeling process and determine the correlation between Input Attributes and the Predictive Attribute. The Data Visualization Editor also enables you to view the Mean, Variance, Minimum and Maximum values for each Input Attribute.

The Dataset Editor displays when you either:

- Click Continue in the Dataset Creation Editor.
- Click the Data Visualization tab in the Knowledge Discovery Center.

Figure A-18. Data Visualization Editor



The Data Visualization Editor consists of the following elements and functionality:

1. *Input Attributes*—selects Input Attributes to be plotted in the Input Attributes vs. Predicted Attributes graph.
2. *Risk Preferences vs. Inputs Graph*—displays the correlation between the Input Attributes and Predicted Attribute.
3. *Discretized Predicted Attribute*—displays the values for the Predictive Attribute.
4. *Input Attributes*—selects an Input Attribute, and displays the Correlation, Mean, Standard Deviation, Minimum, and Maximum Values.
5. *Plot*—re-plots the graph in the Input Attributes vs. Predicted Attributes graph, if you change the selection in the Input Attributes panel.
6. *Continue*—accesses the Settings Editor and temporarily stores your selections in the Knowledge Discovery Center.

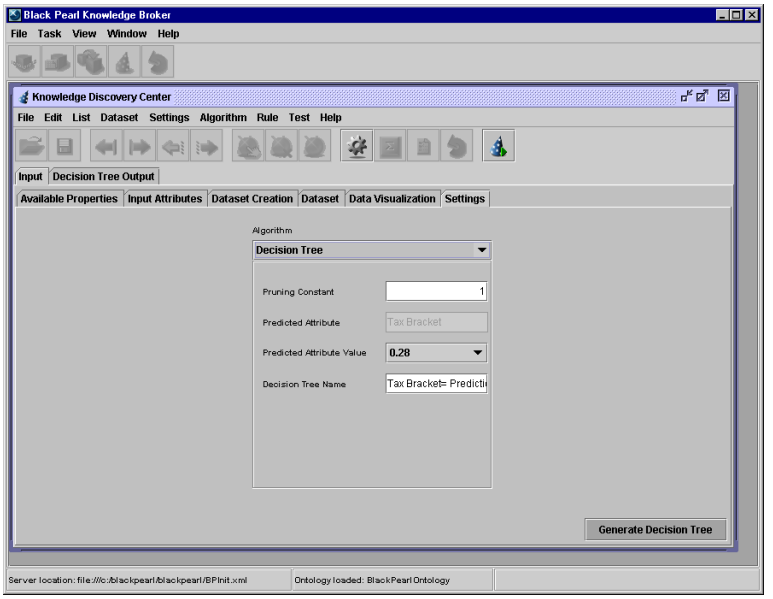
Decision Tree Settings Editor

The *Settings Editor* enables you to select the algorithm to be used in the predictive modeling process. Currently, a Decision Tree and Multi-layer perceptron (MLP) algorithm are supported in the Black Pearl Knowledge Broker. Depending on which algorithm selected, different items display.

The Settings Editor displays when you either:

- Click Continue in the Data Visualization Editor.
- Click the Settings tab in the Knowledge Discovery Center.

Figure A-19. Decision Tree Settings Editor



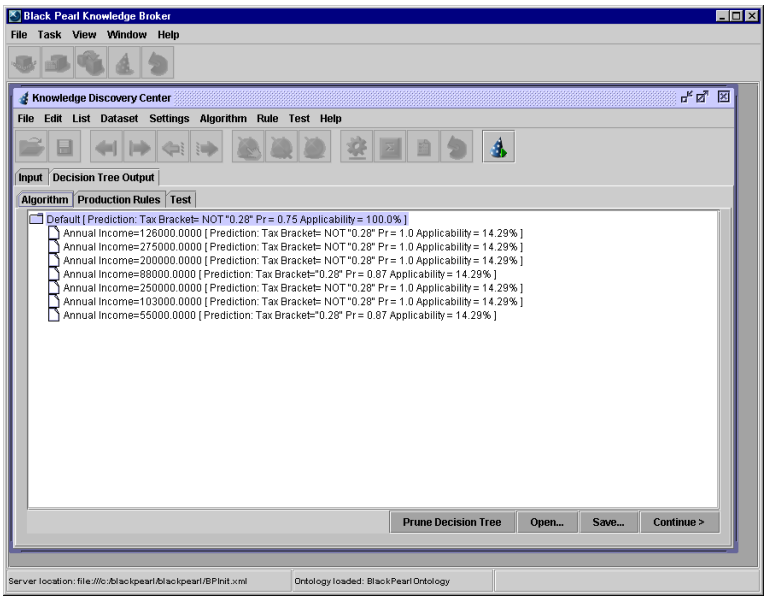
The Decision Tree Settings Editor displays if you select Decision Tree from the Algorithm drop-down list in the Settings Editor. It consists of the following:

1. *Pruning Constant*—sets a value of -0 or 1. Low values reduce pruning severity, while values nearer to and including 1 increase the pruning severity. Pruning removes duplicate rules.
2. *Predicted Attribute*—displays the item selected as the Predicted Attribute.
3. *Predicted Attribute Value*—provides a list of values.
4. *Decision Tree Name*—defines a customized name for your dataset.
5. *Generate Decision Tree*—instructs the Knowledge Discovery Center to create a Decision Tree based on all the defined parameters.

Decision Tree Algorithm Editor

The *Decision Tree Algorithm Editor* enables you to prune the algorithm rules to be used in the predictive modeling process. The Decision Tree Algorithm Editor displays when you click Generate Decision Tree in the Settings Editor

Figure A-20. Decision Tree Algorithm Editor



The Decision Tree Algorithm Editor consists of the following:

1. *Decision Tree*—displays default and leaf nodes for the Decision Tree.
2. *Prune Decision Tree*—remove duplicate nodes.
3. *Open*—accesses a previously saved Decision Tree.
4. *Save*—saves the current Decision Tree.
5. *Continue*—access the Production Rules Editor and temporarily stores your selections in the Knowledge Discovery Center.

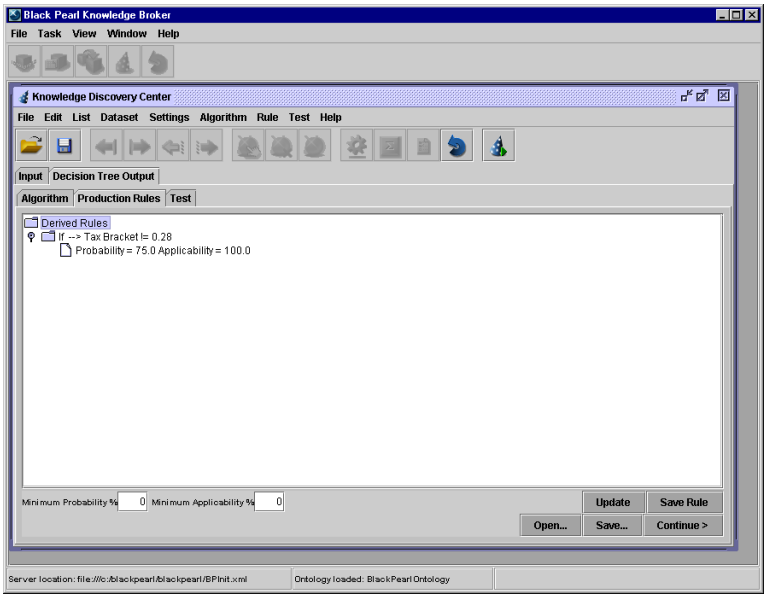
Decision Tree Production Rule Editor

The *Decision Tree Production Rule Editor* enables you to view the rules that can be created as result of the predictive modeling process.

The Decision Tree Production Rule Editor displays when you either:

- Click Continue in the Decision Tree Algorithm Editor.
- Click the Production Rules tab in the Knowledge Discovery Center (Decision Tree Output)

Figure A-21. Decision Tree Production Rule Editor



The Decision Tree Production Rule Editor consists of the following elements and functions:

1. *Rules*—displays a list of all rules that can be created as a result of the predictive modeling process.
2. *Minimum Probability %* —defines the minimum probability parameter.
3. *Minimum Applicability %*—defines the minimum applicability parameter.
4. *Update*—updates the Production Rule set.
5. *Save Rule*—saves a selected rule to a Production Rule set.
6. *Open*—opens a previously saved Production Rule set.
7. *Save*—saves the current Production Rule set.
8. *Continue*—accesses the Decision Tree Test Editor and temporarily stores your selections in the Knowledge Discovery Center.

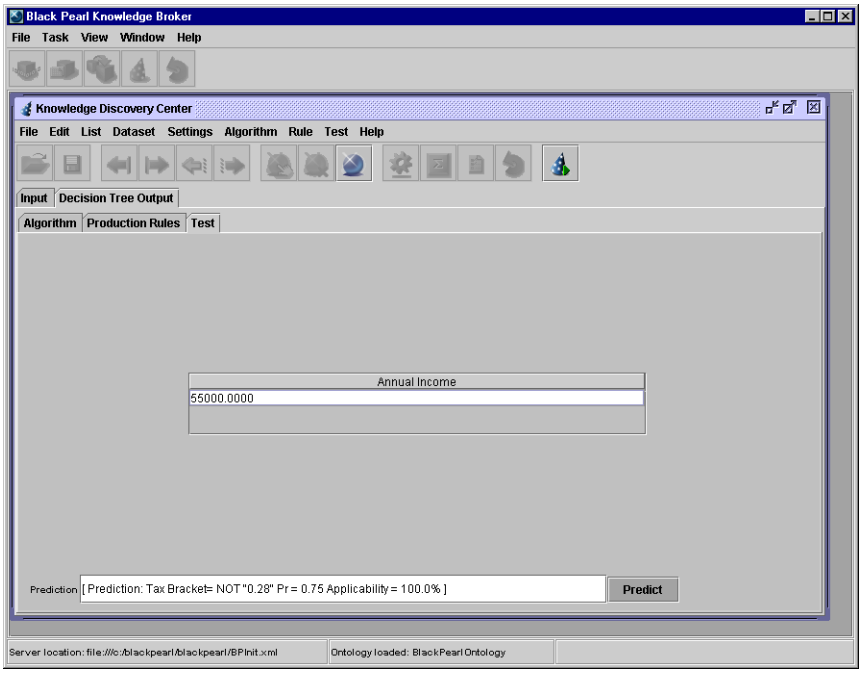
Decision Tree Test Editor

The *Decision Tree Test Editor* enables you to set the test value and view the results of the predictive modeling process.

The Decision Tree Test Editor displays when you either:

- Click Continue in the Decision Tree Production Rule Editor.
- Click the Test tab in the Knowledge Discovery Center (Decision Tree Output).

Figure A-22. Decision Tree Test Editor

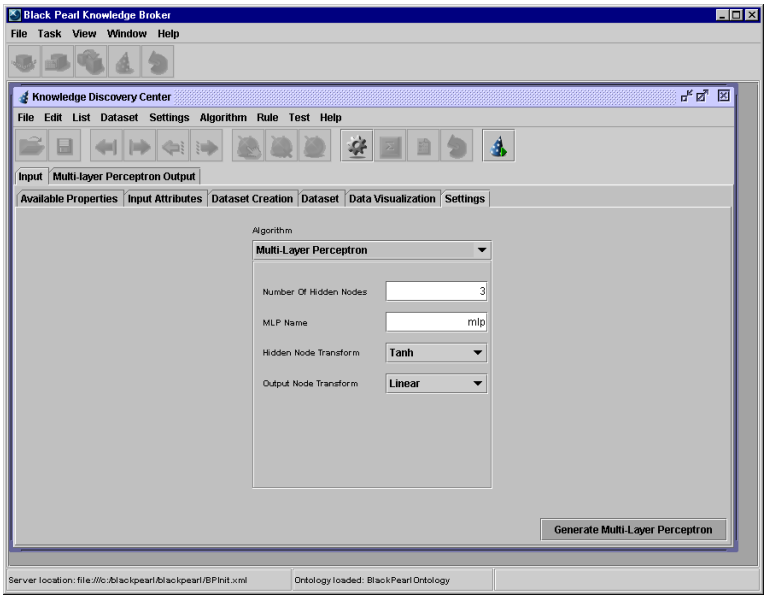


The Decision Tree Test Editor consists of the following:

1 *Attributes Test Value*—defines the test value for each Input Attribute.
2 *Predict*—runs the predictive modeling process.
3 *Prediction*—displays the results.

MLP Settings Editor

Figure A-23. MLP Settings Editor



The MLP Settings Editor consists of the following elements and functionality:

1. *Number of Hidden Nodes*—defines how many hidden layers will be used in the predictive modeling process. Hidden nodes create interrelationships between Input and Predicted Attributes. The higher the number of nodes, the greater the accuracy of the MLP. However, too many nodes can increase the training time and over-train the MLP so that it can only work with the sample data.
2. *MLP Name*—defines a customized name for your dataset.
3. *Hidden Nodes Transform*—defines whether the MLP will use a Tanh, Sigmoid, or Linear hidden node transformation.
4. *Output Node Transform*—defines whether the MLP will use a Tanh, Sigmoid, or Linear output node transformation.

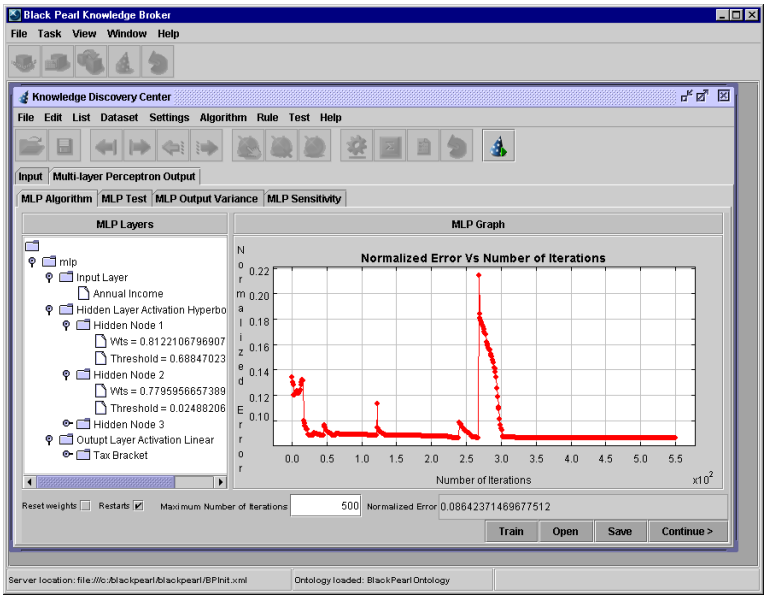
MLP Algorithm Editor

The *MLP Algorithm Editor* enables you to view information about the MLP layers used in the predictive modeling process, view a graph plotting the learning curve, and train the Black Pearl Knowledge Broker.

The MLP Algorithm Editor displays when you either:

- Click Continue in the MLP Settings Editor.
- Click the MLP Algorithm tab in the Knowledge Discovery Center (Multi-layer Perceptron Output).

Figure A-24. MLP Algorithm Editor



The MLP Algorithm Editor consists of the following elements and functionality:

1. *MLP Layers*—displays a list of all layers used in the MLP.
2. *Normalized Error vs. Number of Iterations graph*—displays the learning curve.
3. *Reset Weights*—re-trains the network. The weights are re-initialized at random.
4. *Maximum Number of Iterations*—defines how many learning cycles the MLP can undertake.
5. *Normalized Error*—displays the error rate for the dataset.
6. *Train*—instructs the Black Pearl Knowledge Broker to begin another learning iteration.
7. *Open*—opens a previously saved MLP.
8. *Save*—saves the current MLP.
9. *Continue*—accesses the MLP Test Editor and temporarily stores your selections in the Knowledge Discovery Center.

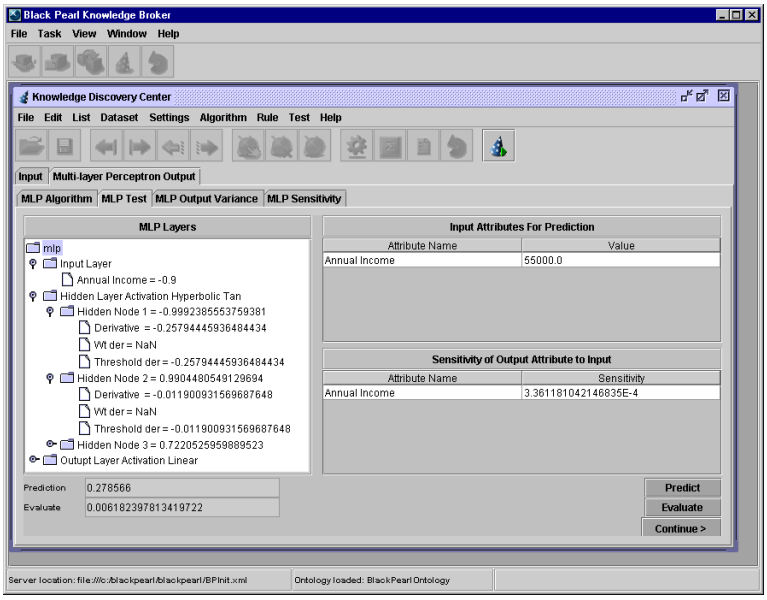
MLP Test Editor

The *MLP Test Editor* enables you to view information derived as a result of the training and to enter values to test the predictive model. This screen enables the user to evaluate the MLP. The upper right-hand panel displays inputs to the neural network. The left-hand panel displays the output for all nodes in the MLP. The lower right-hand panel displays a list indicating how sensitive the output is to a 1% change in the input. This sensitivity metric is useful in determining how sensitive each output is in relation to each input.

The MLP Test Editor displays when you either:

- Click Continue in the MLP Algorithm Editor.
- Click the MLP Test tab in the Knowledge Discovery Center (Multi-layer Perceptron Output).

Figure A-25. MLP Test Editor



The MLP Test Editor consists of the following:

1. *MLP Layers*—displays derived information for all MLP layers.
2. *Prediction*—displays the current prediction value.
3. *Evaluate*—displays the current absolute error of the dataset defined in the Input tab.
4. *Predicted Value*—displays the absolute error value between the Input and Predicted Attributes.
5. *Attribute Sensitivity*—displays the sensitivity value between the Input and Predicted Attributes.
6. *Predict*—changes the inputs and view the new prediction.
7. *Evaluate*—enables you to evaluate the results of the MLP.
8. *Continue*—accesses the MLP Output Variance Editor and temporarily stores your selections in the Knowledge Discovery Center.

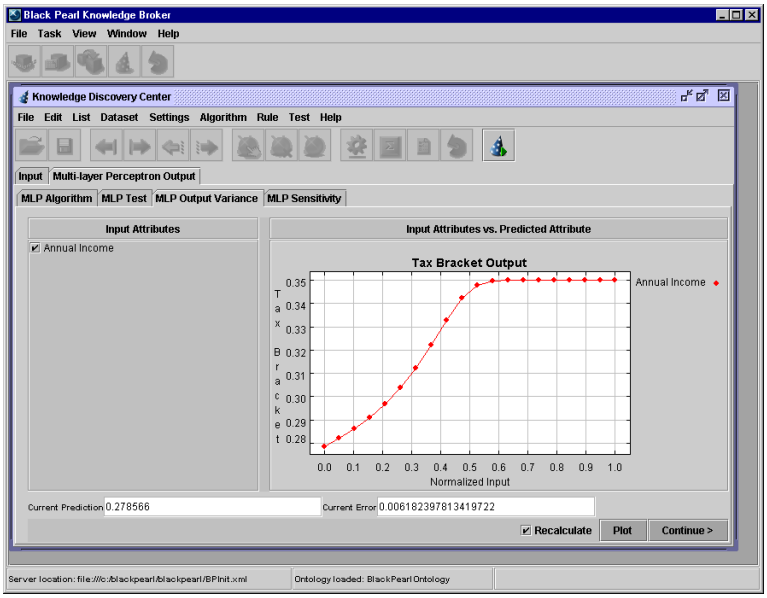
MLP Output Variance Editor

The *MLP Output Variance Editor* enables you to view the variation of the Predicted Attribute value in relation to each Input Attribute.

The MLP Output Variance Editor displays when you either:

- Click Continue in the MLP Test Editor.
- Click the MLP Output Variance tab in the Knowledge Discovery Center (Multi-layer Perceptron Output).

Figure A-26. MLP Output Variance Editor



The MLP Output Variance Editor consists of the following:

1. *Input Attributes*—defines Input Attributes will be plotted in the Risk Preference Output graph.
2. *Risk Preference Output graph*—displays the error relationship between the Input and Predictive Attributes.
3. *Discretized Predicted Attribute*—displays the distinct values for the Predicted Attribute. This displays only if the Predicted Attribute is Discrete.
4. *Current Prediction*—displays the current predicted value.
5. *Current Error field*, which enables you to view the current error value.
6. *Recalculate*—enables the recalculation of the plot. This is useful if there is a change in the Input Attributes in the MLP Test tab.
7. *Plot*—updates any the plot to reflect any changes.
8. *Continue*—accesses the MLP Sensitivity Editor and temporarily stores your selections in the Knowledge Discovery Center.

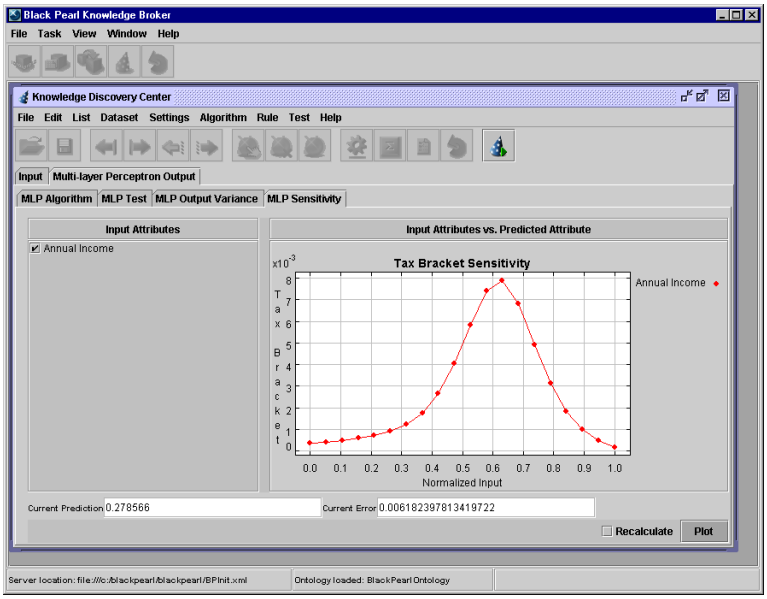
MLP Sensitivity Editor

The *MLP Sensitivity Editor* enables you to view the sensitivity of the Predicted Attribute value in relation to each Input Attribute. Since the MLP is a highly non-linear model, this variation is about the point defined in the MLP Test Editor. Each Input Attribute is varied, one at a time.

The MLP Sensitivity Editor displays when you either:

- Click Continue in the MLP Output Variance Editor.
- Click the MLP Sensitivity tab in the Knowledge Discovery Center in the Knowledge Discovery Center (Multi-layer Perceptron Output).

Figure A-27. MLP Sensitivity Editor



The MLP Sensitivity Editor consists of the following:

1. *Input Attributes*—defines which Input Attributes will be plotted in the Risk Preference Sensitivity graph.
2. *Risk Preference Sensitivity graph*—displays the sensitivity relationship between the Input and Predictive Attributes.
3. *Discretized Predicted Attribute*—displays the distinct values for the Predicted Attribute. This displays only if the Predicted Attribute is a Discrete value.
4. *Current Prediction*—displays the current predicted value.
5. *Current Error*—displays the current error value.
6. *Recalculate*—enables the re-calculation of the plot. This is useful if you made any changes to the Input Attributes in the MLP Test Editor.
7. *Plot*—refresh the plot to reflect any changes.

Appendix

B

Error Messages

This appendix provides you with information about each of the Black Pearl Knowledge Broker error messages which display under certain conditions. Each error message is identified with an Error Number, and errors are grouped according to User Interface.

Launcher Messages (1 to 999)

Error Number: 1

Message: There was an error while saving the Ontology.

Meaning: Data storage limit, running application off the CD, internal error

Action: Determine if you are running the application off the CD or the machine has reached a data storage limit. If neither of these conditions is true, contact Customer Support.

Error Number: 2

Message: The `help.jar` file is missing.

Meaning: Online Help file moved, deleted, or otherwise missing.

Action: Determine if the file is in the `blackpearl/lib` directory. If not, re-install the application. You must first move the following files to a temporary directory to avoid loss of your knowledge base: `blackpearl/BPinit.XML`, `ontology/BPOntology`, `blackpearl/ontology/BPRuleBase`, `kd`, and `Advisor.XML`. After re-installing the application, move these files back to their proper directories.

Error Number: 3 (Obsolete)

Error Number: 4 (Obsolete)

Error Number: 5

Message: There was a problem selecting the internal frame.

Meaning: Low memory or faulty installation.

Action: Determine if the problem is low memory. If not, re-install the application. You must first move the following files to a temporary directory to avoid loss of your knowledge base: `blackpearl/BPinit.XML`, `ontology/BPOntology`, `blackpearl/ontology/BPRuleBase`, `kd`, and `Advisor.XML`. After re-installing the application, move these files back to their proper directories.

Utility Messages (1000 to 1999)

Error Number: 1001

Message: There was an error retrieving Concept `<instanceName>`.

Meaning: Unspecified.

Action: Contact Customer Support.

Error Number: 1002 (Obsolete)

Data Concept Editor Messages (2001 to 2999)

Error Number: 2001

Message: You cannot add a Property to the Root Concept.

Meaning: Low memory, incomplete installation or unspecified

Action: Determine if the problem is low memory. If not, re-install the application. You must first move the following files to a temporary directory to avoid loss of your knowledge base: `blackpearl/BPinit.XML`, `ontology/BPOntology`, `blackpearl/ontology/BPRuleBase`, `kd`, and `Advisor.XML`. After re-installing the application, move these files back to their proper directories. If the problem persists after the re-install, contact Customer Support.

Error Number: 2002

Message: There was a problem adding the Concept.

Meaning: Low memory or unspecified problem.

Action: Determine if the problem is low memory. If not, contact Customer Support.

Error Number: 2003

Message: There was a problem adding the Property.

Meaning: Unspecified problem.

Action: Contact Customer Support

Error Number: 2004

Message: This Concept name already exists.

Meaning: Each Data Concepts must have a unique name.

Action: Type a new for this Data Concept.

Error Number: 2005

Message: A new Concept cannot be added until a Datasource is connected. Please add a Connection.

Meaning: The application requires a connection to a physical datasource before you can define a Data Concept.

Action: Complete the steps in *Adding a Datasource Connection on page 74*.

Error Number: 2006

Message: You must select a Concept or Property.

Meaning: No Data Concept or Data Concept Property selected.

Action: Select a Data Concept or Data Concept Property.

Error Number: 2007

Message: A Property with that name already exists. Please use another name.

Meaning: Each Data Concept Property must have a unique name.

Action: Type a new for this Data Concept Property.

Error Number: 2008

Message: Property not found.

Meaning: The Data Concept Property that you are trying to rename cannot be found by the Black Pearl Knowledge Broker.

Action: Contact Customer Support.

Error Number: 2009

Message: The item's container was not found.

Meaning: The Black Pearl Knowledge Broker cannot find the specified item. It may be missing or renamed.

Action: Contact Customer Support.

Error Number: 2010

Message: There was a problem deleting the Concept: `<conceptName>`.

Meaning: The Black Pearl Knowledge Broker cannot find the specified item. It may be missing or renamed.

Action: Contact Customer Support.

Error Number: 2011

Message: There was a problem deleting the Property: `<propertyName>`.

Meaning: The Black Pearl Knowledge Broker cannot find the specified item. It may be missing or renamed.

Action: Contact Customer Support

Error Number: 2012

Message: There was a problem deleting the container of the Concept: `<conceptName>`.

Meaning: The Black Pearl Knowledge Broker cannot find the specified item. It may be missing or renamed.

Action: Contact Customer Support.

Error Number: 2013

Message: Reserved

Meaning: Reserved

Action: Reserved

Error Number: 2014

Message: There was a problem adding the Connection. A Storage exception was thrown.

Meaning: Unspecified Storage exception.

Action: Determine if the datasource is running. If yes and the problem persists, Contact Customer Support.

Error Number: 2015

Message: There was a problem adding the Connection. A Space exception was thrown.

Meaning: Unspecified Space exception.

Action: Contact Customer Support.

Error Number: 2016

Message: There was a problem adding the Connection.

Meaning: Unspecified problem.

Action: Contact Customer Support.

Error Number: 2017

Message: There was a problem deleting the Connection.

Meaning: Unspecified problem.

Action: Contact Customer Support.

Error Number: 2018

Message: Please select an item in both trees before mapping.

Meaning: Both a Data Item and a Concept Item must be selected before this action can be completed.

Action: Select a Data Item and a Concept Item.

Error Number: 2019

Message: There was a problem with the Map command.

Meaning: Unspecified problem.

Action: Contact Customer Support.

Error Number: 2020

Message: Please select an item from the Conceptual view.

Meaning: A Data Concept or Data Concept Property must be selected before this action can be completed.

Action: Select a Data Concept or Data Concept Property.

Error Number: 2021

Message: Please select an item from the Physical view.

Meaning: A Datasource, Datasource Component, or Datasource Property must be selected before this action can be completed.

Action: Select a Datasource, Datasource Component, or Datasource Property.

Error Number: 2022

Message: The UnMap command was unsuccessful.

Meaning: Unspecified problem.

Action: Contact Customer Support.

Error Number: 2023

Message: The Concept name cannot be empty.

Meaning: You clicked OK before typing a name in the appropriate field.

Action: Type a name in the appropriate field.

Error Number: 2024

Message: You cannot add a Property to the Root Concept.

Meaning: Datasource Properties are automatically populated from your underlying Datasource table columns or pages.

Action: Add a table column or page to the underlying Datasource before proceeding.

Business Concept Editor Messages (3001 to 3999)

Error Number: 3001

Message: Concept `<conceptName>` was not found.

Meaning: The Business Concept that you are trying to locate is not saved within the knowledge base.

Action: Manually search for the item by scrolling through the Concept list.

Error Number: 3002

Message: The found Concept cannot be selected.

Meaning: Unspecified problem.

Action: Contact Customer Support.

Error Number: 3003

Message: A Concept name must have at least one character.

Meaning: You clicked OK before typing a name in the appropriate field.

Action: Type a name in the appropriate field.

Error Number: 3004

Message: A Concept with that name already exists. Please use another name.

Meaning: Each Business Concept must have a unique name.

Action: Type a different name in the appropriate field.

Error Number: 3005

Message: The Parent Concept is no longer available.

Meaning: The Black Pearl Knowledge Broker cannot locate the Parent Concept. It may be renamed or missing.

Action: Contact Customer Support.

Error Number: 3006

Message: There was a problem adding the Concept.

Meaning: Unspecified problem.

Action: Contact Customer Support.

Error Number: 3007

Message: There was a problem deleting the Concept.

Meaning: Unspecified problem.

Action: Contact Customer Support.

Error Number: 3008

Message: Unable to set the current Concept.

Meaning: Unspecified problem.

Action: Contact Customer Support.

Error Number: 3009

Message: The selected rows cannot be grouped. This would create improper nesting.

Meaning:

Action: Select other filter rows for grouping.

Error Number: 3010

Message: Multiple rows cannot be deleted.

Meaning: The application can only delete one filter row at a time.

Action: Delete one filter row at a time.

Error Number: 3011

Message: Grouped rows cannot be deleted.

Meaning: The application cannot delete grouped filter rows.

Action: Ungroup the filter rows, and delete each filter row one at a time.

Rule Editor Messages (5001 to 5999)

Error Number: 5001

Message: There was a problem retrieving the Rules.

Meaning: Unspecified problem.

Action: Contact Customer Support.

Error Number: 5002

Message: The Subject must be defined.

Action: Every rule is in the form: `if <SUBJECT> then <VERB> <DIRECT OBJECT>`.

Action: Select an item from the Concept list.

Error Number: 5003

Message: The Verb must be refined.

Meaning: Every rule is in the form: `if <SUBJECT> then <VERB> <DIRECT OBJECT>.`

Action: Select an item from the Concept list.

Error Number: 5004

Message: The Indirect Object must be defined.

Meaning: Every rule is in the form: `if <SUBJECT> then <VERB> <DIRECT OBJECT>.`

Action: Select an item from the Concept list.

Error Number: 5005

Message: The Direct Object must be defined.

Meaning: Every rule is in the form: `if <SUBJECT> then <VERB> <DIRECT OBJECT>.`

Action: Select an item from the Concept list.

Error Number: 5006

Message: There was a problem retrieving the Rule Concept.

Meaning: Unspecified problem.

Action: Contact Customer Support.

Error Number: 5007

Message: There are no rules to print.

Meaning: No rules are defined.

Action: Define a rule before attempting to print a report.

Knowledge Discovery Center Messages (6001 to 6999)

Error Number: 6001

Message: A Concept must be selected.

Meaning: The action requested requires a Concept.

Action: Select an item from the Concept list.

Error Number: 6002 (Reserved)

Message: Reserved

Error Number: 6003

Message: Value `<x>` is a duplicate for `<name>` and `<oldValue>`.

Meaning: Each value must be unique.

Action: Type a different value in the appropriate field.

Error Number: 6004

Message: The value `<x>` is not a number.

Meaning: The action requested requires a number.

Action: Type a number value in the appropriate field.

Error Number: 6005

Message: The Ontology cannot be opened.

Meaning: Unspecified problem.

Action: Contact Customer Support.

Error Number: 6006

Message: The classification cannot be retrieved.

Meaning: Unspecified problem.

Action: Contact Customer Support.

Error Number: 6007

Message: The Property cannot be retrieved.

Meaning: Unspecified problem.

Action: Contact Customer Support.

Error Number: 6008

Message: The Property names cannot be retrieved.

Meaning: Unspecified problem.

Action: Contact Customer Support.

Error Number: 6009

Message: Cannot open the file.

Meaning: Unspecified problem.

Action: Contact Customer Support.

Error Number: 6010

Message: Cannot save the file.

Meaning: Unspecified problem.

Action: Contact Customer Support.

Error Number: 6011

Message: The File Chooser cannot be opened.

Meaning: Low memory or an unspecified problem.

Action: Determine if memory is low. If not, contact Customer Support.

Error Number: 6012

Message: The extension on the file is invalid.

Meaning: Unspecified problem.

Action: Contact Customer Support.

Error Number: 6013

Message: This is an invalid file. It is corrupt.

Meaning: Renames can corrupt a file.

Action: Determine if you have a valid dataset and correct extension. If you do and the problem persists, contact Customer Support.

Error Number: 6014

Message: The file cannot be read. It may be corrupt.

Meaning: Renames can corrupt a file.

Action: Determine if you have a valid dataset and correct extension. If you do and the problem persists, contact Customer Support.

Error Number: 6015

Message: The file cannot be selected.

Meaning: Unspecified problem.

Action: Contact Customer Support.

Error Number: 6016

Message: The file cannot be opened for saving.

Meaning: The file is set for read-only.

Action: De-select the read-only option.

Error Number: 6017

Message: There are no Available Attributes selected.

Meaning: The action requested requires at least one Available Attribute.

Action: Select Available Attributes in the Available Attributes Editor.

Error Number: 6018

Message: It is not possible to write to the file `<fileName>`.

Meaning: Unspecified problem.

Action: Contact Customer Support.

Error Number: 6019

Message: There is no Predicted Attribute selected.

Meaning: The action requested requires a Predicted Attribute.

Action: Select an item in the Input Attributes Editor.

Error Number: 6020

Message: There was a problem retrieving Instances from the Ontology.

Meaning: Unspecified problem.

Action: Contact Customer Support.

Error Number: 6021

Message: The Attribute `<attName>` cannot be found.

Meaning: Unspecified problem.

Action: Contact Customer Support.

Error Number: 6022

Message: None of the Attributes are discrete.

Meaning: Unspecified problem.

Action: Contact Customer Support.

Error Number: 6023

Message: There is no Decision Tree to prune.

Meaning: You have not created a Decision Tree.

Action: Generate a Decision Tree (see *Generating and Updating a Decision Tree on page 161*).

Error Number: 6024

Message: There are no examples to prune. Please select a Dataset.

Meaning: There is no dataset defined.

Action: Define a dataset in the Dataset Creation Editor.

Error Number: 6025

Message: It is not possible to prune the Decision Tree.

Meaning: Unspecified problem.

Action: Contact Customer Support.

Error Number: 6026

Message: There is no Rule to save.

Meaning: The action requested requires at least one rule.

Action: Generate a Decision Tree, which should create rules (see *Generating and Updating a Decision Tree on page 161*).

Error Number: 6027

Message: There was a problem retrieving the Rulebase.

Meaning: Unspecified problem.

Action: Contact Customer Support.

Error Number: 6028

Message: There was a problem saving the Rule.

Meaning: Unspecified problem.

Action: Contact Customer Support.

Error Number: 6029

Message: There is no Rule collection in the Rulebase.

Meaning: The action requested requires at least one rule.

Action: Generate a Decision Tree, which should create rules (see *Generating and Updating a Decision Tree on page 161*).

Error Number: 6030

Message: There are no Rules to display.

Meaning: The action requested requires at least one rule.

Action: Generate a Decision Tree, which should create rules (see *Generating and Updating a Decision Tree on page 161*).

Error Number: 6031

Message: There was a problem retrieving the Rule Concept.

Meaning: Unspecified problem.

Action: Contact Customer Support.

Error Number: 6032

Message: There is no Decision Tree.

Meaning: The action requested requires a Decision Tree.

Action: Generate a Decision Tree (see *Generating and Updating a Decision Tree on page 161*).

Error Number: 6033

Message: There is nothing to Predict.

Meaning: The action requested requires either a Decision Tree or MLP.

Action: Generate a Decision Tree (*Generating and Updating a Decision Tree on page 161*) or define an MLP (*Creating and Training a Multi-Layer Perceptron on page 167*).

Error Number: 6034

Message: There are no Input Attributes available.

Meaning: The action requested requires Input Attributes.

Action: Select Input Attributes in the Input Attributes Editor.

Error Number: 6035

Message: It is not possible to obtain discrete values for `<attName>`.

Meaning: The attribute <attName> does not have any valid values.

Action: Remove this attribute from the predictive model.

Error Number: 6036

Message: The Prediction is always the same, regardless of the input.

Meaning: The Decision Tree has a single default rule that does not use any of the Input Attributes.

Action: Select other Input Attributes.

Error Number: 6037

Message: There was a problem creating the kd data directory.

Meaning: Unspecified problem.

Action: Contact Customer Support.

Error Number: 6038

Message: There are no Available Attributes selected.

Meaning: The action requested requires Available Attributes.

Action: Select items in the Available Attributes Editor.

Error Number: 6039

Message: There are no Input Attributes selected.

Meaning: The action requested requires Input Attributes.

Action: Select items in the Input Attributes Editor.

Error Number: 6040

Message: There is no data to create the Decision Tree.

Meaning: Your datasource is empty.

Action: Contact your Database Administrator.

Error Number: 6041

Message: There are no examples to create a Network.

Meaning: Your datasource is empty.

Action: Contact your Database Administrator.

Error Number: 6042

Message: There is a problem enabling the tab.

Meaning: Unspecified problem.

Action: Contact Customer Support.

Error Number: 6043

Message: There is no Network to train.

Meaning: Unspecified problem.

Action: Contact Customer Support.

Error Number: 6044

Message: The given number of iterations is invalid.

Meaning: There iteration value is too low or too high.

Action: Type another value in the appropriate field.

Error Number: 6045

Message: There are no examples to train the Network.

Meaning: Unspecified problem.

Action: Contact Customer Support.

Error Number: 6046

Message: There are no Instances in the training set.

Meaning: Unspecified problem.

Action: Contact Customer Support.

Error Number: 6047

Message: The training set cannot be set.

Meaning: Unspecified problem.

Action: Contact Customer Support.

Error Number: 6048

Message: There was a problem during training.

Meaning: Unspecified problem.

Action: Contact Customer Support.

Error Number: 6049

Message: There was a problem displaying the MLP.

Meaning: Unspecified problem.

Action: Contact Customer Support.

Error Number: 6050

Message: There is no MLP to save.

Meaning: The requested action requires an MLP.

Action: Define an MLP (see *Creating and Training a Multi-Layer Perceptron on page 167*).

Error Number: 6051

Message: There was a problem while saving the MLP. The current Concept is null.

Meaning: Unspecified problem.

Action: Contact Customer Support.

Error Number: 6052

Message: This is an invalid file. Cannot read the Predicted Attribute name.

Meaning: The file extension is invalid or the file is corrupted.

Action: Contact Customer Support.

Error Number: 6053 (Obsolete)

Message: 6054

Message: The Attribute `<attName>` is not available.

Meaning: Unspecified problem.

Action: Contact Customer Support.

Error Number: 6055

Message: No data is available for the prediction.

Meaning: Unspecified problem.

Action: Contact Customer Support.

Error Number: 6056

Message: The Attribute `<attName>` is unavailable.

Meaning: Unspecified problem.

Action: Contact Customer Support.

Error Number: 6057

Message: Value for the Attribute `<attName>` is not valid. It should be one of `<attName>`.

Meaning: Incorrect value.

Action: Type the correct value in the appropriate field.

Error Number: 6058

Message: There was a problem during the prediction.

Meaning: Unspecified problem.

Action: Contact Customer Support.

Error Number: 6059

Message: It is not possible to calculate Sensitivity.

Meaning: Unspecified problem.

Action: Contact Customer Support.

Error Number: 6060

Message: The number of hidden nodes should be between `<MinNumber>` and `<MaxNumber>`.

Meaning: Incorrect value for the hidden nodes.

Action: Type the correct value in the appropriate field.

Error Number: 6061

Message: There was a problem setting the Network topology.

Meaning: Unspecified problem.

Action: Contact Customer Support.

Scripting Messages (7001 to 7999)

Error Number: 7001

Message: It is not possible to play the Track.

Meaning: Unspecified problem.

Action: Contact Customer Support.

Error Number: 7002

Message: Please open a script file first.

Meaning: The action requested requires an open script file.

Action: Open a script file.

Error Number: 7003

Message: Knowledge Discovery Scripting cannot open the File Chooser.

Meaning: Low memory or an unspecified problem.

Action: Determine whether the memory is low. If not, contact Customer Support.

Error Number: 7004

Message: Knowledge Discovery Scripting cannot open the selected file. The file must be in `<dirName>`.

Meaning: The requested file is in an invalid directory.

Action: Move the file to the appropriate directory, as specified in the message.

Error Number: 7005

Message: The file extension is invalid. The file should end with `<extName>`.

Meaning: The requested file has an invalid file extension.

Action: Rename the file with the correct extension, as specified in the message,

Error Number: 7006

Message: Please close the Knowledge Discovery Scripting window before closing the Knowledge Discovery Center.

Meaning: The Scripting Editor must be closed before you can exit the Knowledge Discovery Center.

Action: Close the Scripting Editor.

Glossary

Action. An event that effects a change (*See also* Verb).

Agent. An intelligent, autonomous, and mobile entity capable of acting on behalf of a human, another agent, or a software program. *Intelligent* means that the agent is capable of perceiving its environment, learning, collaborating, communicating, adapting, and making decisions. *Autonomous* means that the agent acts in accordance with a set of programmed or learned functions that it attempts to optimize. *Mobile* means that the agent is capable of associating or de-associating with other agents in an attempt to optimize its actions.

Algorithm. A step-by-step procedure or set of rules written for the purpose of solving a specific problem or completing a particular task. Algorithms can be written as plain English, programming language, or mathematical equation.

Antecedent. The part of rule that precedes the action (*See also* Subject).

API. Application Program Interface. A means by which two independently-written software programs can interact with one another to perform additional functions.

Association of Ideas. The ability to unite in some manner two or more concepts, based on contiguity and/or resemblance (*See also* Contiguity, Resemblance).

Attribute. The character or property of an object.

Asynchronous. Asynchronous communication describes a process where an exchange of information proceeds without a requirement for any indication of whether transmission was successful or unsuccessful (*See also* synchronous).

Binning. Process of converting continuous variables to discrete variables (*See also* Continuous Variables, Discrete Variables).

Business Concept. A vocabulary term that is relevant to your industry or company. You define Business Concepts in the ontology's Business Concept Editor (*See also* Concept).

Client. A Client program requests data and services from a Server (*qv*), performs some processing, and presents the output to the Client user. The Knowledge Broker can be installed as a Client version that requires software connection to a Knowledge Broker Server to function.

Compositionality. The ability to infer knowledge from the meanings, patterns, and relationships of individual concepts or information, based on a system of rules.

Concept. An abstract image about the who, what, when, and where of data and data relationships. Within the Black Pearl Knowledge Broker, concepts are the vocabulary terms used for defining rules, querying disparate datasources, analyzing information patterns, recognizing implications, and making actionable recommendations.

Concept Comparison. A feature in the Rule Editor that allows you to compare two properties from two different concepts.

Concept Relation. The filtering of concepts so that a single concept may have different meanings, dependent on the context.

Condition. The antecedent portion of a rule (*See also* Rules).

Consequent. The part of the rule that receives the action (*See also* Verb).

Constraint. A range or definition for the space of acceptable solutions to a user- or agent-generated question. Black Pearl applies constraints to searches and rules. Constraints help the Black Pearl Knowledge Broker bind variables to values and limit number of combinations needed to arrive at the goal-state.

Context. The relationship between two or more data elements or concepts.

Contiguity. Concepts frequently experienced together are mentally associated together.

Continuous Variable. Any value in an interval of real numbers.

Data. A specific person, place, thing, or event object.

Data Concept. A vocabulary term that maps directly to one or more datasource objects.

Dataset. The collection of datasource Instances.

Datasource. The disparate Web, relational database, and legacy files containing data pertinent to your business.

Datasource Connection. The URLs, strings, or pathnames to your Web, relational database, and legacy files.

Data Visualization. A graphic method for presenting information.

Decision Tree. A graphical representation of the relationships between a dependent variable (output) and a set of independent variables (input). Usually in the form of a tree-shaped structure that represents a set of decisions, it examines data and determines which property contains the most information on which to base a decision. This property is used as a decision node to split the dataset into two groups based on the value of that property. At each subsequent decision node, the data is split again, resulting in a decision tree or collection of nodes.

Deduction. An instance of deriving a conclusion by reasoning about the logical consequence of known information.

Dependent Variable. The output variable or response of an action.

Direct Object. The part of the rule that receives the Verb (*See also* Action).

Discrete Variable. A finite set of values.

Dynamic. Makes use of existing data elements, concepts, contexts, information, and knowledge to create additional knowledge.

Editor. A User Interface (UI) that enables a person to communicate with the Black Pearl Knowledge Broker to select options, set parameters, and save, modify, or delete information.

Explicit Knowledge. A set of codified experience and know-how that is communicated through formal language, including data elements, concepts, contexts, and information (*See also* Tacit Knowledge).

Filter. A device specifying the parameters that the Black Pearl Knowledge Broker will use to evaluate, process, and sort data.

Independent Variable. The input variable or condition for an action.

Induction. The instance of inferring a conclusion. (*See also* Infer.)

Infer. The process of arriving at a conclusion by reasoning from evidence or premise.

Information Space. Black Pearl Knowledge Broker architecture is based on an Information Space that allows agents to work with each other by posting new information into the space. The Information Space allows multiple agents to work independently and cooperate in solving a problem.

Instance. A specific record associated with a Data or Business Concept. For example, every person identified as a customer is an Instance of the Customer Data Concept, while every person identified as a male customer is an Instance of the Male Business Concept.

Intelligent Agent. An agent is considered to be intelligent if it is capable of perceiving its environment, learning, collaborating, communicating, adapting, and making decisions.

Iterative. An iterative process occurs when the process repeats, taking its result output and feeding it back into itself to arrive at a progressively more refined or detailed result.

JDBC. A Java computer language-based technology that Knowledge Broker uses to gain access to information stored in tabular-format databases (*See also* ODBC).

Knowledge. A group of four interrelated things: the understanding of patterns of information that consistently and completely repeat themselves over time; the understanding of the implications of those patterns on future behavior; the ability to make plans and recommendations based on those implications; and the ability to act on those plans.

Knowledge Base. A collection of datasource connections, ontology concepts and concepts relations, rules, and predictive models related to a particular industry or business domain.

Knowledge Discovery. The Black Pearl Knowledge Broker process that analyzes historical data to determine patterns, relationships, and trends that can be used to predict future values and real-time recommendations.

Knowledge Discovery Method. An algorithm designed to accomplish a Discovery Task (*See also* Knowledge Discovery Task).

Knowledge Discovery Task. A request for a specific new knowledge.

Knowledge Discovery System. A software-based system that either independently performs or interactively supports a person in completing a Discovery Task.

Learning. The Black Pearl Knowledge Broker has an ability to adapt or learn from experience. The learning agents adapt to a customers likes and dislikes, which agents to cooperate with and trust and which ones to avoid. Learning is based on the Black Pearl Knowledge Broker ability to recognize situations it has seen before and improve its performance based on prior experience.

Map. The link between a Datasource Connection and a Data Concept.

Mean. The representative average for the entire dataset.

Metadata. The description of the location, format, and meaning of data stored in your disparate datasources.

Model. An abstract conception or representation of a portion of reality.

Multi-layer Perceptron (MLP). A predictive algorithm.

Node. The decision point or end of a link in a neural network or decision tree.

Object. An individual person, place, thing, or event either having or being apprehended as having a distinct and demonstrable existence in space and/or time.

ODBC. A Microsoft Windows-centric technology that Knowledge Broker uses to gain access to information stored in tabular-format databases (*See also* JDBC).

Ontology. A collection of concepts and contexts used to provide a common vocabulary for defining rules, querying disparate datasources, analyzing information patterns, recognizing implications, and making actionable recommendations.

Operation. An action specified in a rule.

Optimizer. An interface to optimization algorithms that are responsible for dictating the best allocation of assets.

Pattern. An orderly arrangement or repetition of data objects.

Persistence. An object that stays in memory and can fulfill multiple requests. The Black Pearl Knowledge Broker supports the persistence of concept profiles and relationships.

Planning. Taking account of constraints to select the best action or operation to perform for a given situation or set of circumstances. The Black Pearl Knowledge Broker implements a planning agent which makes decisions along the way to its goal state and is aware when the goal state is reached.

Policy. A business practice or rule.

Population. Data and Business Concepts used as the antecedent portion in a rule.

Predictive Model. A data-mining process whereby a mathematical function is used to analyze your data to discover metadata (*qv*) and make predictions about actions, behavior, or requirements. It is a method for "learning by example."

Profile. Information describing a customer's likes and dislikes.

Property. A characteristic of an object.

Pruning. The process of eliminating either lower-level splits or entire sub-trees in a decision tree.

Real-time Recommendation. Advice that recognizes the implications of patterns on future behavior.

Reasoning. A set of rules and facts; a working memory of derived data and facts and an inference engine containing reasoning logic used to process rules and data. The Black Pearl Knowledge Broker reasoning engine supports both forward- and backward-chaining to derive and test facts from data.

Recommendation. A suggestion for products, services, or actions.

Resemblance. When two concepts are considered to be similar, whatever properties associated with one concept are automatically associated with the other concept.

Root Concept. The overall name for an ontology.

Rule. Describes how you expect a person or thing to behave in a particular situation. Rules are the means for "teaching" the computer how to respond to situations relevant to your business requirements, and are the basis for generating actionable recommendations. In Knowledge Broker, rules take a familiar "if-then" format.

Rule Base. A collection of rules.

Sampling. The process of creating a sub-set of representative data from all available data.

Scripting. An automated process for a series of actions.

Server. A Server program provides data and services to a Client (*qv*). The Knowledge Broker can be installed as a Server version that supports a number of Knowledge Broker Clients.

Standalone. Knowledge Broker can be installed as a Standalone version that combines both Client and Server within one program and provides full functionality.

Subject. The part of rule that precedes the Verb (*See also* Antecedent).

Synchronous. Asynchronous communication describes a process where an exchange of information continues without a requirement for any indication of whether an individual transmission was successful or unsuccessful. (*See also* Asynchronous).

Tacit Knowledge. A set of personal experiences, know-how, and mental images that is rarely (if ever) communicated with others (*See also* Explicit Knowledge).

Task Editor. The different functional components of the Knowledge Broker's graphical user interface that encapsulate discrete sets of functionality. Also known as Task Center.

Value. The content of a field or variable. It can refer to alphabetical or numerical content.

Variance. A measure of the spread within a dataset.

Variable. A value that is a measurement of something real and that can be used in a mathematical model. Changes in the value of the variable are a representation or proxy for changes or developments in the real object

Verb. An event that effects a change (*See also* Action).

Index

D

L

M

N

O

P

R

S

T

U

V

W