BLACK PEARL

Knowledge Broker Training

Units

Unit 1: Training Overview

Unit 2: Knowledge Engineering Approach for Assessment and Planning

Unit 3: Approach to Knowledge Engineering

Unit 4: Knowledge Broker Technical Implementation

Appendices

Appendix A: ABC Brokerage Case Study

- A1: ABC Brokerage Case Study Proposal and Position
- A2: ABC Brokerage Case Study Assessment and Planning
- A3: ABC Brokerage Case Study Design Solution
- A4: ABC Brokerage Case Study Build Solution

Appendix B: Installing Applications

Appendix C: M2 Overview

Appendix D: Feedback

BLACK PEARL

Unit 1: Knowledge Broker Overview

Black Pearl Software, Inc.
November 2, 2000

Note to participant

The names, characters, places, and incidents in this workbook are fictitious. Any resemblance to actual businesses, products, persons, events or locales is entirely coincidental.

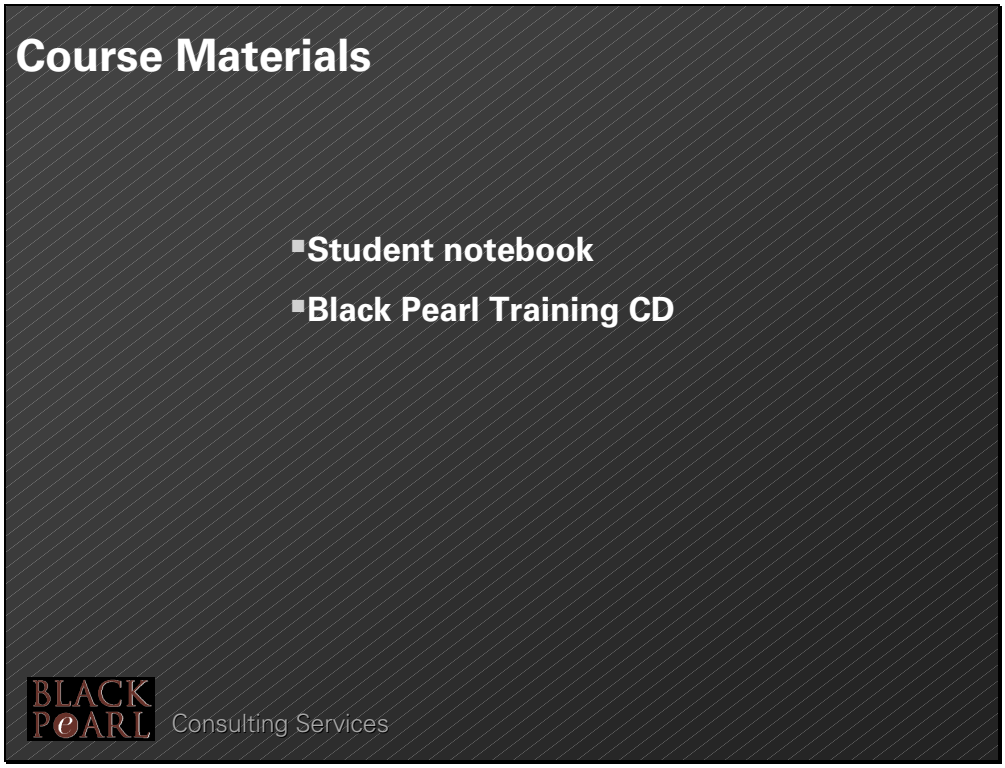
Your organization has signed a contract and non-disclosure entitling your participation in this training program. This entitles you to participate in this training and to use and retain these materials. Neither this program nor any part thereof may be reused or reproduced in any form or medium now known or hereafter invented or created without the express written permission of Black Pearl, Inc.

000X

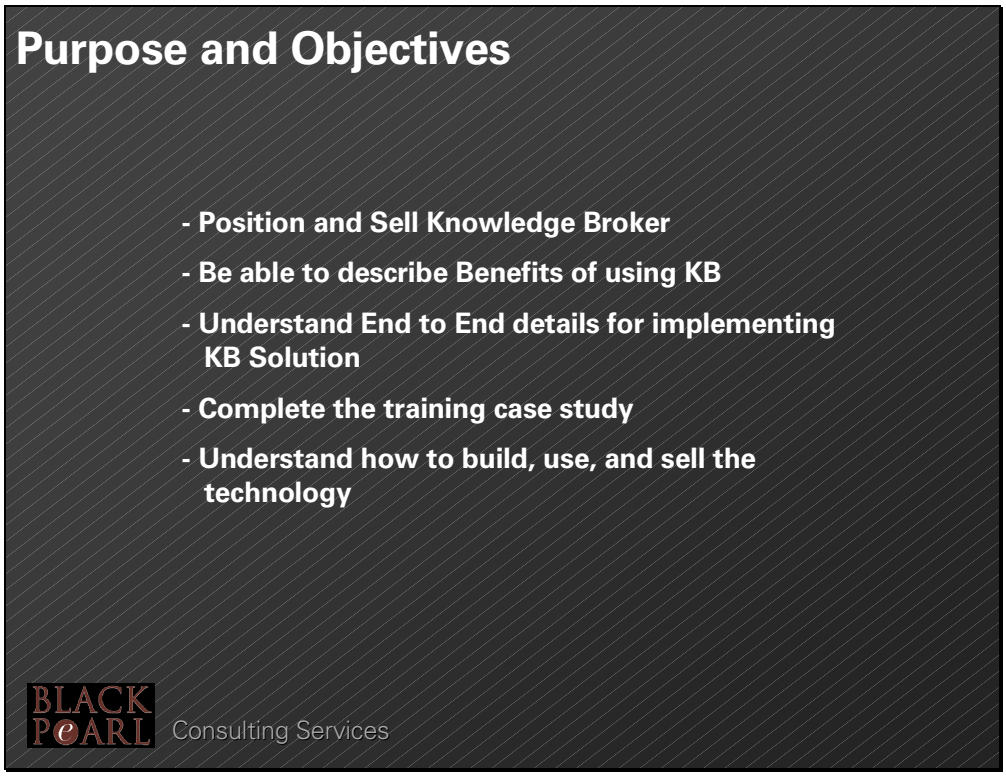
Contents

1. Knowledge Broker Fundamentals

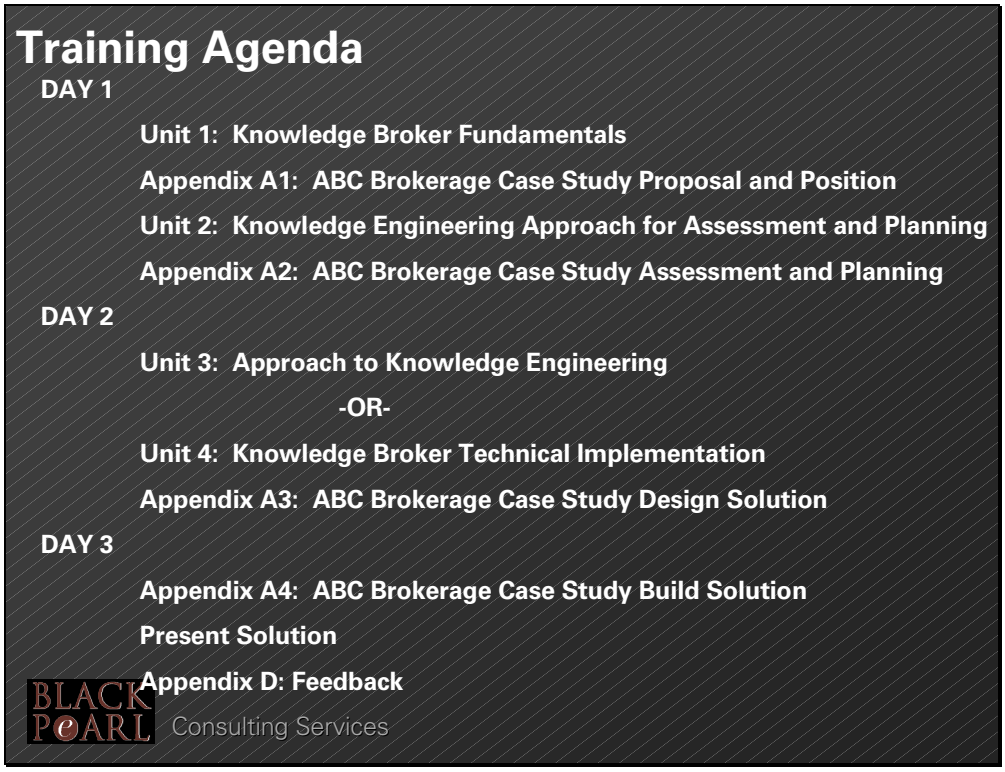
2. Course Materials



3. Purpose and Objectives



4. Training Agenda

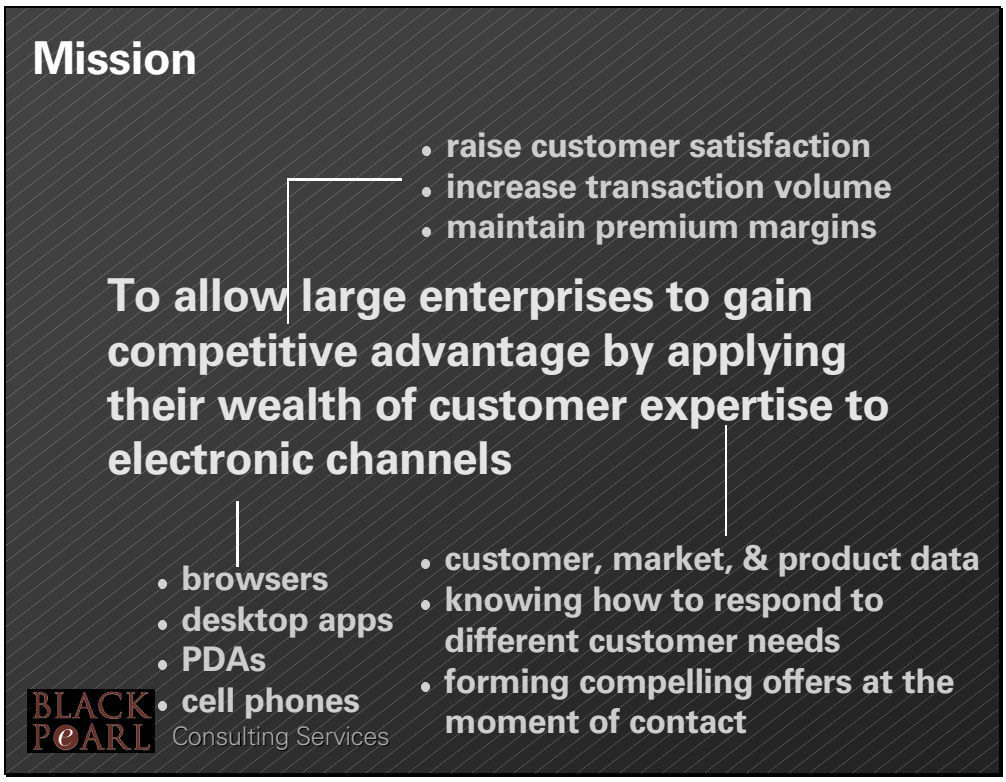


5. Introduction

6. The Market


The Market
A wide gap has emerged between the interests of online buyers and sellers
In traditional commerce, this gap is filled by sellers who are sensitive to buyer needs
Black Pearl's Knowledge Broker closes this gap by pursuing the goals of both buyers and sellers within the fast-changing context of e-commerce
Going forward, Black Pearl's technologies can be used to form relationships and negotiate terms between communities of buyers and sellers
BLACK PEARL
Consulting Services

7. Mission



CUSTOMER EXPERTISE:

Data about who the customer is, likes & dislikes, goals, implicit interests. What competition is doing. What products are available.

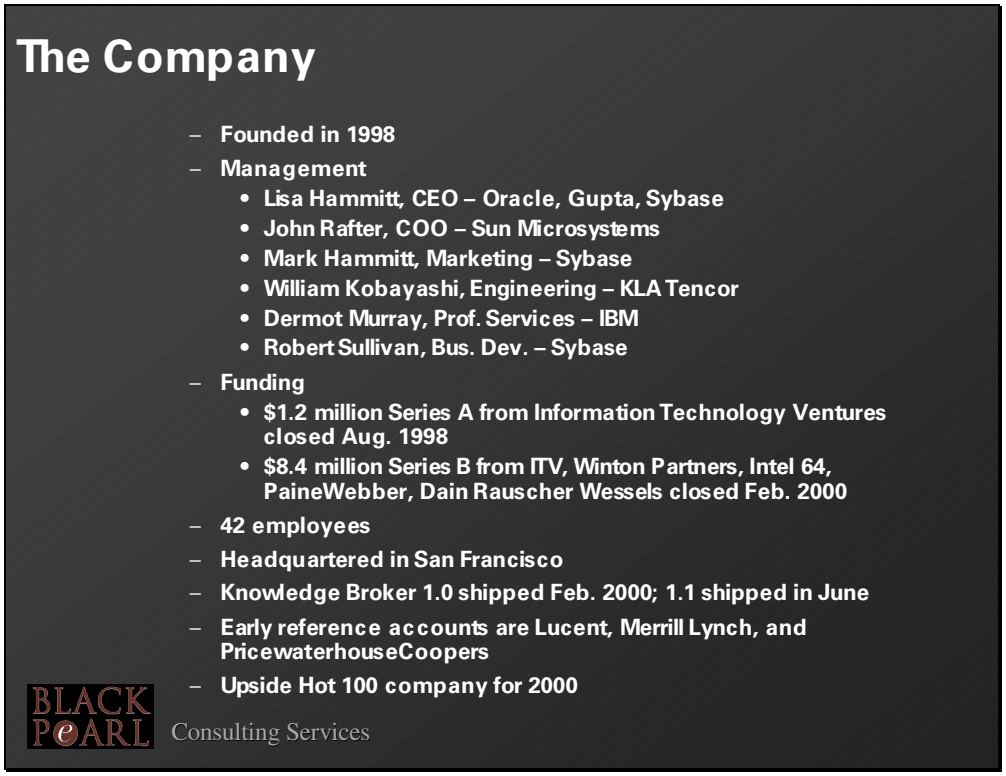
ELECTRONIC CHANNELS: Supporting 360° view

COMPETITIVE ADVANTAGE: It protects brand equity.

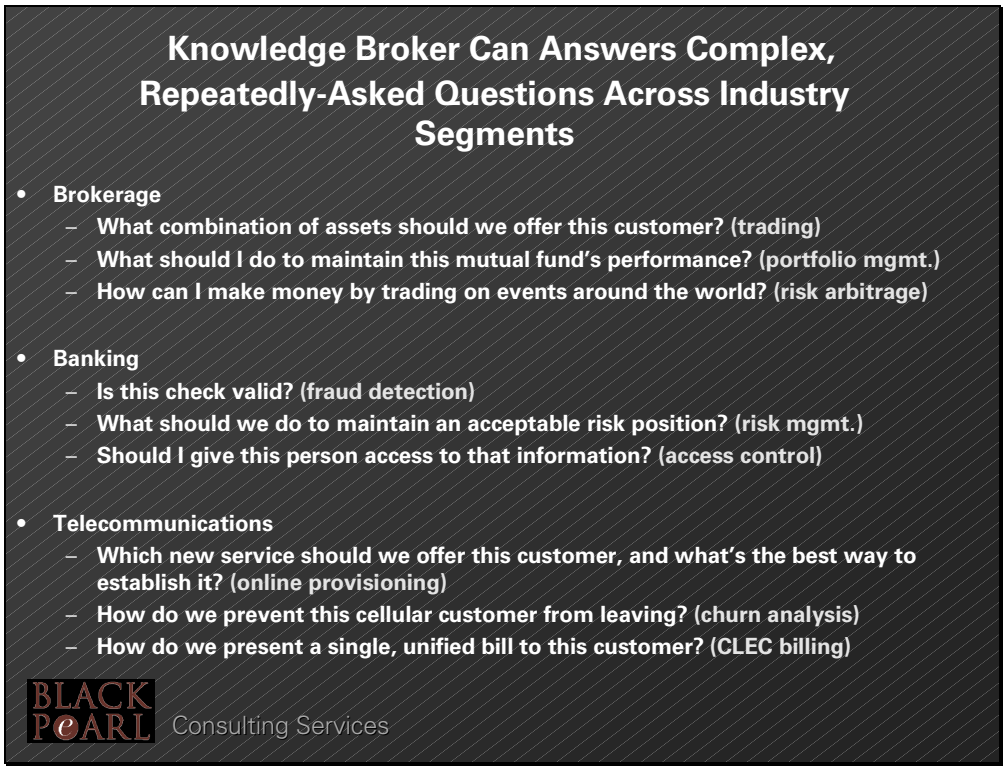
Message resonates with CEOs frustrated by the genericism of electronic channels. IT *should* make differentiation easier. *Instead,* information technologies have been a barrier to differentiation, reducing every electronic forum to one of generic serf-service.

TRANSITION: We're focusing in particular on areas requiring relatively high expertise... <BUILD>

8. The Company



9. Knowledge Broker Can Span Across Industry Segments



10. Knowledge Broker Can Span Across Industry Segments

Knowledge Broker Can Answers Complex, Repeatedly-Asked Questions Across Industry Segments

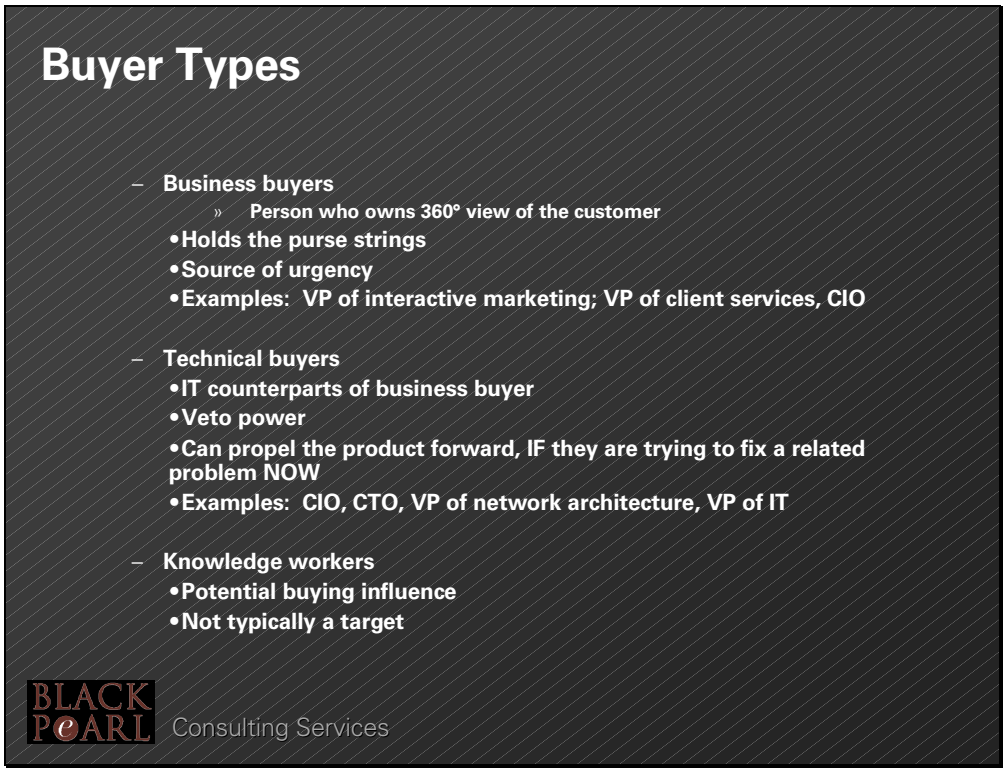
Health Care

Which doctor should this person be directed to? (triage)
How can I most efficiently treat this patient? (disease mgmt.)
What benefits is this person eligible for and at what price? (benefits admin.)

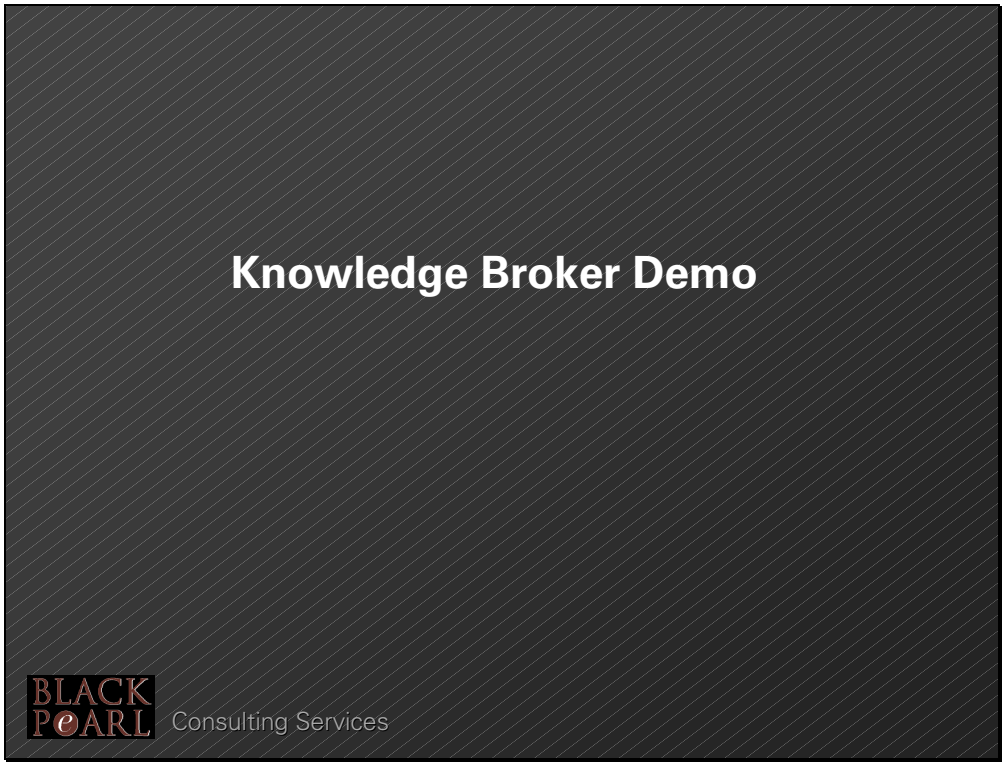
- **Manufacturing**
  - **What other goods or services should we offer this customer? (industry portals)**
  - **What goods should we produce, where should we make them and where should we ship them? (supply-chain mgmt.)**
- **Cross-Industry**
  - **What is a likely cause of this person's computer difficulty? (systems & network mgmt.)**
  - **What special offers or discounts are available to this customer? (SFA)**

BLACK PEARL Consulting Services

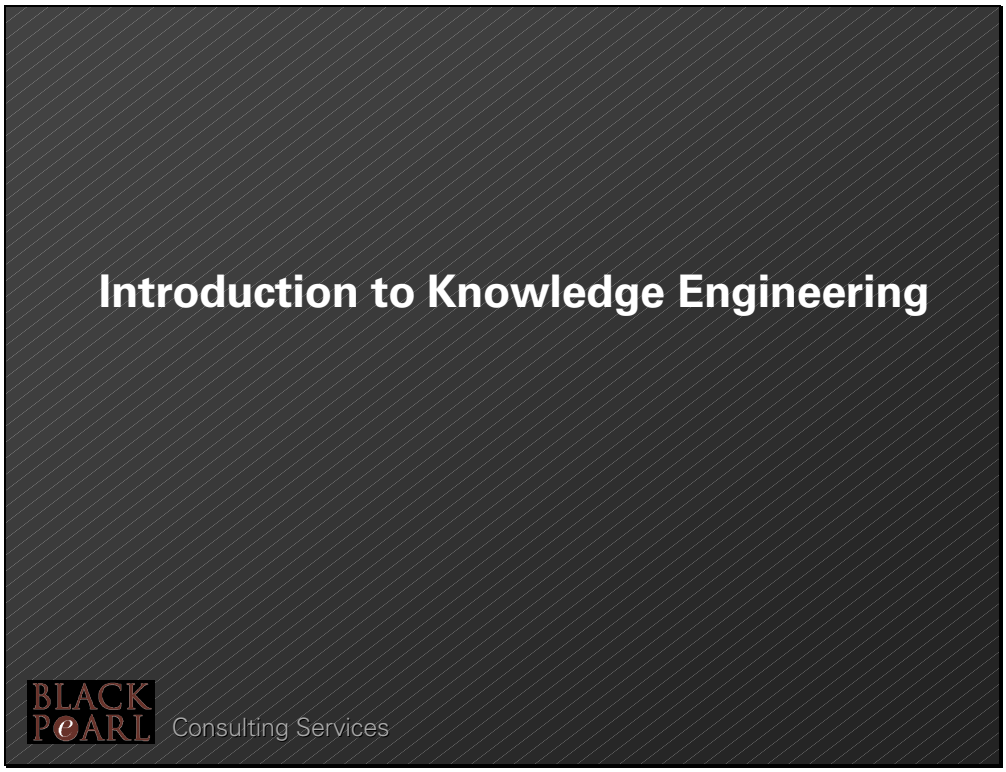
11. Buyer Types



12. Knowledge Broker Demo

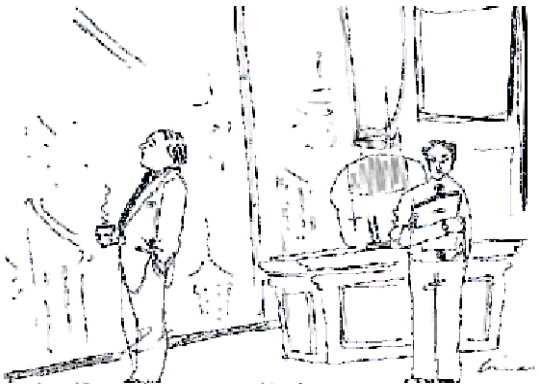


13. Introduction to Knowledge Engineering



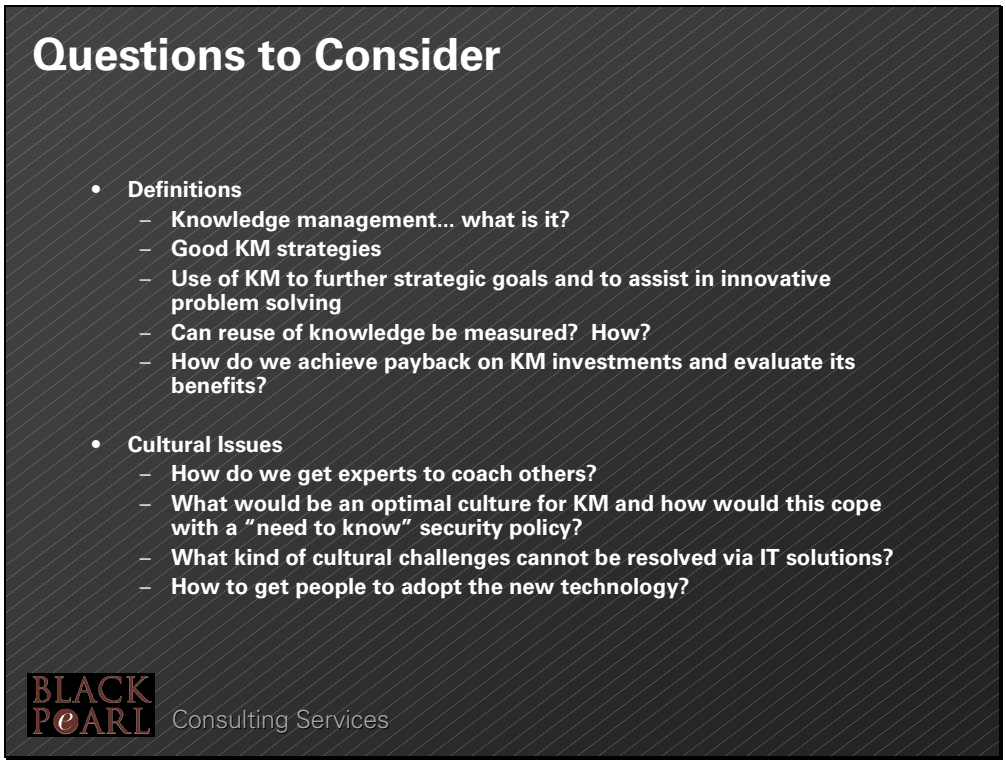
14. This is Hard...

This is Hard ...

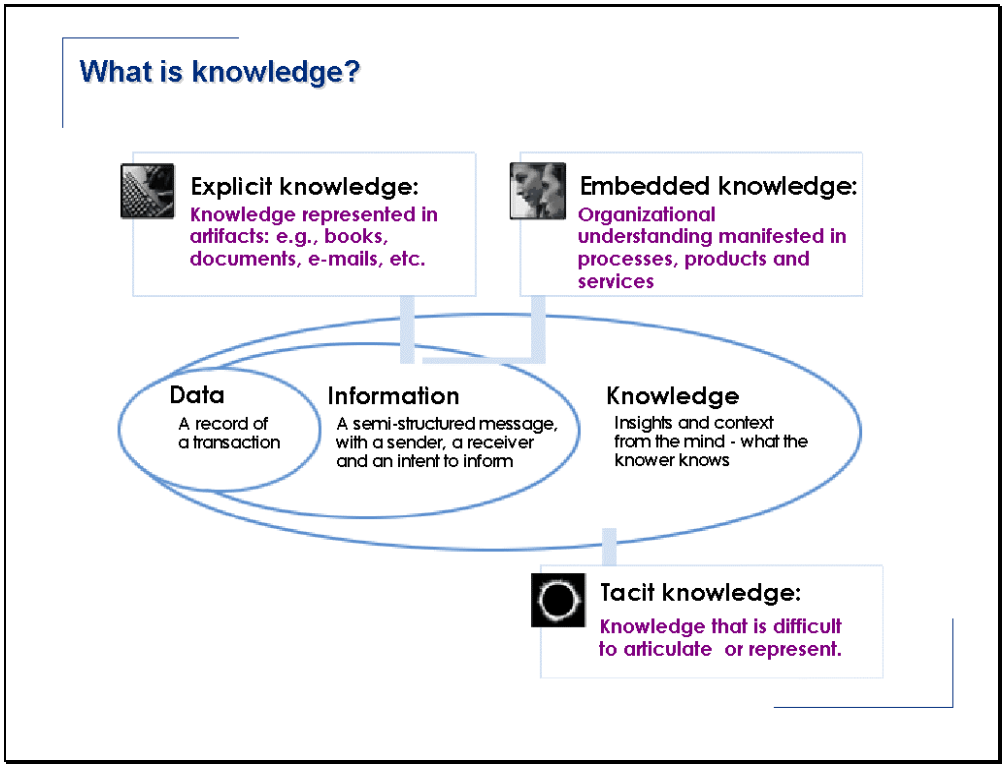


"Fenby, find our Intellectual Capital and transfer some of it to the Minsk office."

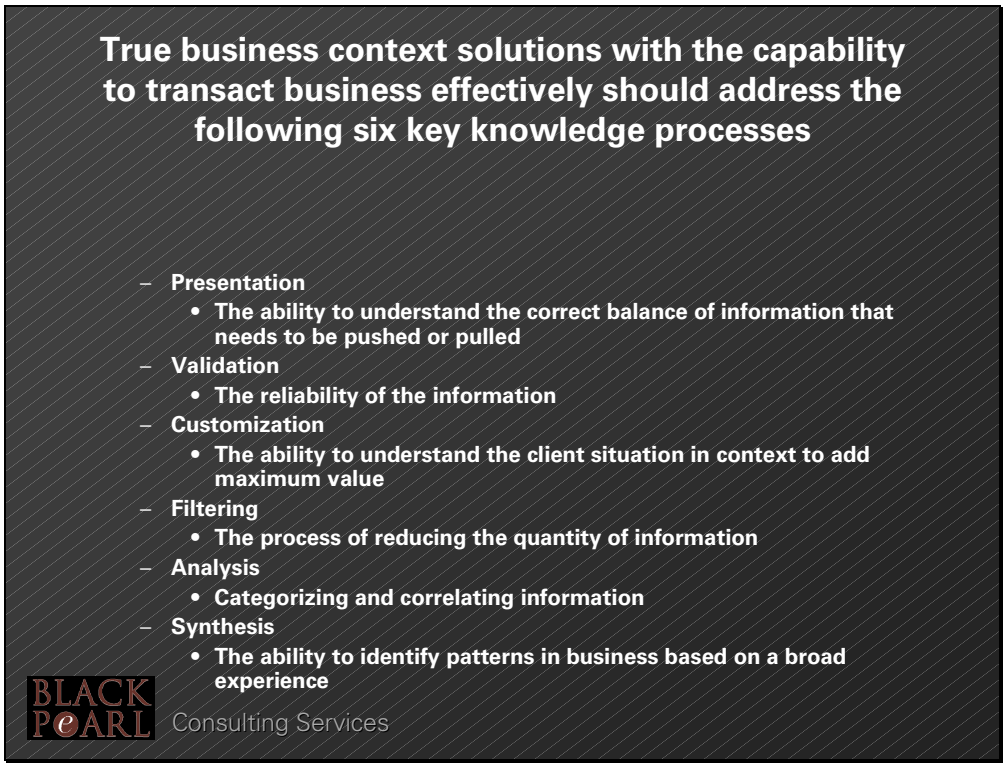
15. Questions to Consider



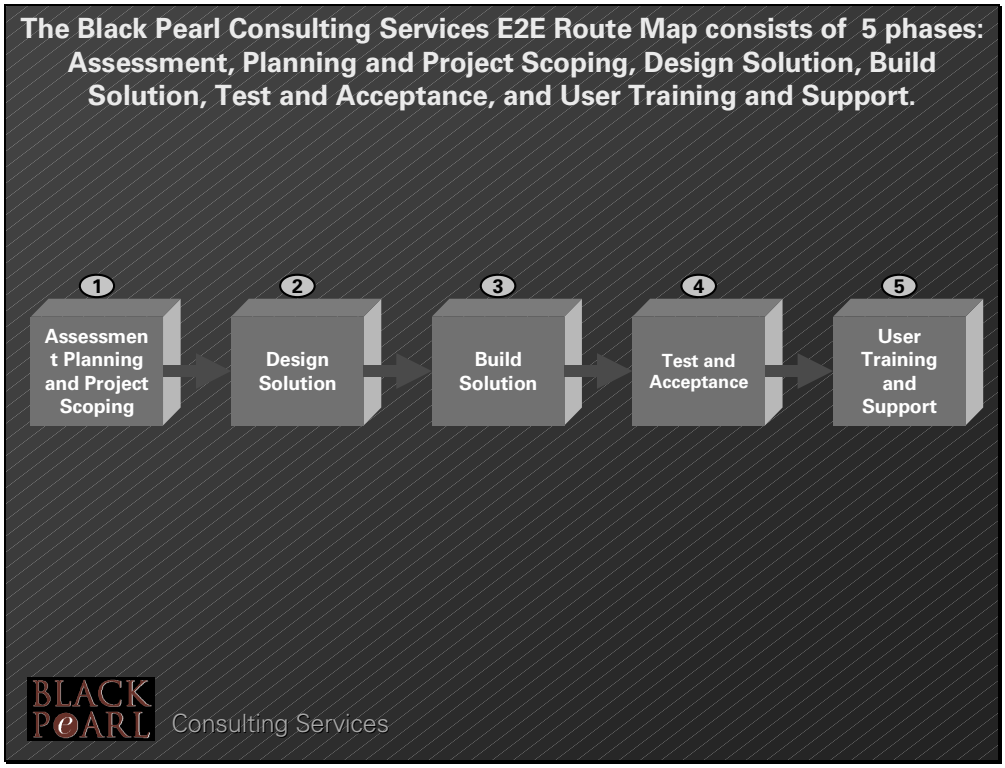
16. What is Knowledge?



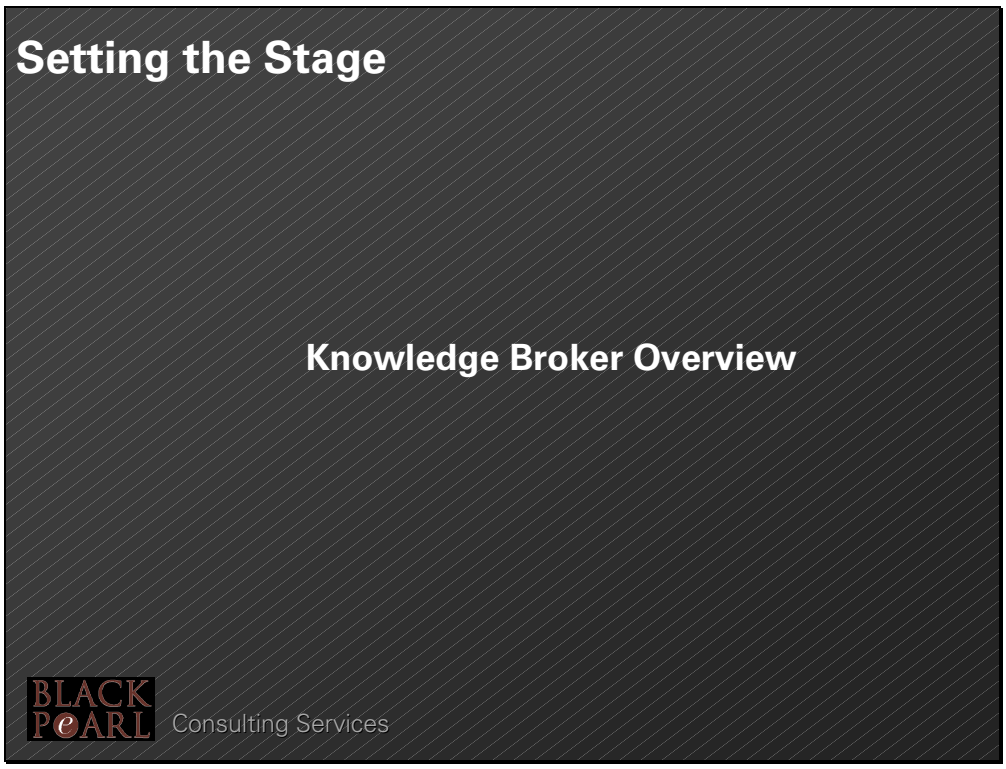
17. Business Context Solutions



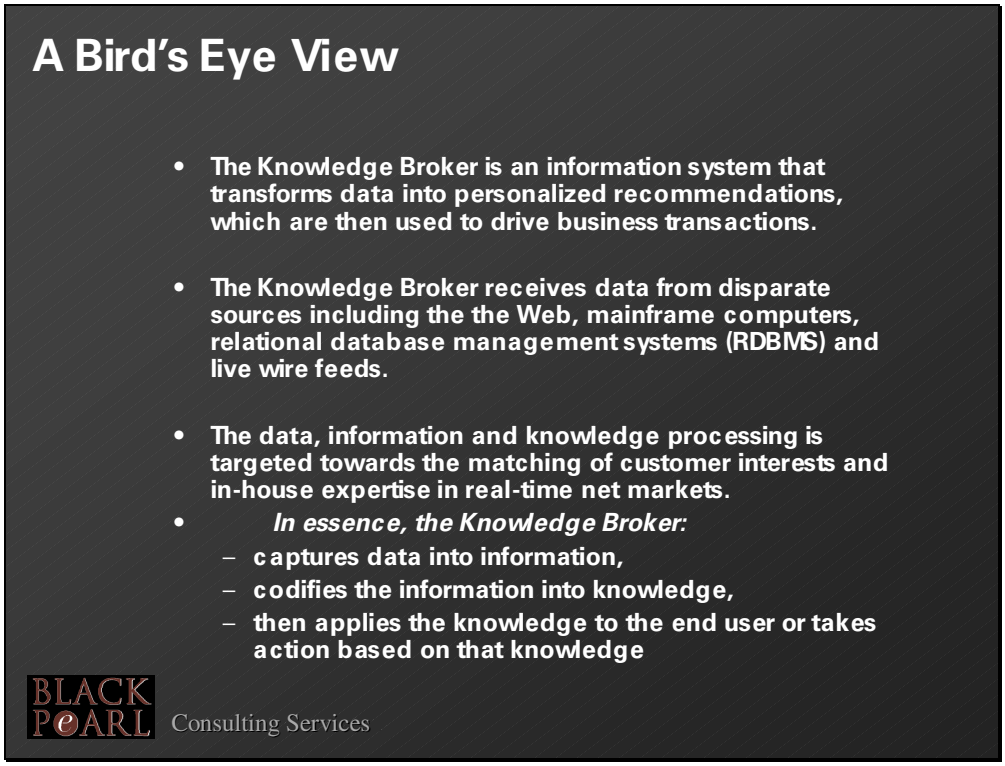
18. Black Pearl Consulting Services



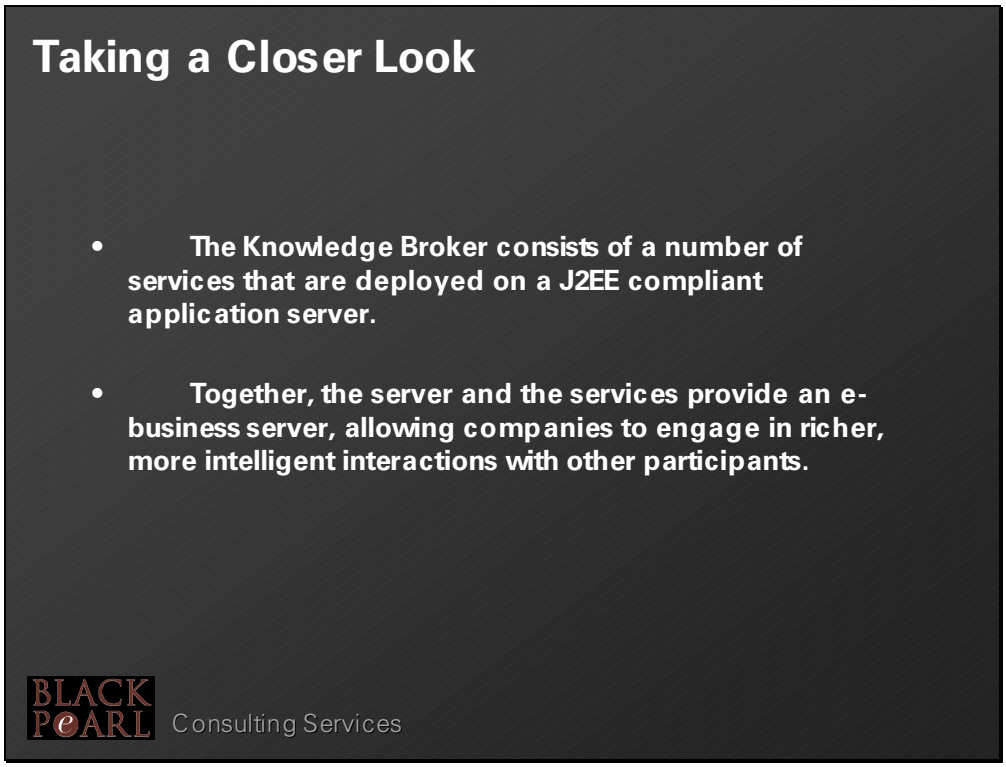
19. Setting the Stage



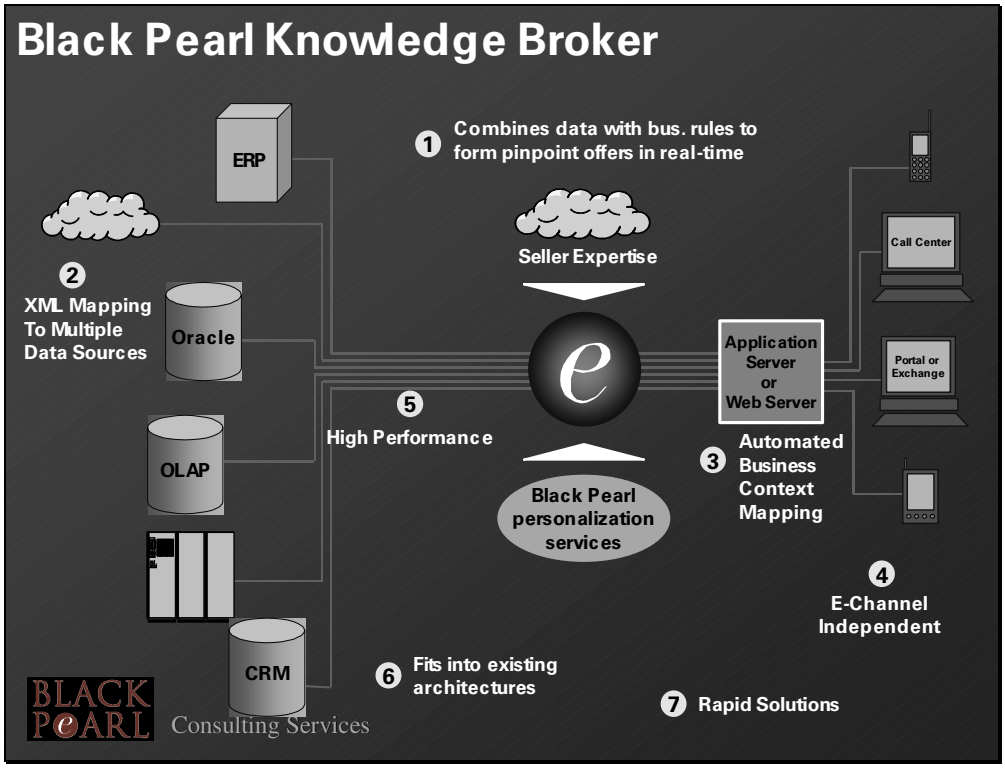
20. A Bird's Eye View of Knowledge Broker



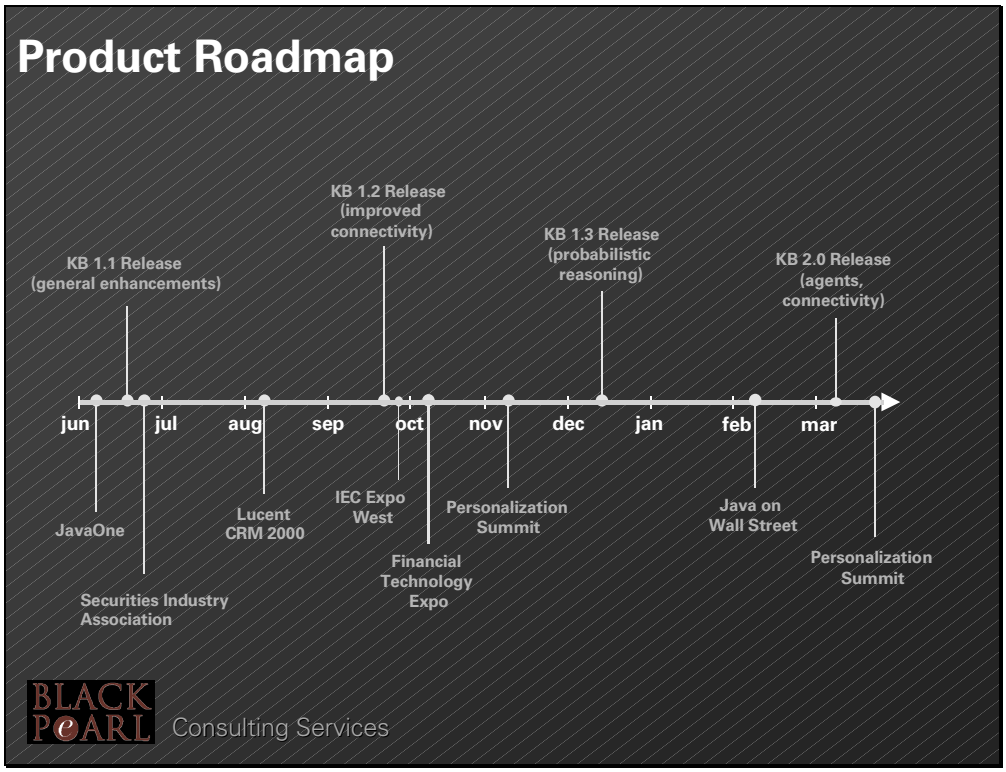
21. A Closer Look at Knowledge Broker



22. Black Pearl Knowledge Broker



23. Product Roadmap



24. Customers

Customers

- Merrill Lynch wants to offer custom-tailored buy/sell offers to its clients — first, call centers, then, the web and, eventually, handheld devices.

Lucent Technologies wants to provide tailored online provisioning to small businesses through multiple channels: web, wireless, call centers, email and fax.

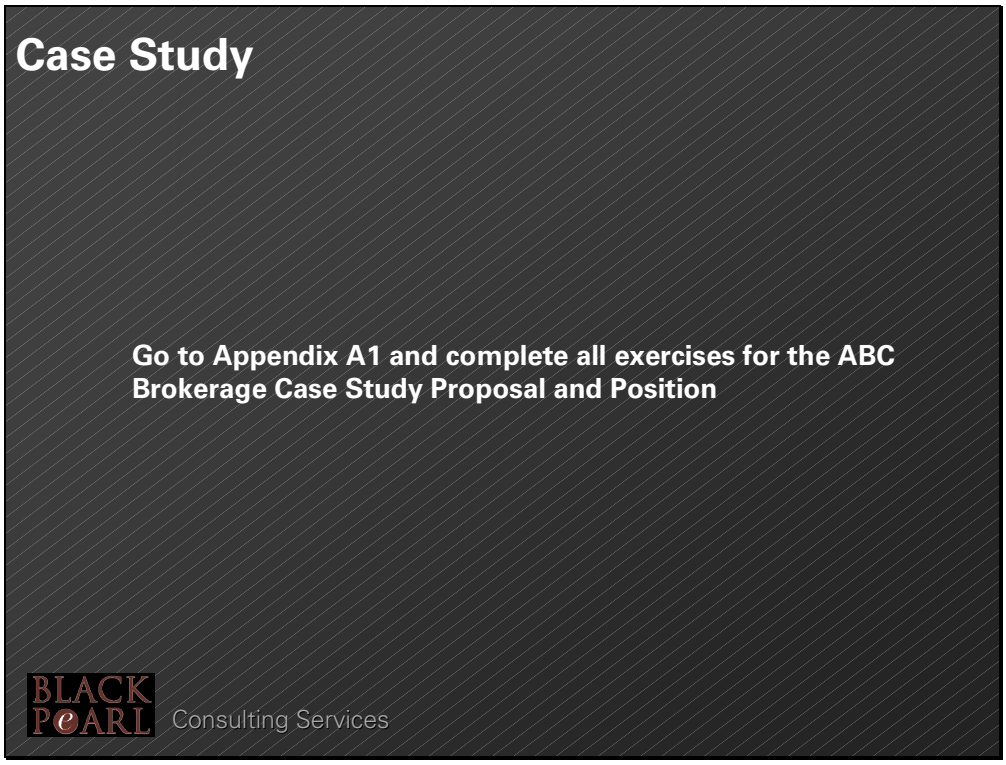
PricewaterhouseCoopers wants to optimize their clients' working capital holdings against dynamic industry performance — rolling it out to thousands of consultants to consistently monitor client financial health.

25. Partners


Partners
intel.
AVAYA
communication
The Former Enterprise Networks Group
of Lucent Technologies
hp
invent
eLoyalty
SYBASE
ORACLE
SOFTWARE POWERS THE INTERNET
Deloitte
Consulting
bea
Sun
microsystems
BLACK
PEARL
Consulting Services

26. Black Pearl Solution Benefits

27. Case Study



BLACK PEARL

Unit 2: Approach to Assessment and Planning

Black Pearl Software, Inc.
November 2, 2000

Note to participant

The names, characters, places, and incidents in this workbook are fictitious. Any resemblance to actual businesses, products, persons, events or locales is entirely coincidental.

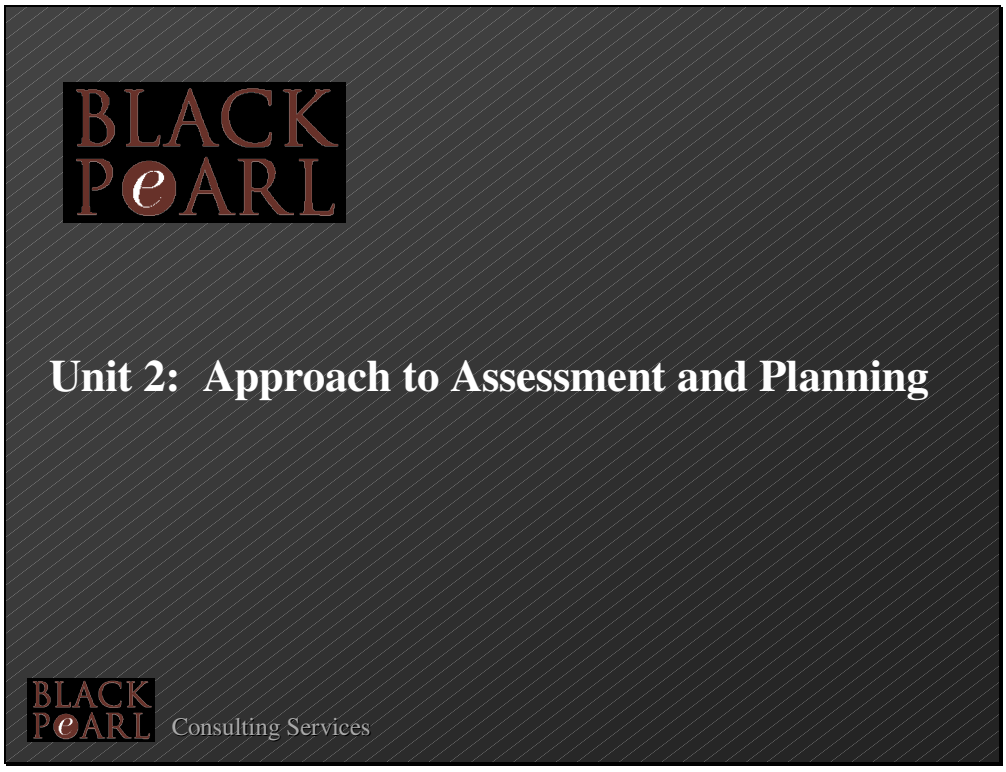
Your organization has signed a contract and non-disclosure entitling your participation in this training program. This entitles you to participate in this training and to use and retain these materials. Neither this program nor any part thereof may be reused or reproduced in any form or medium now known or hereafter invented or created without the express written permission of Black Pearl, Inc.

000X

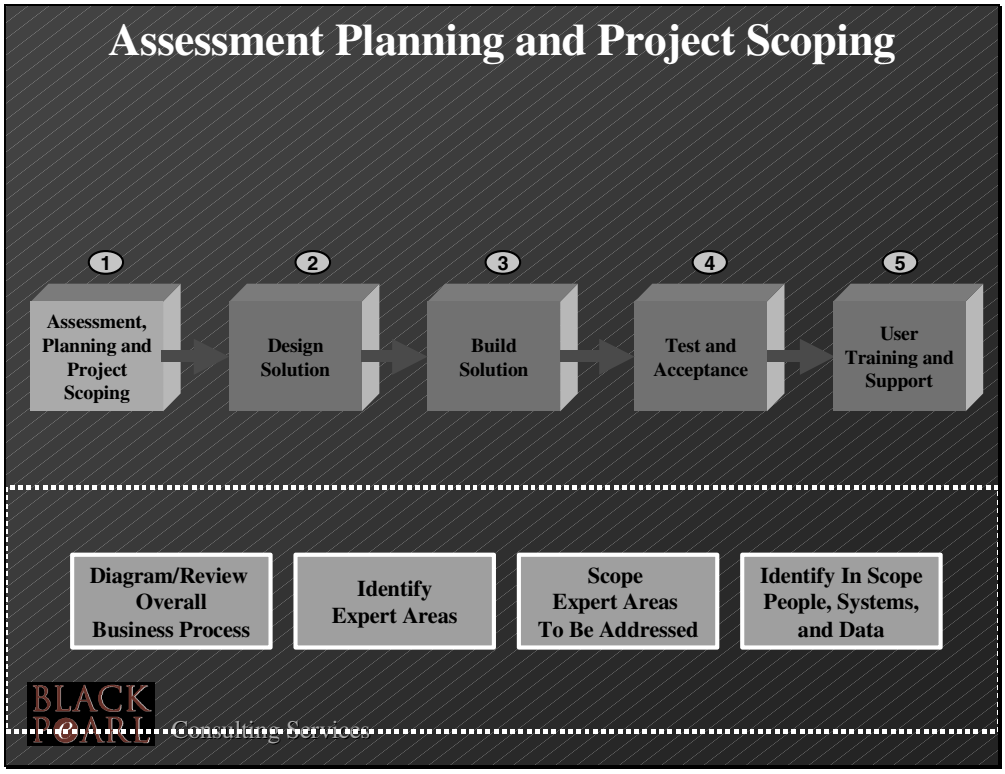
Contents

Contents

1. Approach to Assessment and Planning



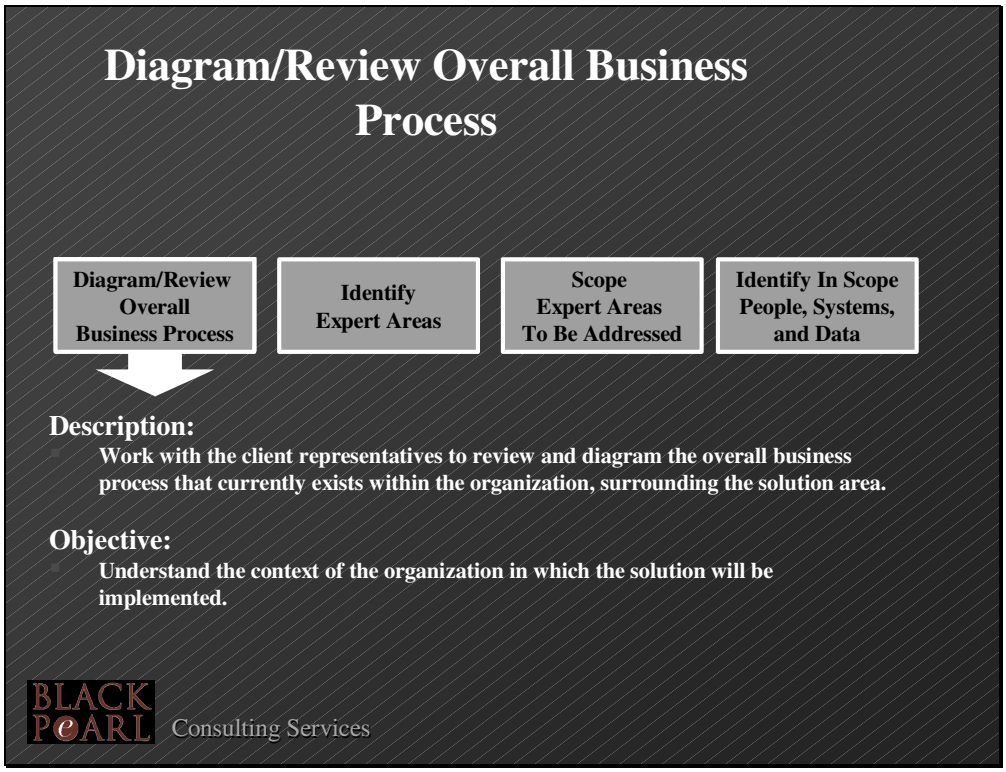
2. Assessment Planning and Project Scoping



Attaining an understanding of the overall process and the components involved in the process is critical to the success of a Knowledge Engineering project. The initial project meeting with the project sponsor should be spent performing the following four steps:

- Diagram/Review overall business process
- Identify expert areas
- Scope expert areas to be addressed
- Identify in scope people, systems, and data

3. Diagram/Review Overall Business Process



Reviewing and diagramming the overall business process is a working session where all of the business activities surrounding the solution area are identified and documented. A workflow diagram with swim lanes to identify any customer, company, system, and/or data interactions, has proven to be an effective method for documenting the initial context of the solution area. The workflow diagram facilitates the discussion by graphically depicting the process. Each lane may or may not be used depending on the complexity and structure of the solution area. Business activities are placed in the respective swim lane according to the following criteria:

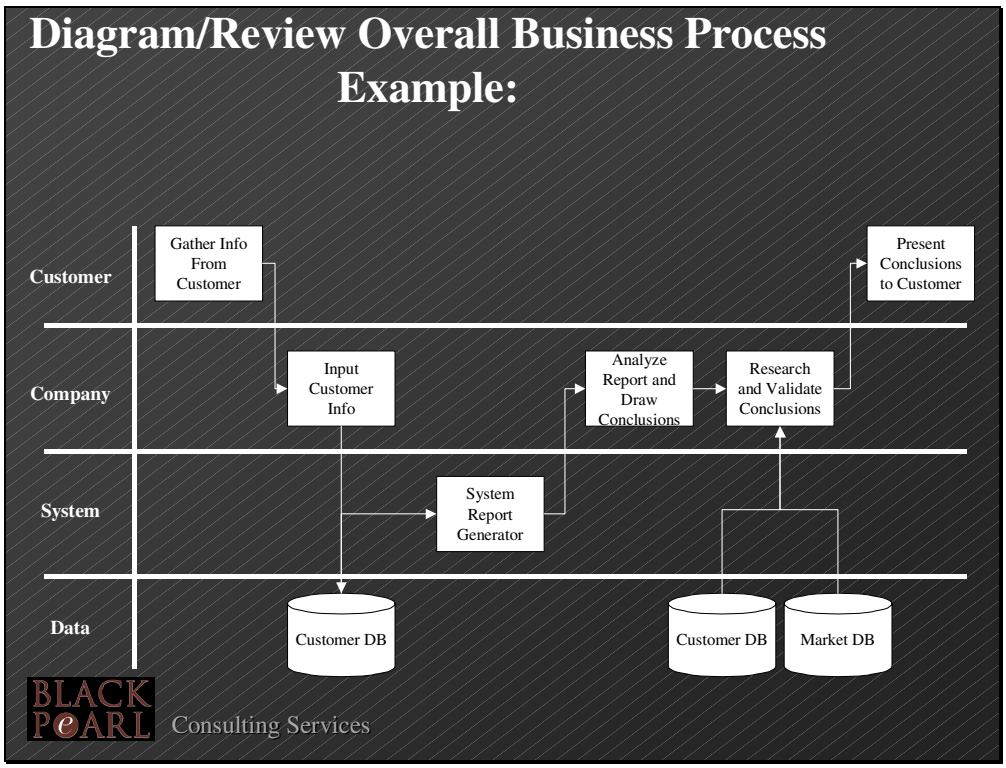
Customer—Business activities that involve customer interaction or are completely customer driven (example: customer interview).

Company—Business activities that are performed by an individual or group within the organization (example: credit approval).

Systems—Business activities that are completely performed by a system process (example: daily reporting)

Data—Data stores that are contributed to or used by any of the above groups.

4. Diagram/Review Overall Business Process



This example depicts a generic business process driven by customer input.

5. Identify Expert Areas

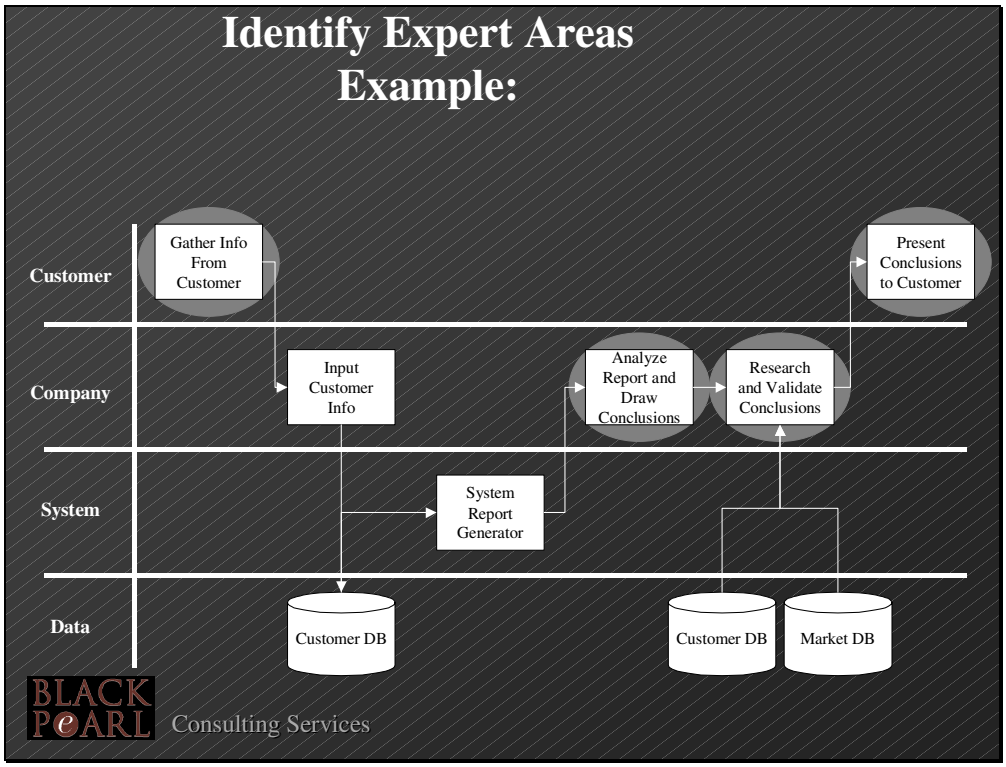
All of the activities within the Customer and Company lanes of the process diagram should be analyzed to ensure that all expert areas are identified. An activity is labeled an expert area if it requires any kind of specialized training, knowledge, expertise, and/or experience to perform.

Traits of Expert and Non-Expert Areas:

Expert Areas—Requires freethinking, decision making, and/or judgment.

Non-Expert Areas—Typically very structured (i.e. uses scripts) and/or constrained (i.e. only limited options are available).

6. Identify Expert Areas Example:



This example shows that there are 4 expert areas that exist within the overall process. For this example, the activities were included or excluded as expert areas based on the following descriptions from the client:

Gather Info From Customer—This activity was described as being somewhat free form in its approach. Even though the individual, who is responsible for gathering the information from the customer, had an initial set of questions, the majority of the information gathering was done with follow-up questioning.

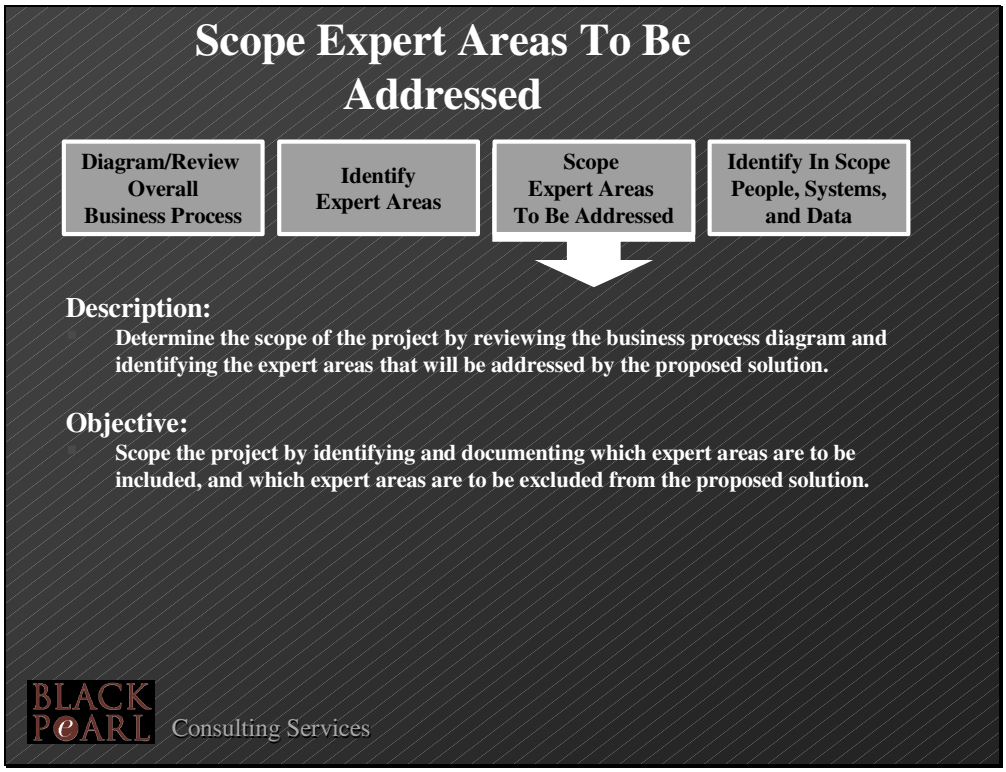
Input Customer Info—This activity was described as basic data entry.

Analyze Report and Draw Conclusions—This activity was described as requiring industry knowledge and experience. The group performing this activity examines the information provided in the report and proposes some conclusions about the customer's situation.

Research and Validate Conclusions—This activity was described as requiring industry knowledge and experience. The group performing this activity must justify the conclusions proposed by the previous group by researching additional information about the customer.

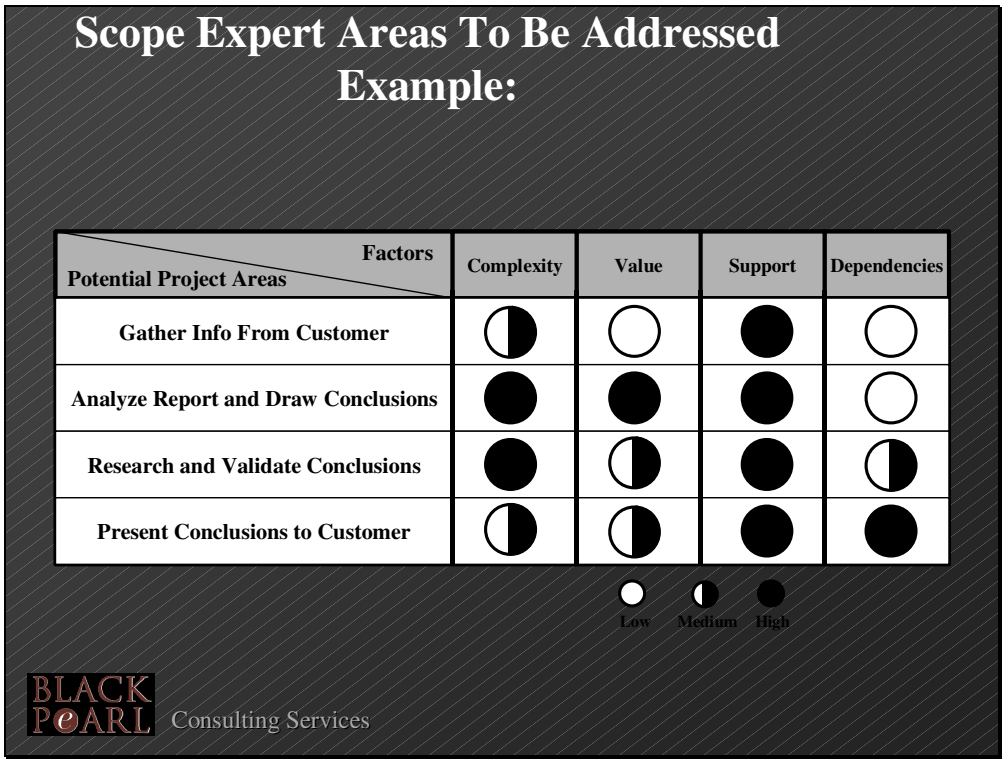
Present Conclusions to Customer—This activity was described as requiring selling expertise. The individual, who presents the conclusions, customizes their presentation to the customer, based on what conclusions they are given and how much supporting research they have to justify these conclusions.

7. Scope Expert Areas To Be Addressed



The focus of the knowledge engineering process can be misdirected if it is unclear which activities are included within the scope of the project. When working with the client to determine which expert areas are included, it is important to consider the complexity of each expert area as well as the value, relationship, and dependencies within each area.

8. Scope Expert Areas To Be Addressed Example



Each Expert Area should be mapped against the following criteria to help the Client to determine an appropriate scope of the project:

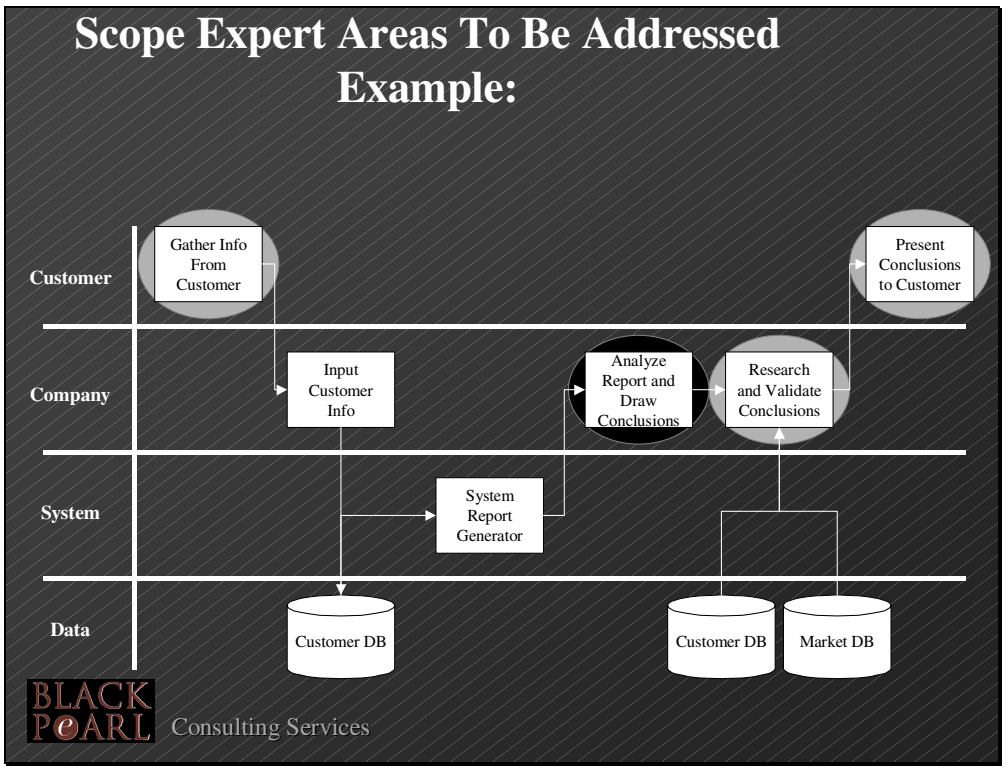
Complexity – The relative complexity of each Expert Area.

Value – The relative value that the company recognizes in each Expert Area.

Support – The relative amount of expertise that exists within the company for each Expert Area.

Dependencies – The relative number of dependencies (hand offs) that exist for each Expert Area.

9. Scope Expert Areas To Be Addressed Example:



This example shows that even though 4 expert areas exist within the process, only 1 expert area will be addressed for this project. For this example, the Client agreed that each of the expert areas were included or excluded based on the following:

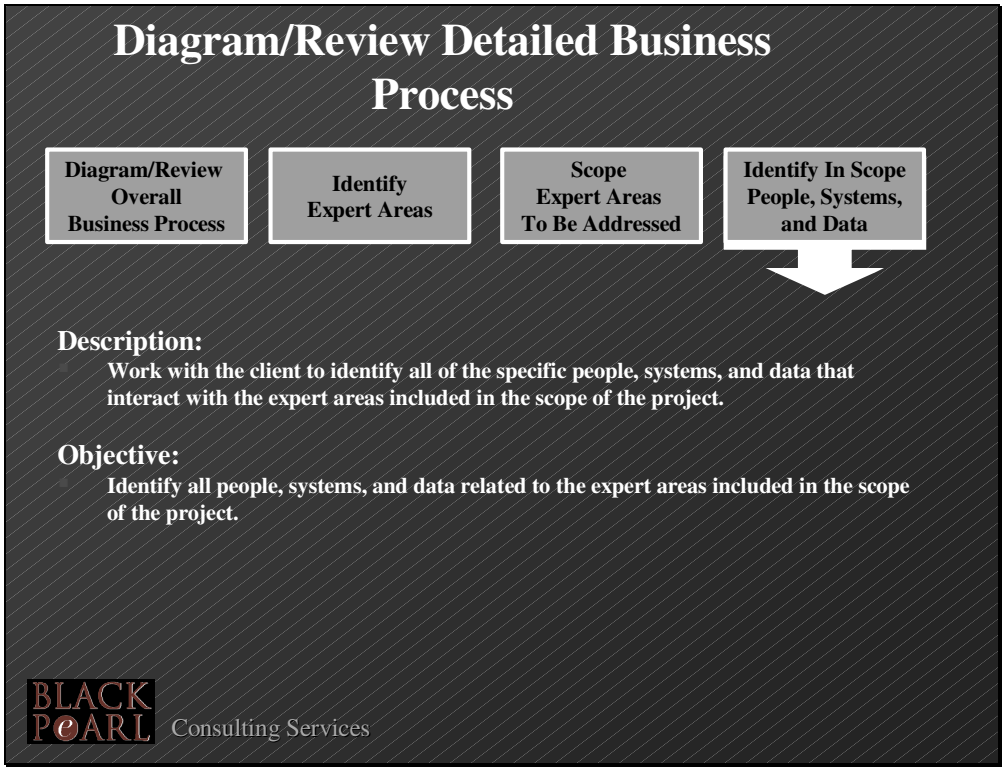
Gather Info From Customer—This expert area was excluded from the scope of the project, although it is valuable, the area currently only requires a single resource to perform and it was not determined to be a bottleneck to the process.

Analyze Report and Draw Conclusions—This expert area was included in the scope of the project since it was a high value area that was currently extremely constrained in qualified resources.

Research and Validate Conclusions—This expert area was excluded from the scope of the project. Although it is a high value area to the Client, its inclusion would have pushed out the completion date of the overall project.

Present Conclusions to Customer—This expert area was excluded from the scope of the project. Although it is valuable, the area currently only requires a single resource to perform and it was not determined to be an overall bottleneck to the process.

10. Diagram/Review Detailed Business Process



In order to move the project forward, all of the people, systems, and data involved in the expert area must be identified and documented.

People—The people who are most important to the success of the project are the experts within the organization. The experts need to be identified as soon as possible and the Client project sponsor needs set up workshop sessions with these individuals and the project team, to begin requirements gathering.

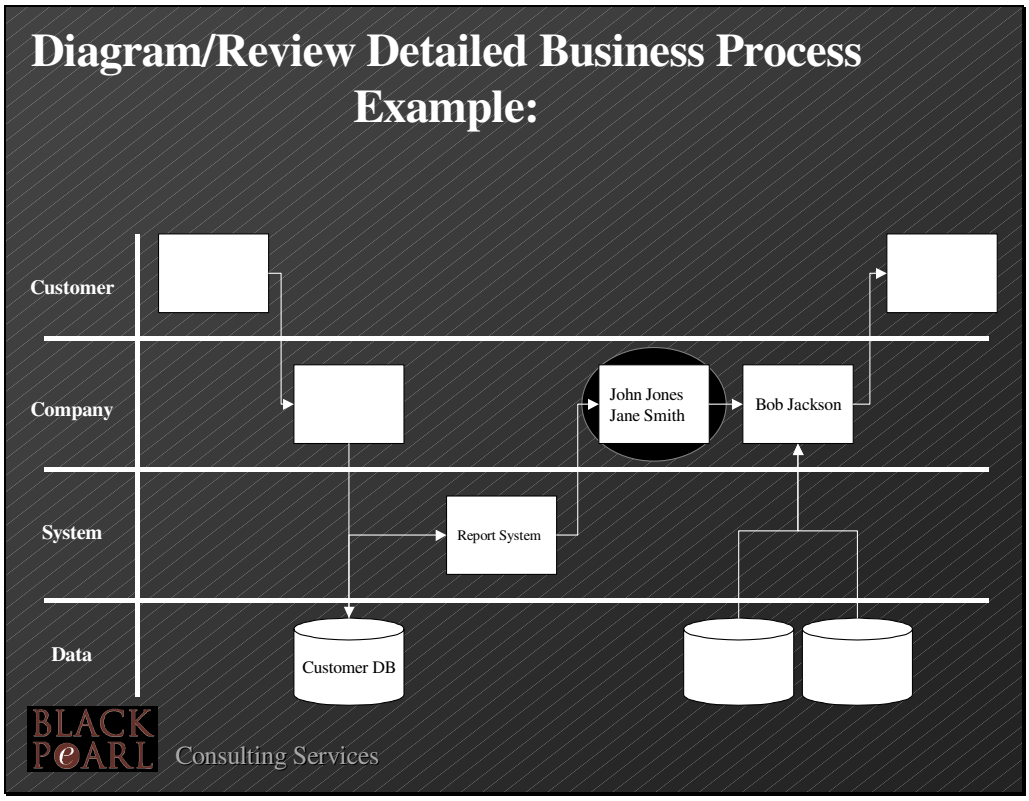
Systems—Any system processes, which are involved in or around an expert area, should be investigated to understand impact on the project. A system process can have impact in two ways:

- **Within expert area**—Processing logic is to be incorporated into the solution. In this situation, it is important to determine the inputs, outputs and the processing logic within the system.

- **Around expert area**—The solution will accept output or provide input to a current system process. This situation requires that the input or output methods from the current system be fully understood so that the interface will function as intended.

Data—Any data source, which is used by an expert area, should be investigated. The data, data models, and data documentation relating to these data sources should be reviewed as soon as possible to understand what available data and to identify compliance or other issues.

11. Diagram/Review Detailed Business Process Example:



This example shows the details for the portion of the overall process determined to be in scope. The experts for the area have been identified as well as the people or systems immediately upstream and downstream in the workflow. It is important to understand both the inputs and the outputs to the expert area to design the solution to work in the context of the overall business process. The implications for an incomplete or incorrect understanding of the expert area are unexpected project issues and budget or scope overruns.

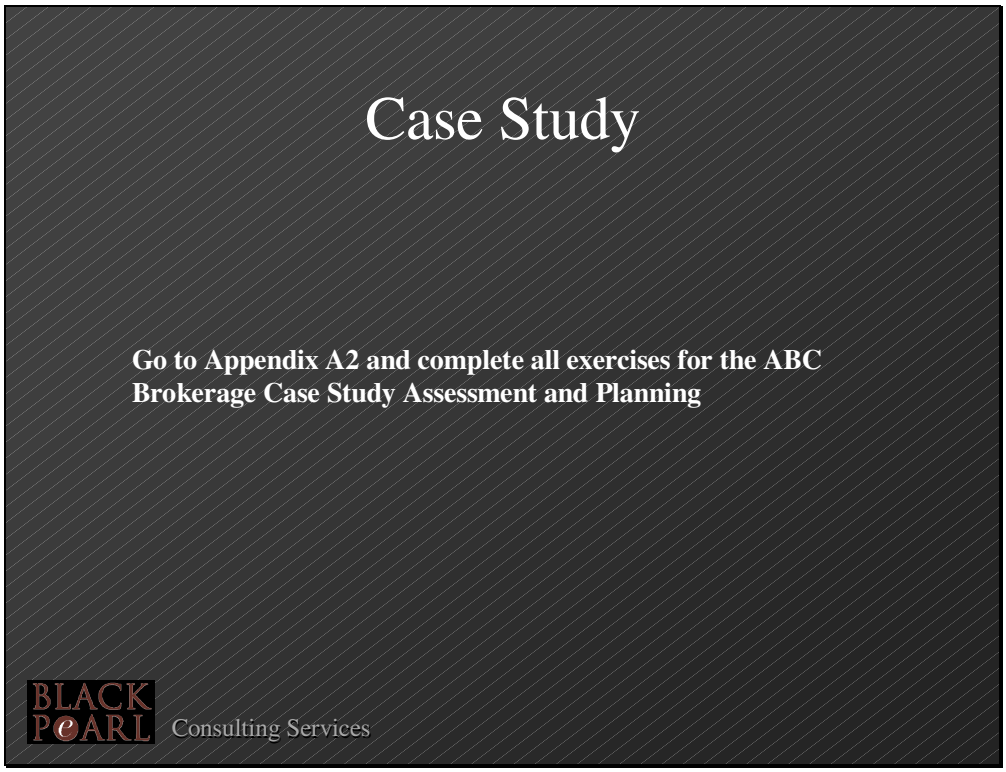
In the diagram above we have indicated the people, data and systems that must be fully understood to properly review and diagram the detailed business area. Shown below is the list of interviewees, systems and data sources.

People—John Jones, Jane Smith, Bob Jackson

Systems—Report System

Data—Customer DB

12. Case Study



BLACK PEARL

Unit 3: Approach to Knowledge Engineering

Black Pearl Software, Inc.

November 2, 2000

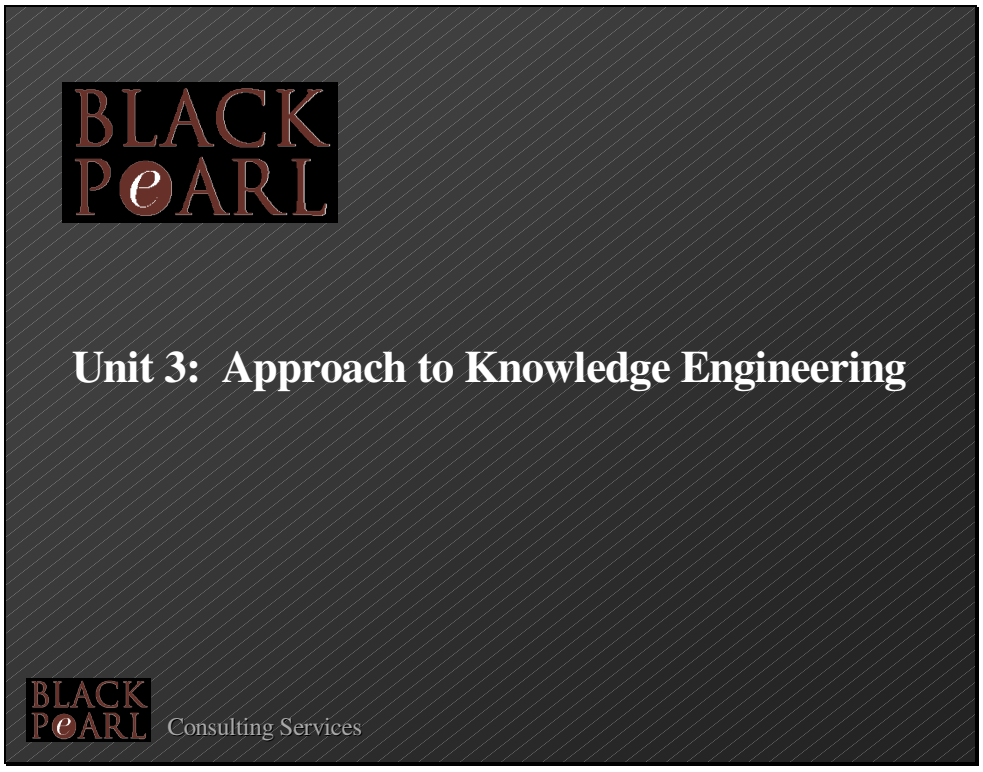
Note to participant

The names, characters, places, and incidents in this workbook are fictitious. Any resemblance to actual businesses, products, persons, events or locales is entirely coincidental.

Your organization has signed a contract and non-disclosure entitling your participation in this training program. This entitles you to participate in this training and to use and retain these materials. Neither this program nor any part thereof may be reused or reproduced in any form or medium now known or hereafter invented or created without the express written permission of Black Pearl, Inc.

Contents

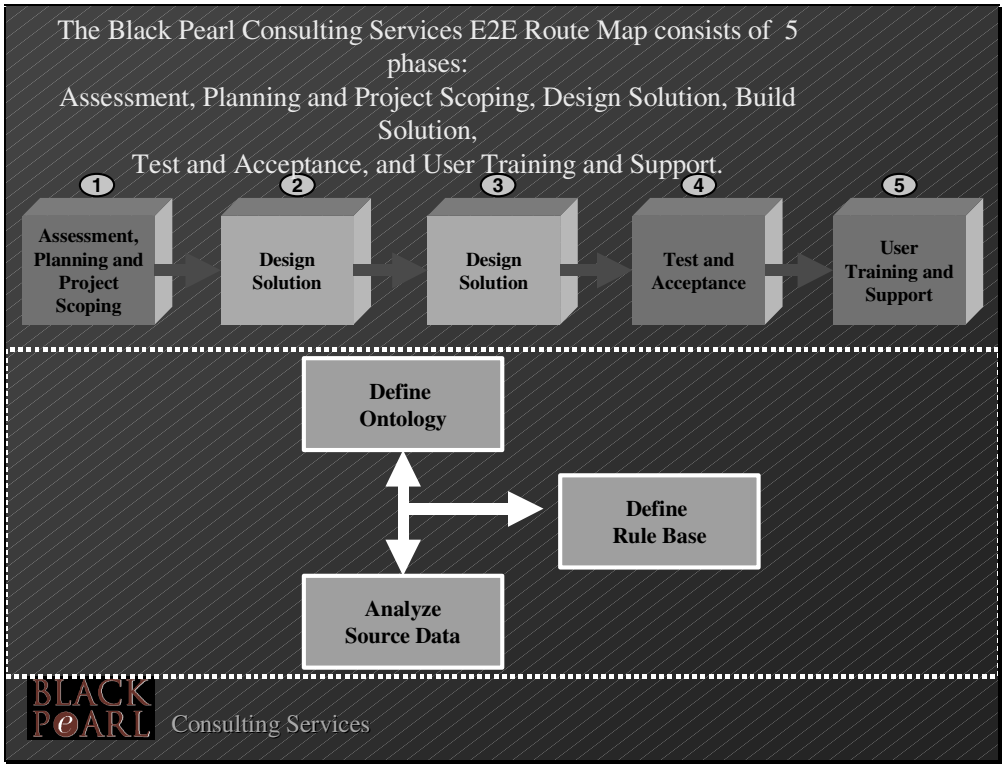
1. Approach to Knowledge Engineering



NOTE: This Unit deals with the methods used to define an ontology and a rule base for a project. As with any engineering project, it is recommended that specialist analysis tools be used to document and seek approval for the requirements definition. This makes life easier for both the analyst and the sponsor. The choice of analysis tool will ultimately be at the discretion of the client and may range from unstructured tools such as Visio, Excel, and Word to specialist tools such as KADS22 (from the University of Amsterdam) and m2 (from Modelware, Australia).

The examples shown in this presentation use m2, as this is the tool of choice of Black Pearl Professional Services. These examples could be translated into other formats, if required. The overriding purpose of this presentation is to teach the **method** by which a rule base is developed.

2. E2E Route Map



The first phase will have covered:

Defining the context in which the decision-making activity takes place. This provides the scope for the project by defining the activities and actors that will be addressed, as well as those activities and actors that are prerequisites for the decision-making process, and those activities and actors that depend on the process.

Scoping the decision process itself. This involves defining the set of recommendations that will be possible outcomes as well as high level definitions of the ways in which those recommendations are to be reached.

Having scoped the project, the next step is to define the detailed Knowledge Base model. This is comprised of the 3 basic activities shown, namely, **define the ontology**, **analyze source data** and **define the rule base**. These 3 tasks will require several iterations before the Knowledge Base is complete. The order and manner in which each of

these activities is undertaken will vary to some degree from project to project.

The **ontology** defines the vocabulary where the **rule base** is expressed. This is referred to as the **decision environment**. The basic ontology must be defined before any attempt is made to define the rules. But once the process has been initiated, it is possible to build the ontology and rule base in parallel. As part of the process of defining the ontology, a reference must be made to the sources of data that are the physical expressions of the ontology.

3. Define the Ontology

The ontology is a vocabulary used when constructing a rule. The vocabulary is defined to facilitate agreement (and corresponding sign off) by the client as well as to enable reuse. Many projects start as a proof of concept and then progresses to a full development. Even a proof of concept can require an extensive ontology and rule base, comprising of hundreds of elements. In order to manage this scale of model, and to be able to successfully build on it during subsequent projects, a structured approach pays substantial dividends.

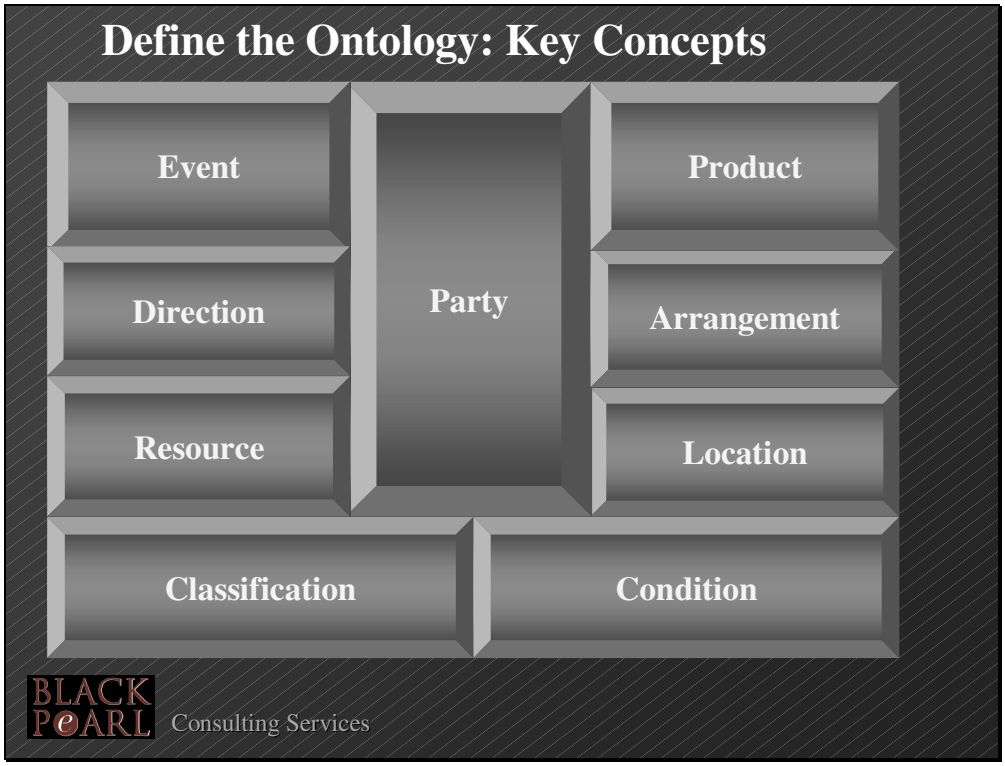
An ontology is a hierarchical structure, allowing more and more definition to be added the deeper you drill down. One of the difficult decisions in starting a new ontology is to define what the top layers of the ontology look like. The approach we have chosen is to always start an ontology with a predefined set of 9 **key concepts**, into which all subsequent items

will be categorized. These 9 key concepts include concepts such as Party and Product; further details are given in the following slide.

Each **key data concept** is then **sub-classified** according to the sub sets of each key data concept that have meaning within the decision-making environment. For instance, **Party** may be sub-classified as **Customer**, **Supplier** etc. Sub-classifications are defined down to whatever level of detail is appropriate. Each sub-classification will inherit the characteristics of its parent classification, for example, relationships and descriptors.

Each sub-classification may be further defined by means of **descriptors** (which provide information that describes a sub-classification, such as Name, ID) and **relationships** to other sub-classifications.

4. Define the Ontology: Key Concepts



These are the 9 **key concepts**, which provide the first level of sub-classification within the ontology.

Party includes any individual, organization or organization unit (such as branch, department etc), for example, Customer, Broker, Agent.

Product includes any product or service, for example, Pension Plan, Mutual Fund, Deposit, and Brokerage.

Event includes any thing that happens that affects the decision making process, for example, product price change, marriage.

Arrangement includes any agreement between two or more Parties, for example, Customer Account, Employment Contract.

Location includes any point of delivery, (for example, telephone number, postal address, e-mail address) or physical area, for example, city, state, country)

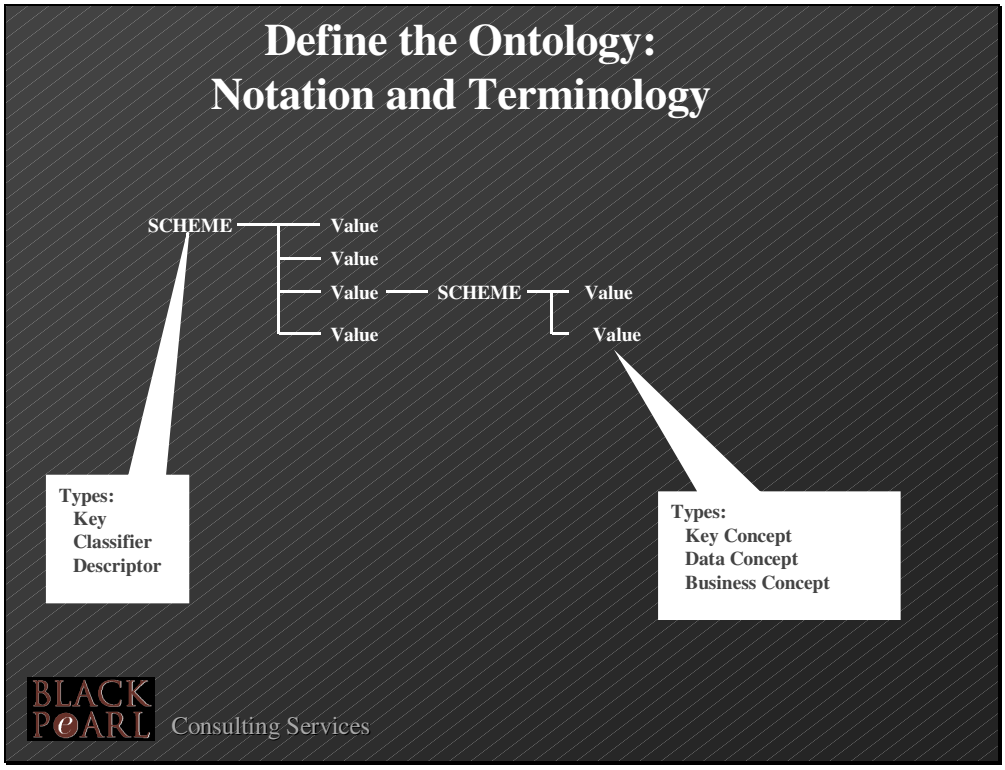
Resource includes any item of value, for example, telephone equipment, computer equipment, stock certificates.

Classification is a means of providing generalized groupings that may apply to more than one of the other key data concepts, for example, Standard Industry Classification

Condition defines conditions that may be attached to one or more key data concepts, for example, the customer sets customer investment preferences and restrictions and the same set of conditions will also be used to classify Products.

Direction specifies the manner in which business is to be carried out. This may be due to internal or external regulation or best practice. Direction also includes actions that may be recommended as a result of rule conditions being met.

5. Define the Ontology: Notation and Terminology



The ontology is documented by a hierarchy of schemes and values, each scheme being described by one or more values, each value may be further subdivided by one or schemes, and so on.

A **SCHEME** may be one of 3 types:

Key the root of the hierarchy (i.e. top)

Classifier a means of classifying a Value (sub class)

Descriptor a means of describing a Value (property)

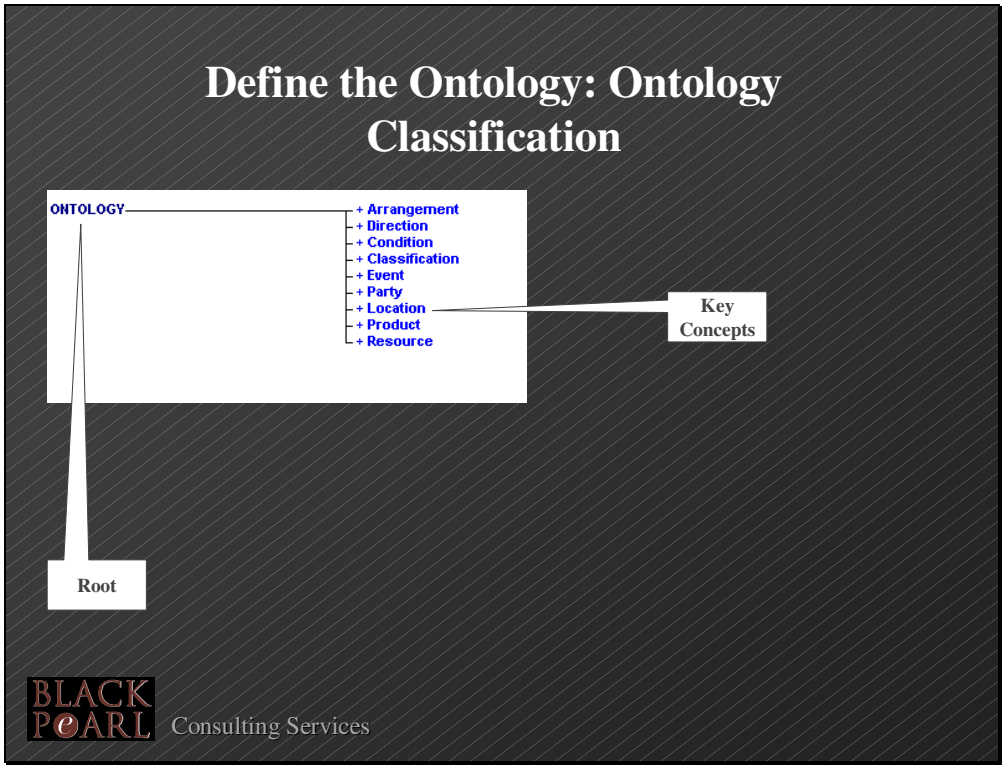
A **Value** may be one of 3 types:

Key Concept—One of the highest level classifications of the ontology.

Data Concept—A major business grouping of data (entity, class)

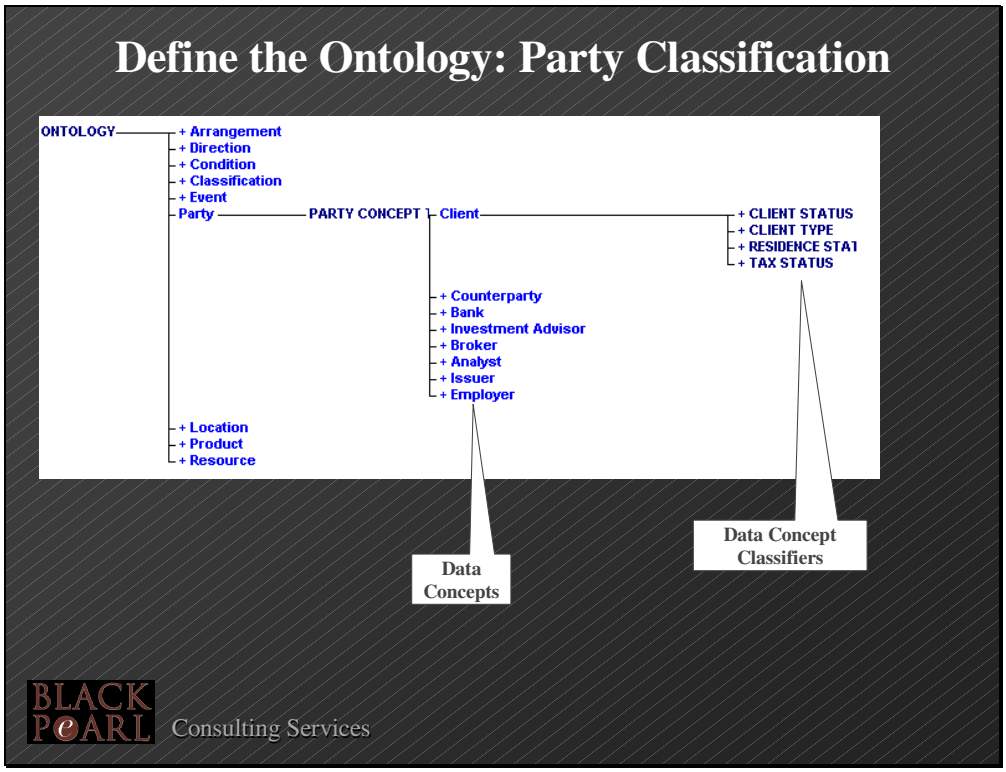
Business Concept—A classification grouping (sub type/sub class/sub set) that will be used within the rule specification.

6. Define the Ontology: Ontology Classification



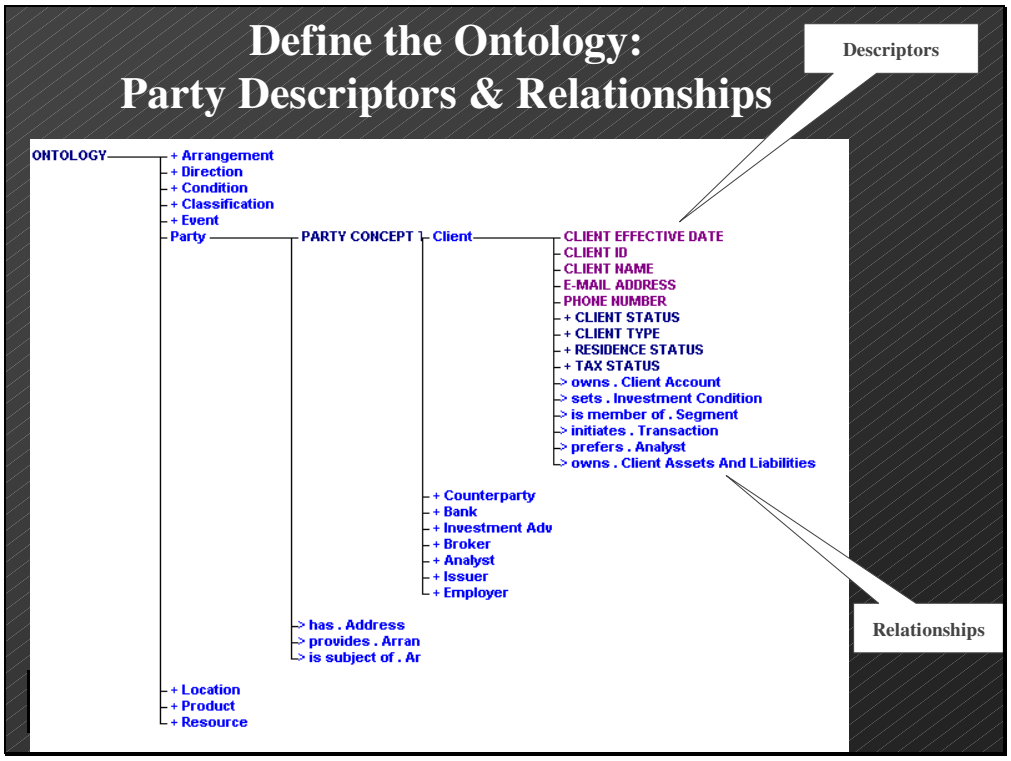
As can be seen here, the ontology is initially classified into the 9 key concepts.

7. Define the Ontology: Party Classification



The key concept Party is sub-classified according to the **data concepts** that are of interest.

8. Define the Ontology: Party Descriptors & Relationships



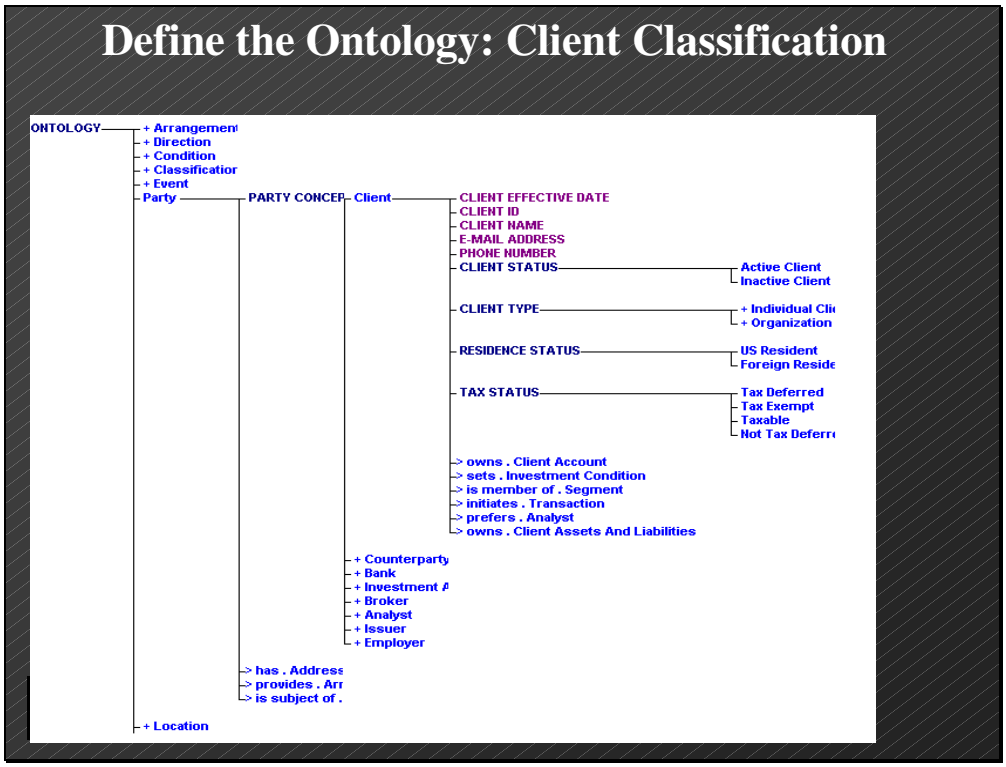
Descriptors and Relationships are defined for each sub-classification, making full use of the inheritance principles implied by the ontology hierarchy.

In the illustration shown, Party has relationships has.address, provides.arrangement, is subject of.Arrangement. This implies that all sub-classifications of Party (i.e. Client, Counter party, Bank etc) also have those relationships.

Client also has its own set of relationships. These only apply to Clients, not the other Party types. As we drill down the hierarchy, we will find further examples of specific descriptors and relationships, particularly when looking at Individual and Corporate Clients.

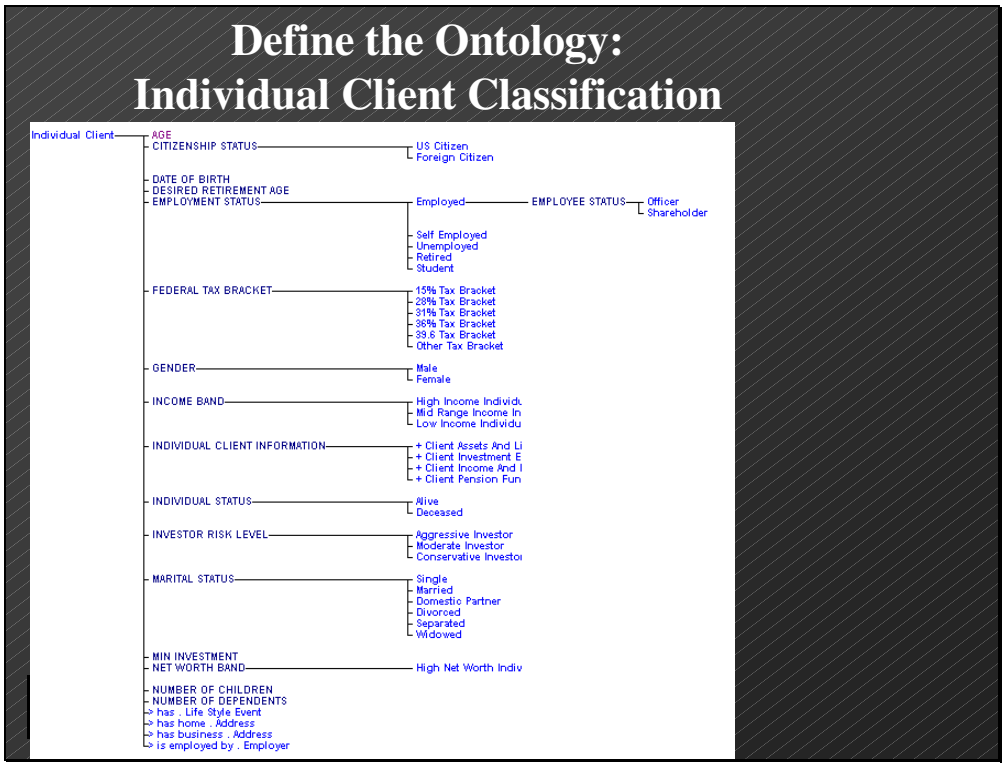
Client has descriptors Client ID, Client name etc.

9. Define the Ontology: Client Classification



Carrying on the drill down through the Client data concept,

10. Define the Ontology: Individual Client Classification



and drilling into Individual Client. Notice that Individual Client is still connected to Client Type, and that the hierarchy has been shrunk.

11. Analyze Source Data

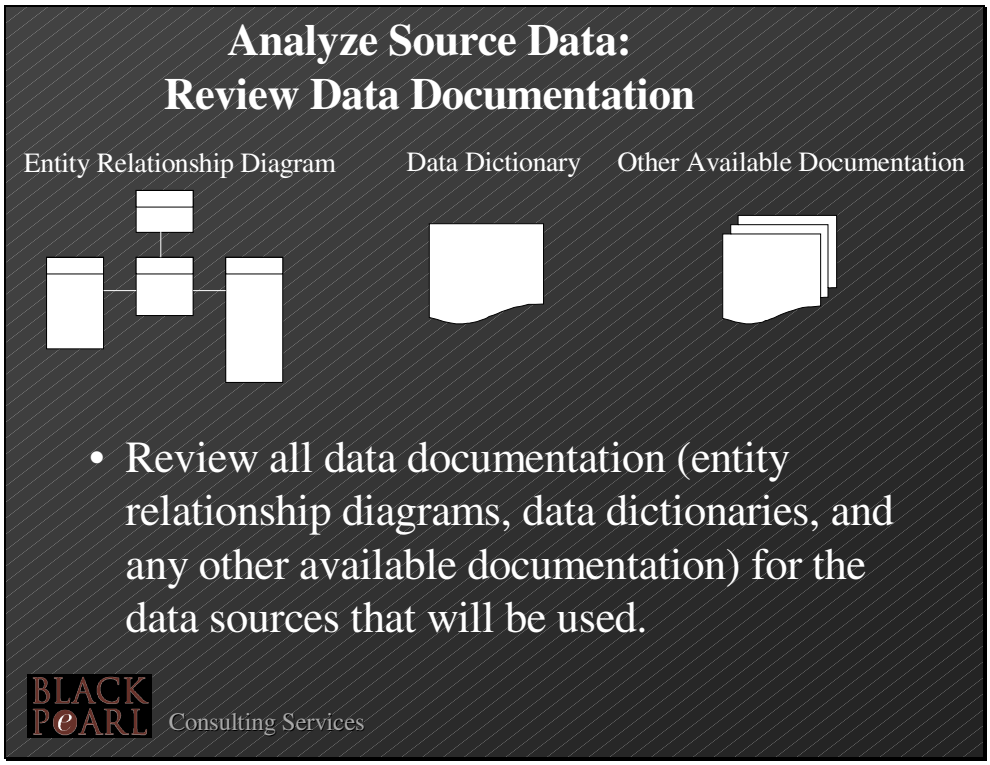
Having determined the initial shape of the ontology based on the business requirements for the rule processing, it is now time to start to flesh out the details by reference to the available data sources. In many organizations, it is possible to find the same information in a variety of databases. The purpose of this first activity is to identify all of the candidate data sources and determine which of these provide the best sources of data.

For the selected data sources, all available documentation is identified and analyzed.

Relevant data is documented.

And finally, the selected and documented source data is mapped to the ontology data elements previously modeled.

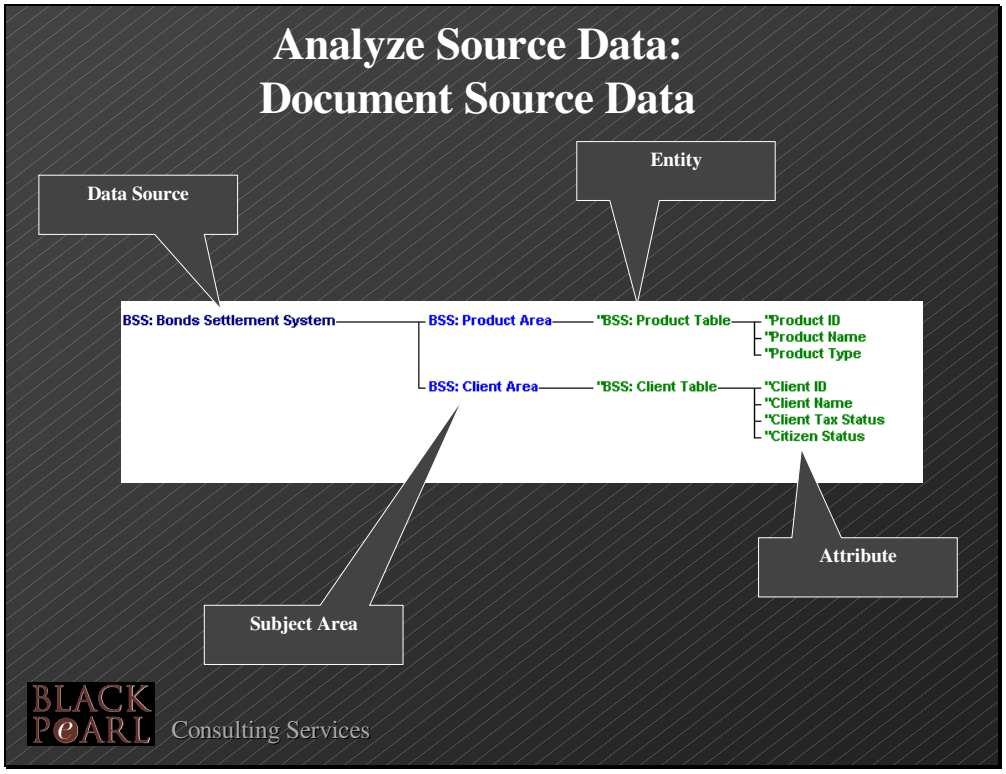
12. Analyze Source Data: Review Data Documentation



The purpose of this task is to analyze and understand the available documentation for the selected data sources. The key points to look for are:

- Data structures (e.g. unique identifiers, relationships between tables)
- Table physical names, logical names, and definition.
- Table column physical names, logical names, value domain, and definition.
- Only business data needs to be analyzed at this stage. Tables used for system control, security, auditing and replicated data (such as print tables) can be ignored.
- While the source data will usually exist on a database, it is also possible to find data on real-time feeds, web pages etc.

13. Analyze Source Data: Document Source Data



When documenting the selected **data sources**, it is only necessary to document data items of interest to the rule base. It is not necessary to document the entire database, or even entire tables.

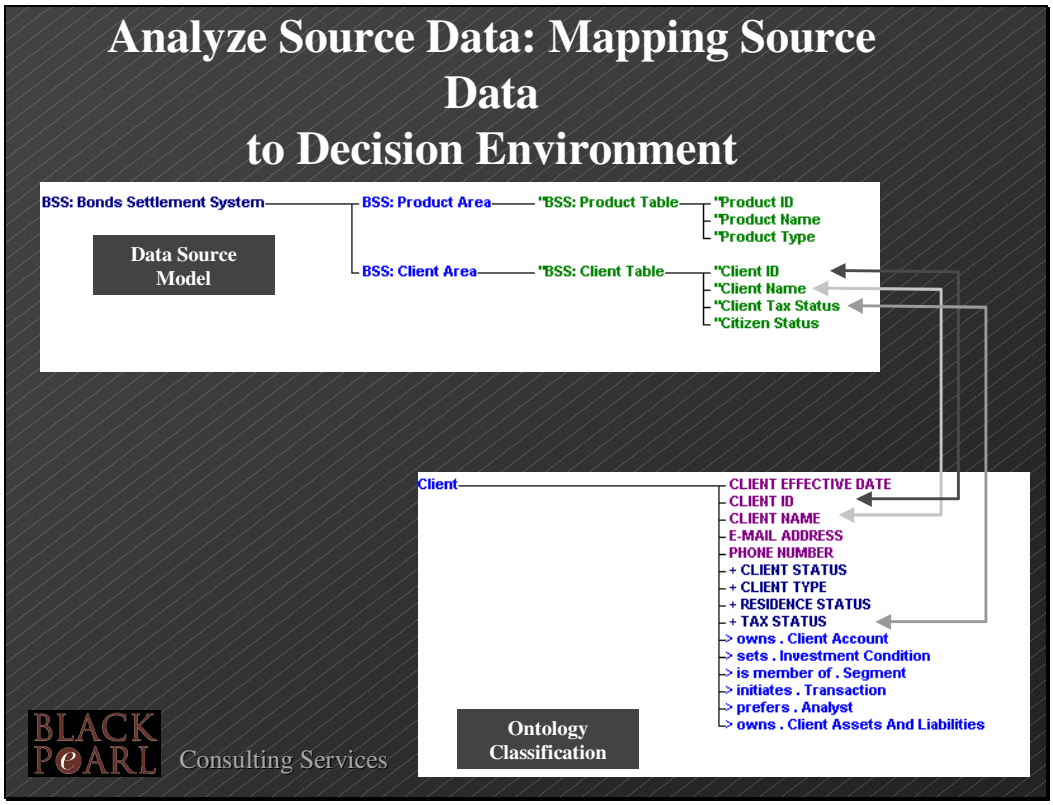
The data sources are documented in a hierarchical manner, with one hierarchy defined for each **data source** (e.g. database). Each data source is subdivided by **subject area**. A subject area represents a logical grouping of data within the data source.

Within each subject area is a list of **entities**. The physical names and definitions (if required) are recorded.

Within each entity is a list of the attributes of interest. Again, the physical names and definitions (if required) are recorded.

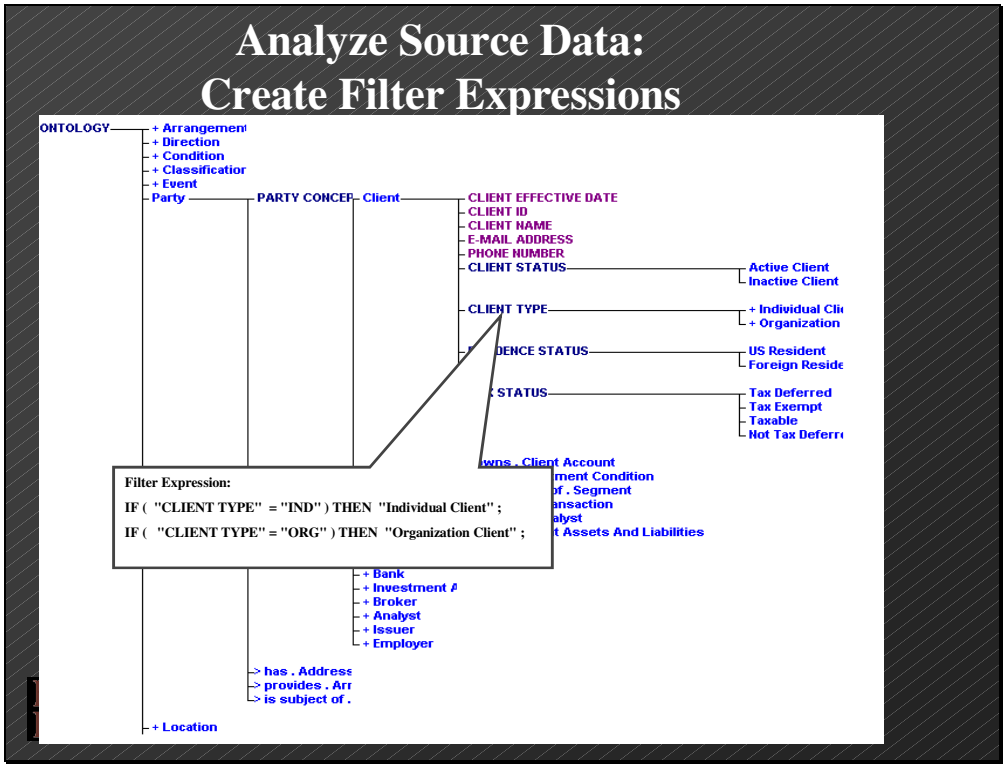
While the example shown here details the documentation of an MS Access database, the same structure can be used for other data formats (data feed, HTML, etc).

14. Analyze Source Data: Mapping Source Data to Decision Environment



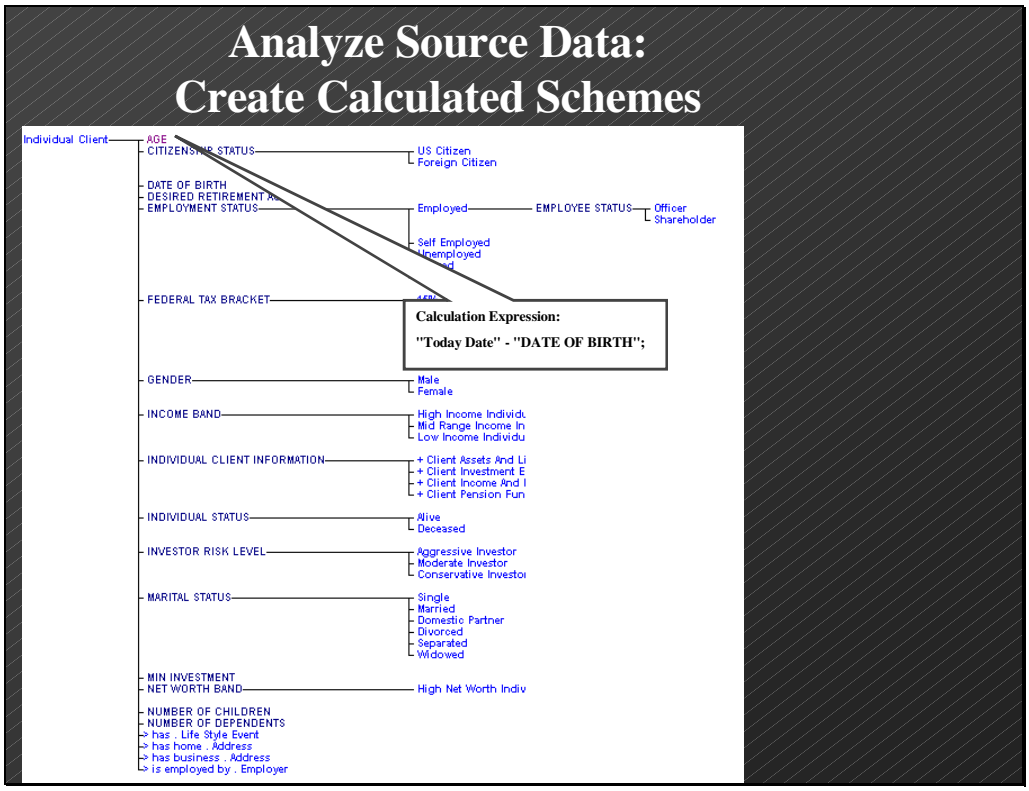
The source data elements can now be mapped to the Ontology classification elements.

15. Analyze Source Data: Create Filter Expressions



A filter expression is required for each scheme that has child values that are to be used while building the rules.

16. Analyze Source Data: Create Calculated Schemes

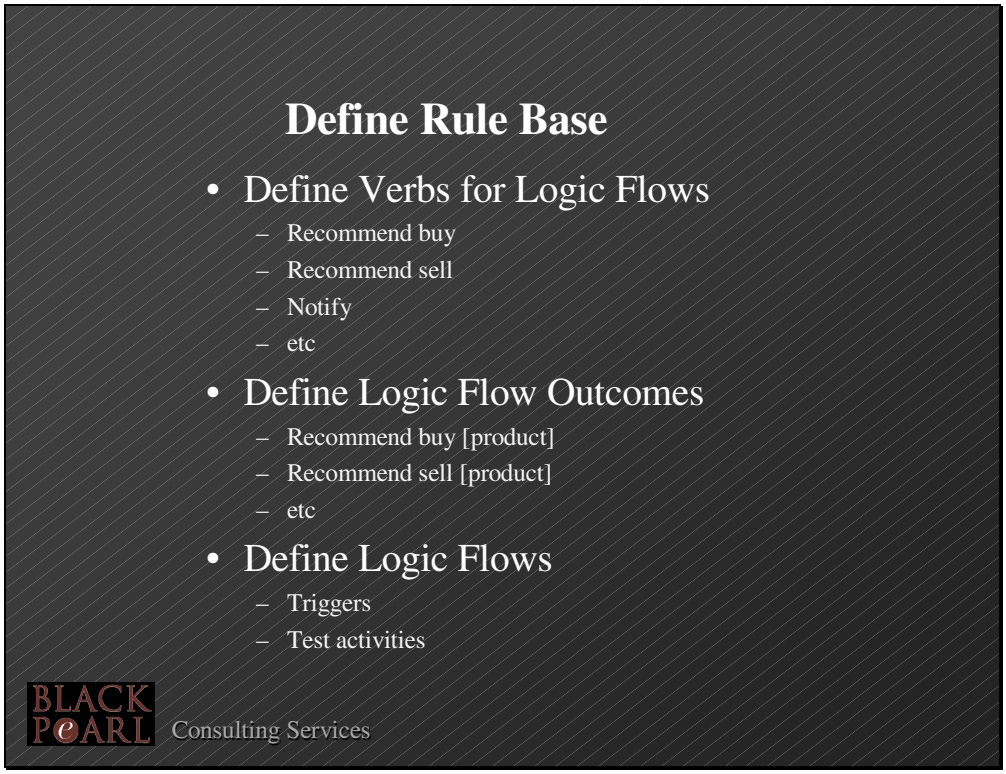


A calculated scheme is derived from other schemes, not sourced directly from the data source. For example, the rules may require making a test based on age, while the source data only contains date of birth. A calculated scheme can be created to calculate age as today minus Date of Birth.

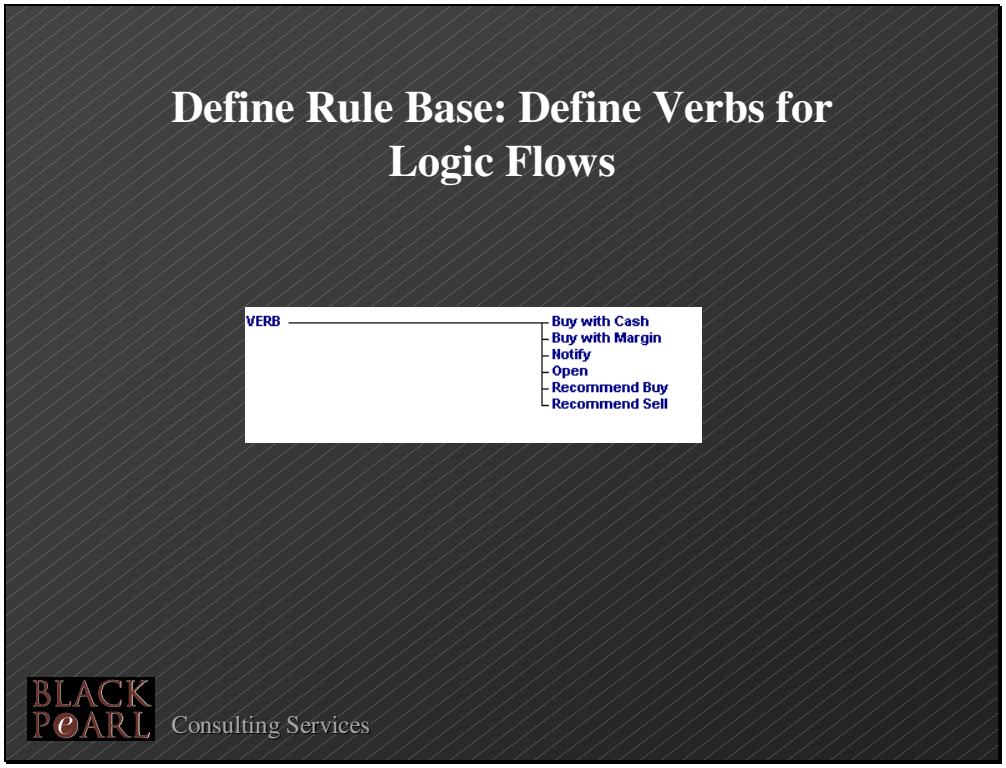
Common calculated schemes include:

- Sum of 2 or more schemes
- Time period between 2 dates
- Future value of funds
- Ratios

17. Define Rule Base



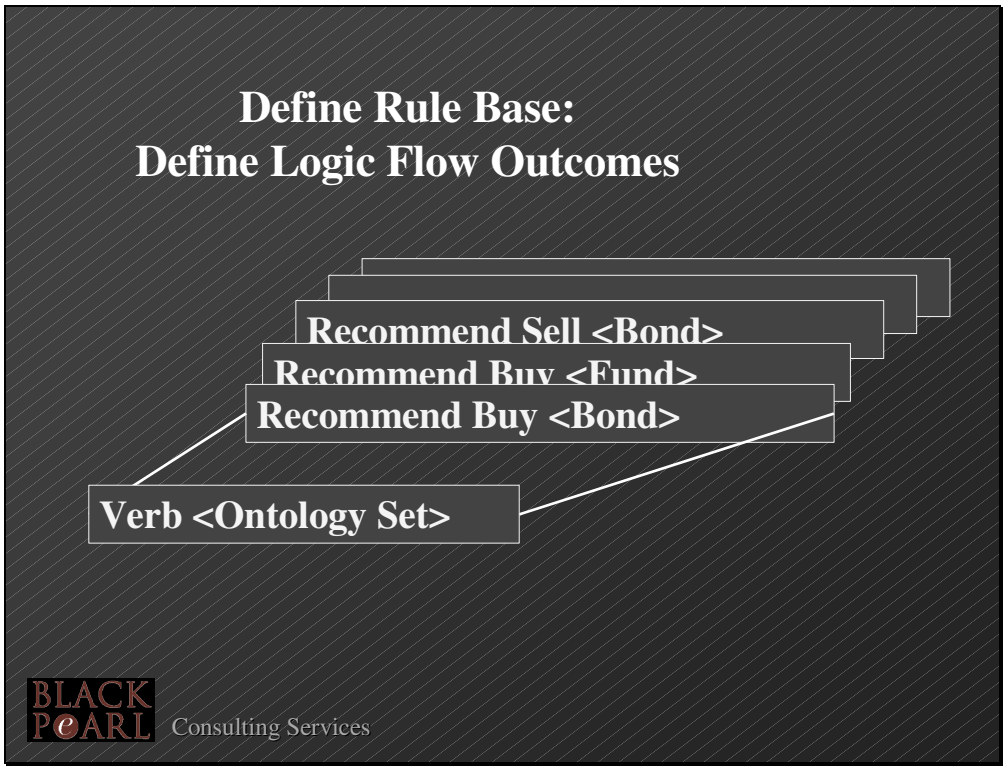
18. Define Rule Base: Define Verbs for Logic Flows



The first stage in defining the rule base is to determine the verbs that will cause an action at the end of each rule. This will have been addressed in Stage 1:**Assessment, Planning and Project Scoping** but may not have been formally documented in the Knowledge Engineering tool.

At its simplest, the rule base will have a single verb such as **Recommend.**

19. Define Rule Base: Define Logic Flow Outcomes

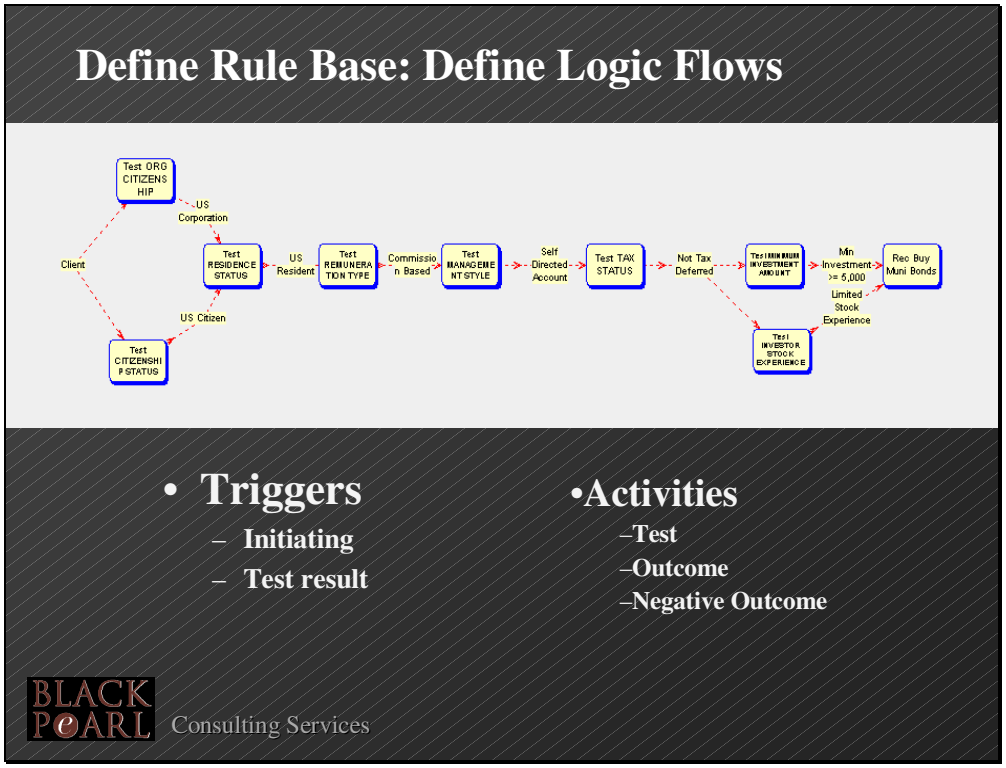


An initial set of logic flow outcomes will have been produced in Stage 1: **Assessment, Planning and Project Scoping.** This must now be completed and formally documented.

The format of a logic flow outcome is verb (see previous slide) followed by a result set defined by the ontology. The most common ontology sets are either defined under the Product data concept, or a recommendation data concept that lists all the recommendations.

An ontology set may be a single or multiple instances of data. For example, Recommend Buy Bonds will produce a result set of all Products that are Bonds.

20. Define Rule Base: Define Logic Flows



A workflow is a means of graphically representing a rule. It has a single start point (known as the **Initiating Trigger**), proceeds through a number of test points (**Test Activities**), and terminates in an **Outcome Activity**.

The diagram notation shows triggers as red dotted lines with directional arrows and activities as yellow boxes.

Trigger names correspond to **Business Concepts** in the ontology.

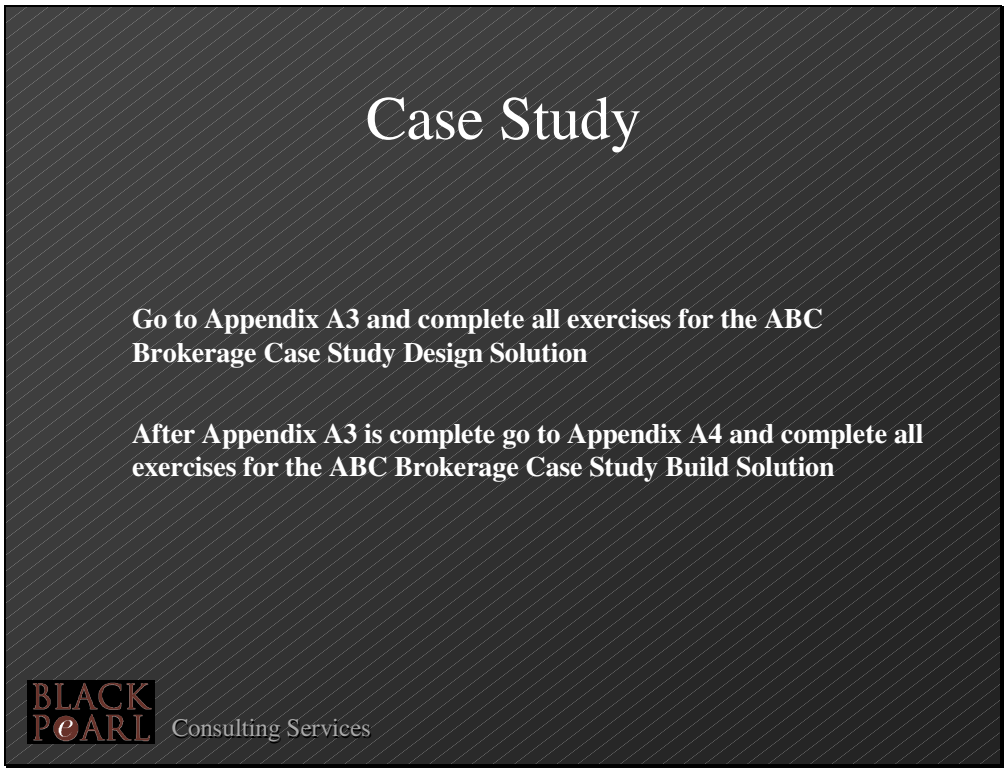
Test Activity names have the format TEST <ontology scheme>. The output triggers from each Test Activity correspond to the scheme's child values.

Flow generally proceeds from left to right; there should be no reversals of direction (definitely no loops). This is not a flow chart.

Tests that proceed sequentially correspond to AND connections.

Tests that proceed in parallel correspond to OR connections.

21. Case Study



BLACK PEARL

Unit 4: Knowledge Broker

Technical Implementation

Black Pearl Software, Inc.
November 2, 2000

Note to participant

The names, characters, places, and incidents in this workbook are fictitious. Any resemblance to actual businesses, products, persons, events or locales is entirely coincidental.

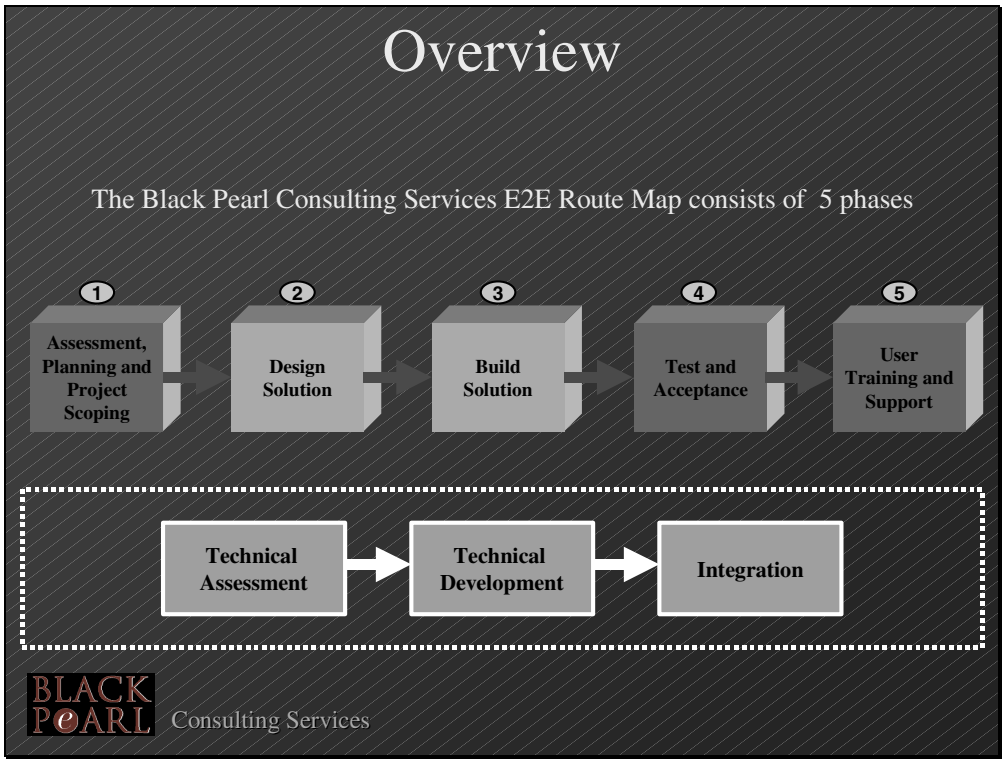
Your organization has signed a contract and non-disclosure entitling your participation in this training program. This entitles you to participate in this training and to use and retain these materials. Neither this program nor any part thereof may be reused or reproduced in any form or medium now known or hereafter invented or created without the express written permission of Black Pearl, Inc.

000X

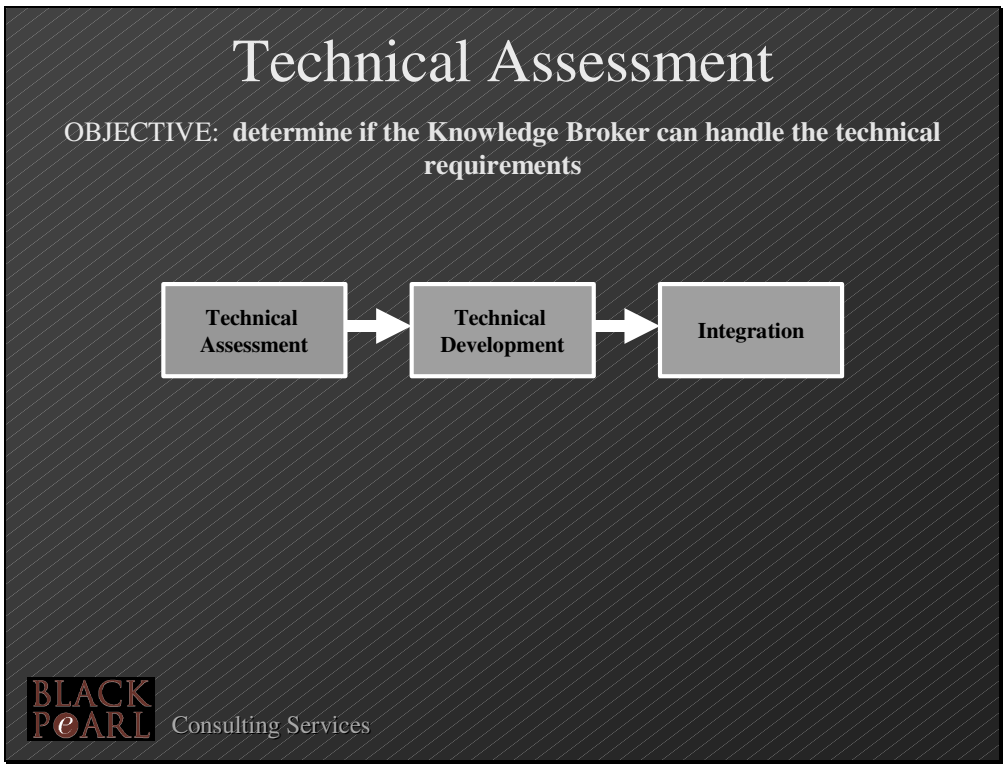
Contents

1. Knowledge Broker Technical Implementation

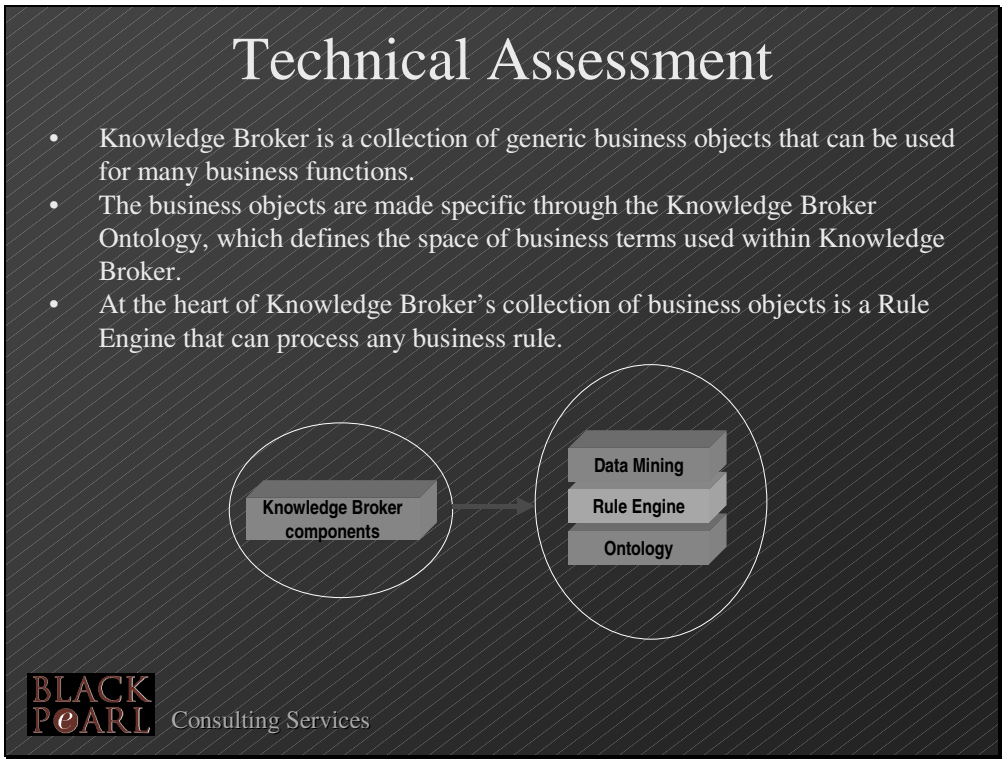
2. Overview


Overview
The Black Pearl Consulting Services E2E Route Map consists of 5 phases
1
Assessment, Planning and Project Scoping
2
Design Solution
3
Build Solution
4
Test and Acceptance
5
User Training and Support
Technical Assessment
Technical Development
Integration
BLACK PEARL
Consulting Services

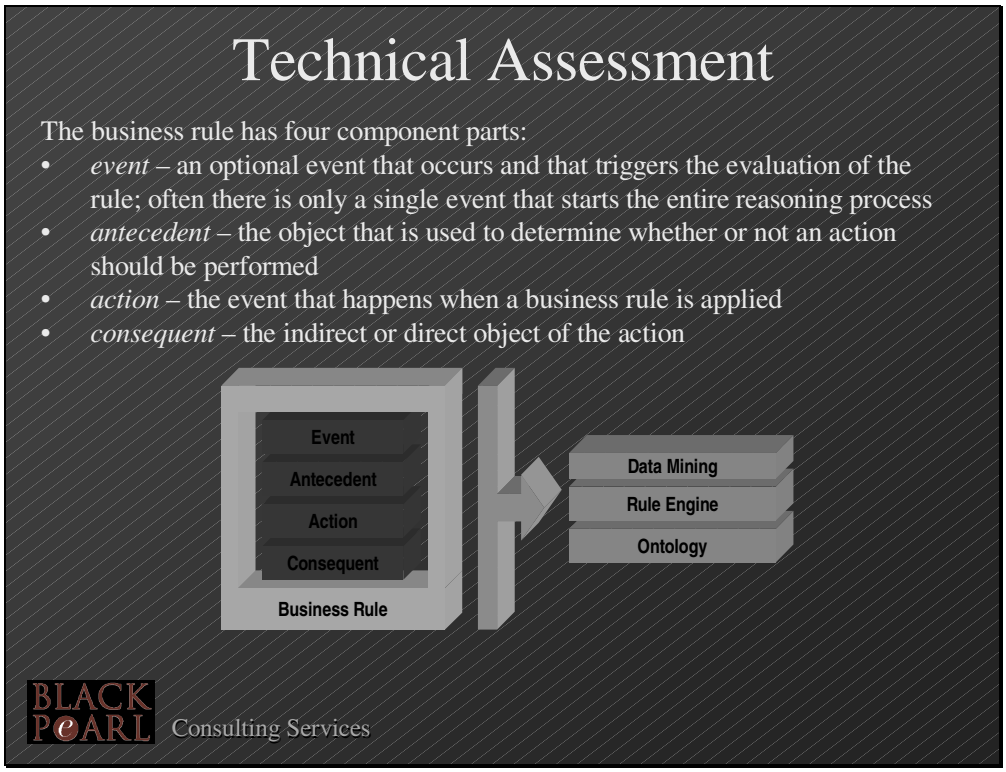
3. Technical Assessment



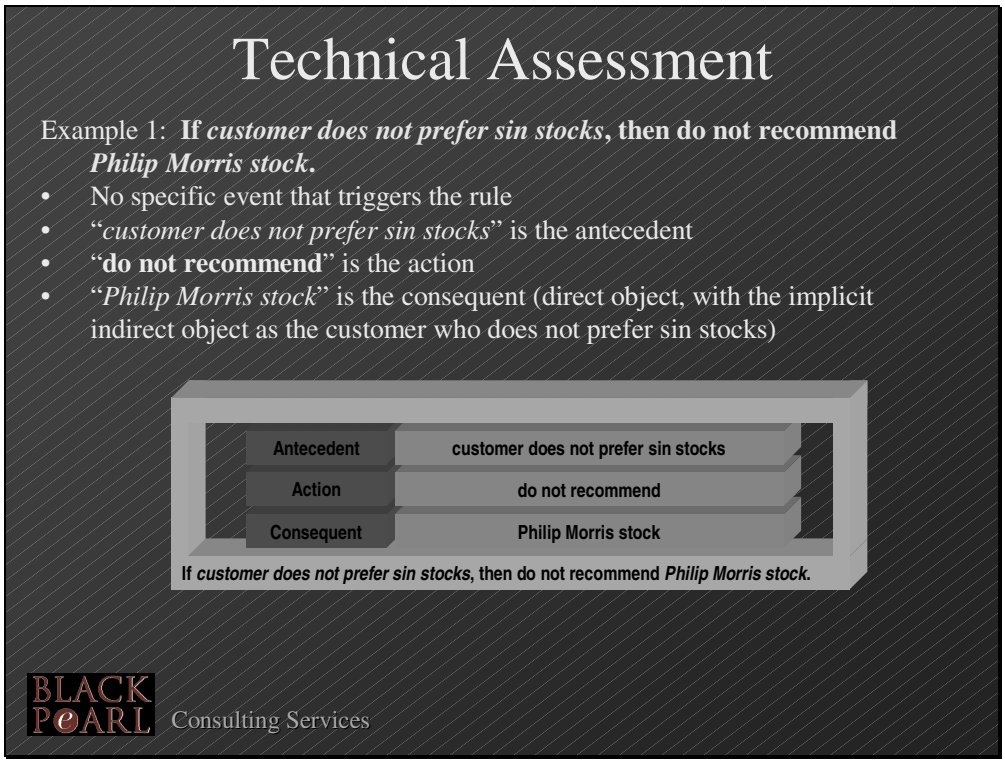
4. Technical Assessement



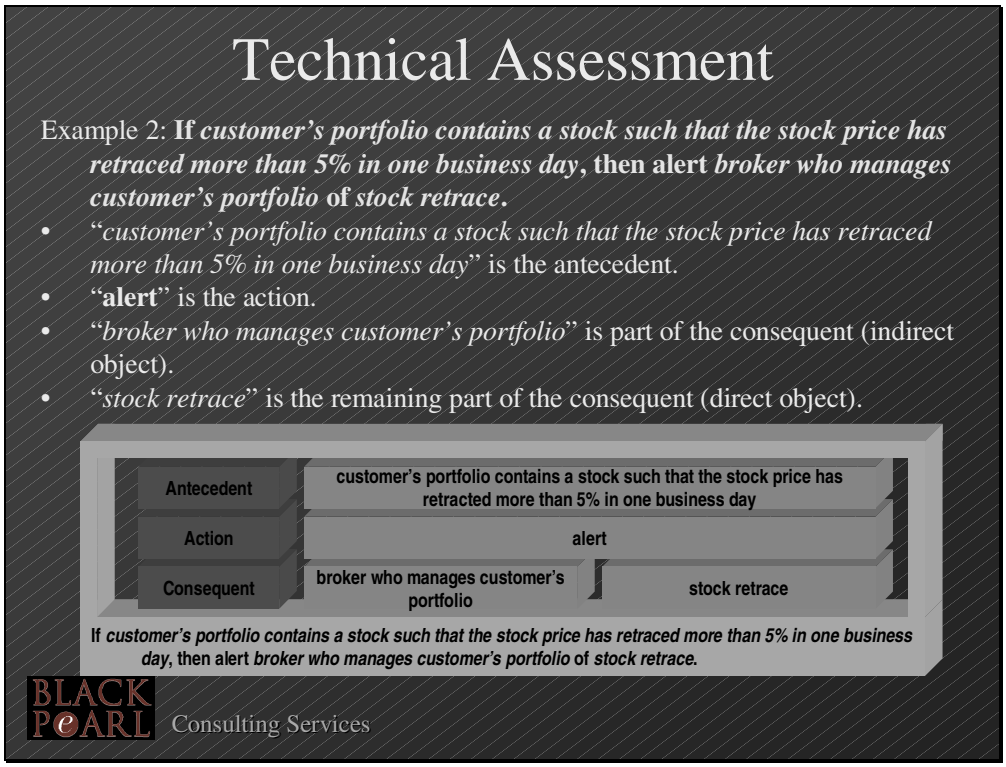
5. Technical Assessment



6. Technical Assessment



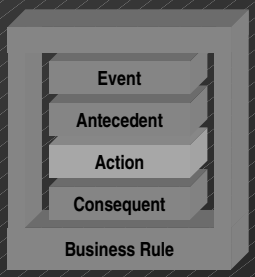
7. Technical Assessment



8. Technical Assessment

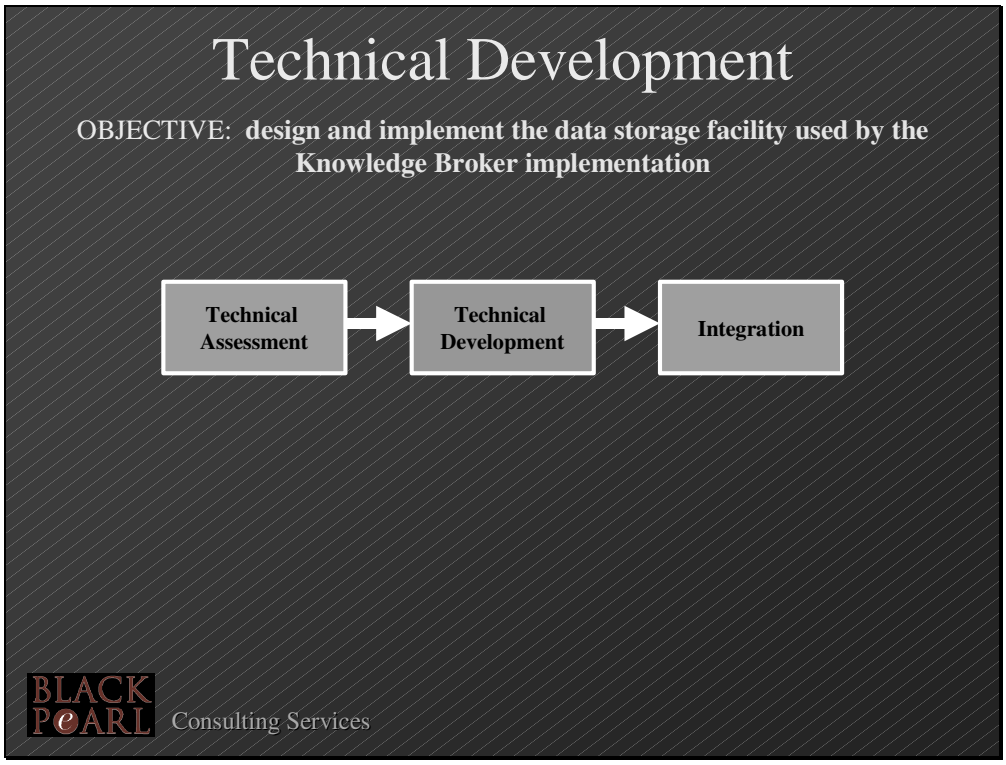
Technical Assessment

- The **action** is the most constrained part of the business rule.
- Different actions can be defined in and processed by Knowledge Broker.
- Knowledge Broker v.1 only supports actions that require **implicit** indirect objects, as shown in Example 1.
- Actions with **explicit** indirect objects require Knowledge Broker v. 1 enhancements, as shown in Example 2. These enhancements are handled by Black Pearl Product Development and Engineering.

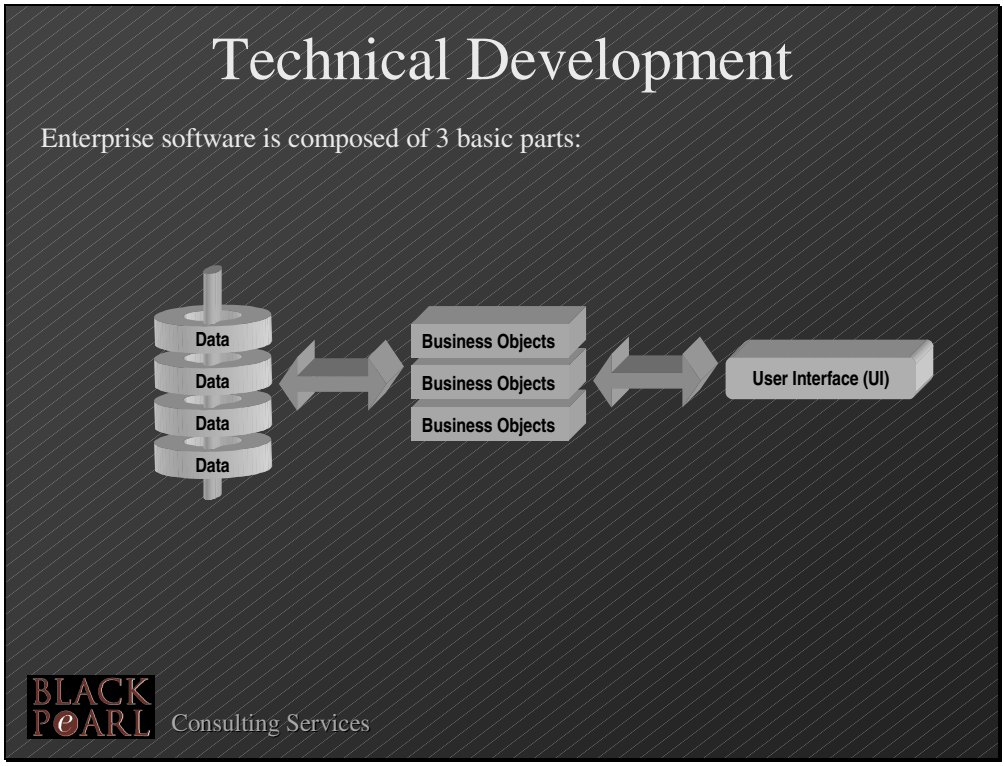


BLACK PEARL Consulting Services

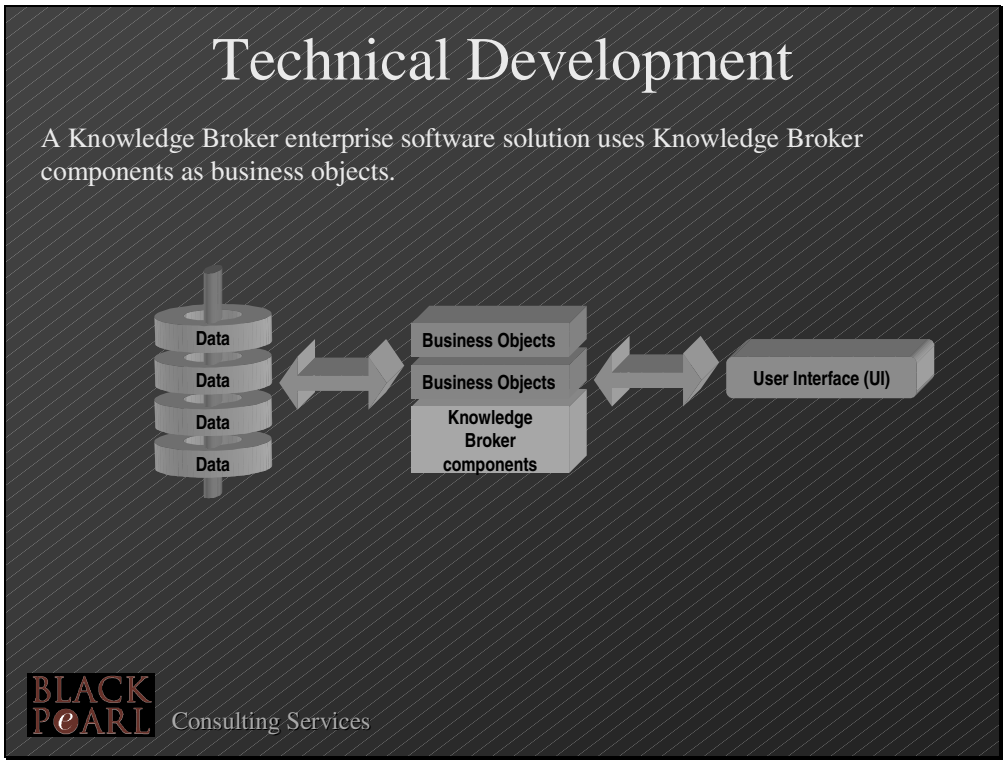
9. Technical Development



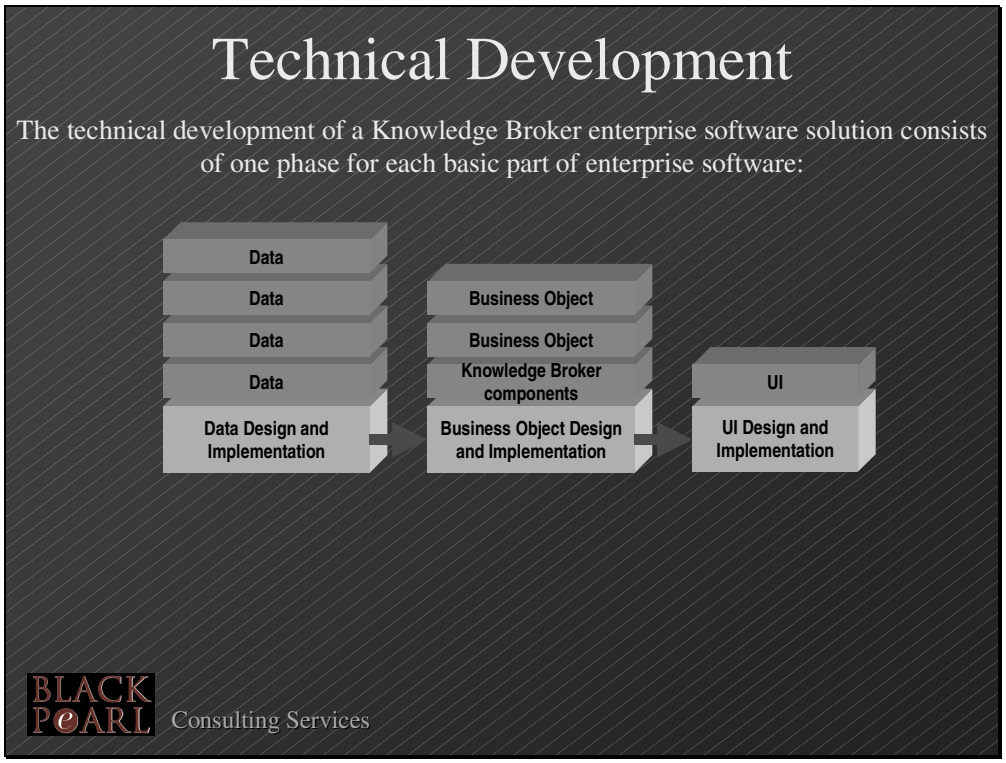
10. Technical Development



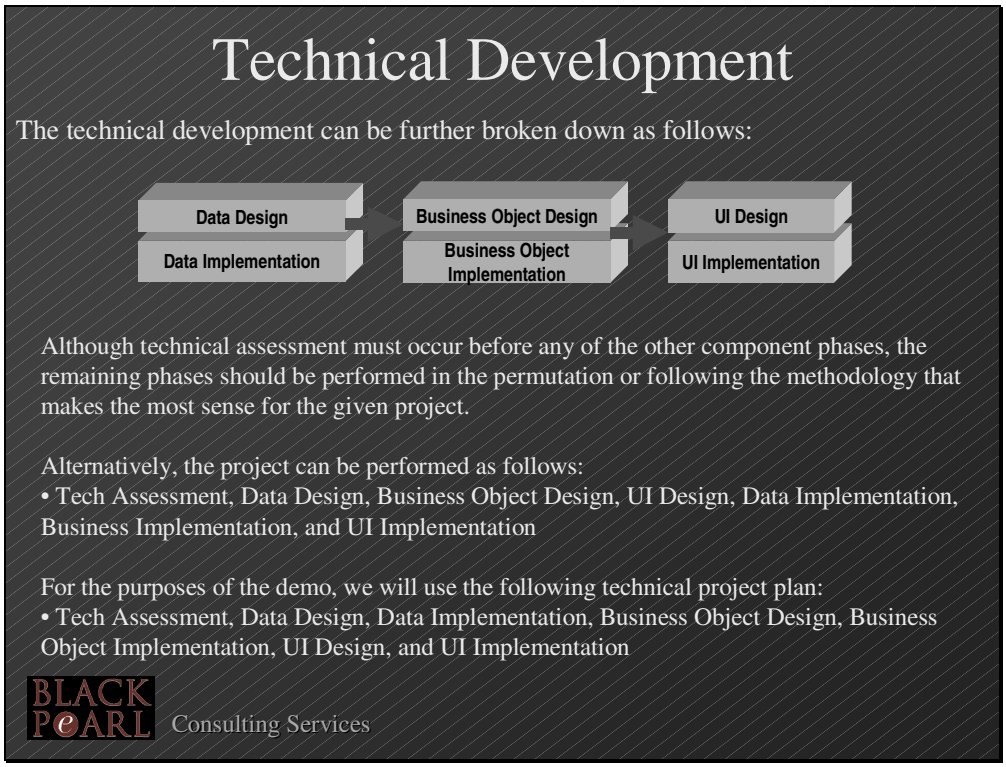
11. Technical Development



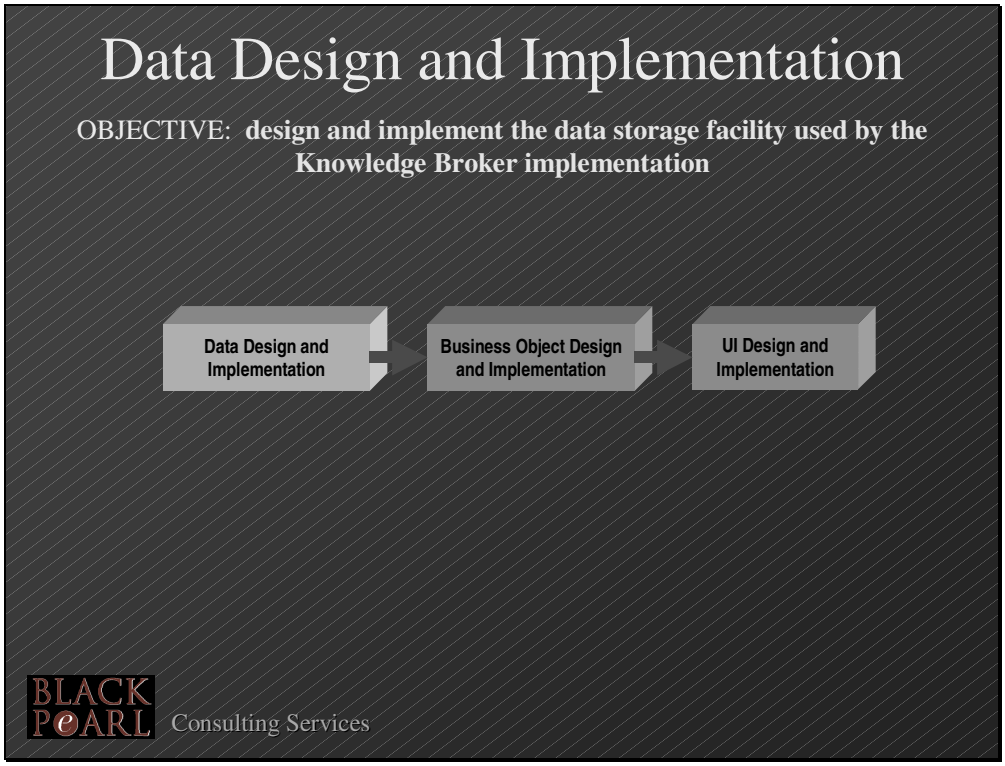
12. Technical Development



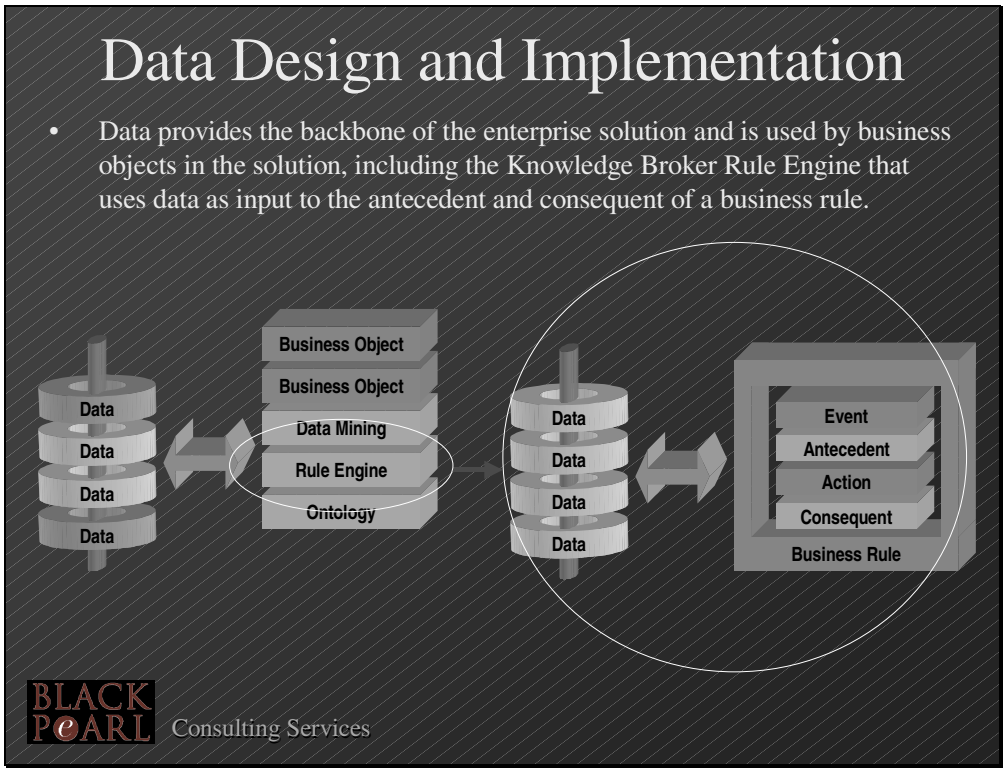
13. Technical Development



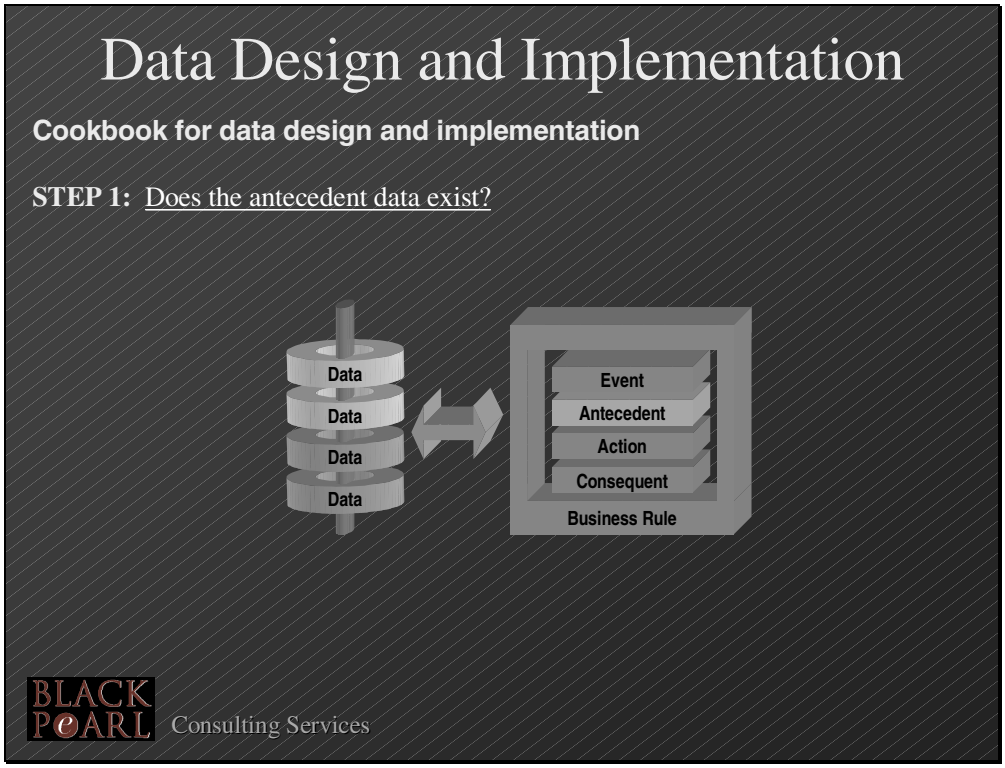
14. Data Design and Implementation



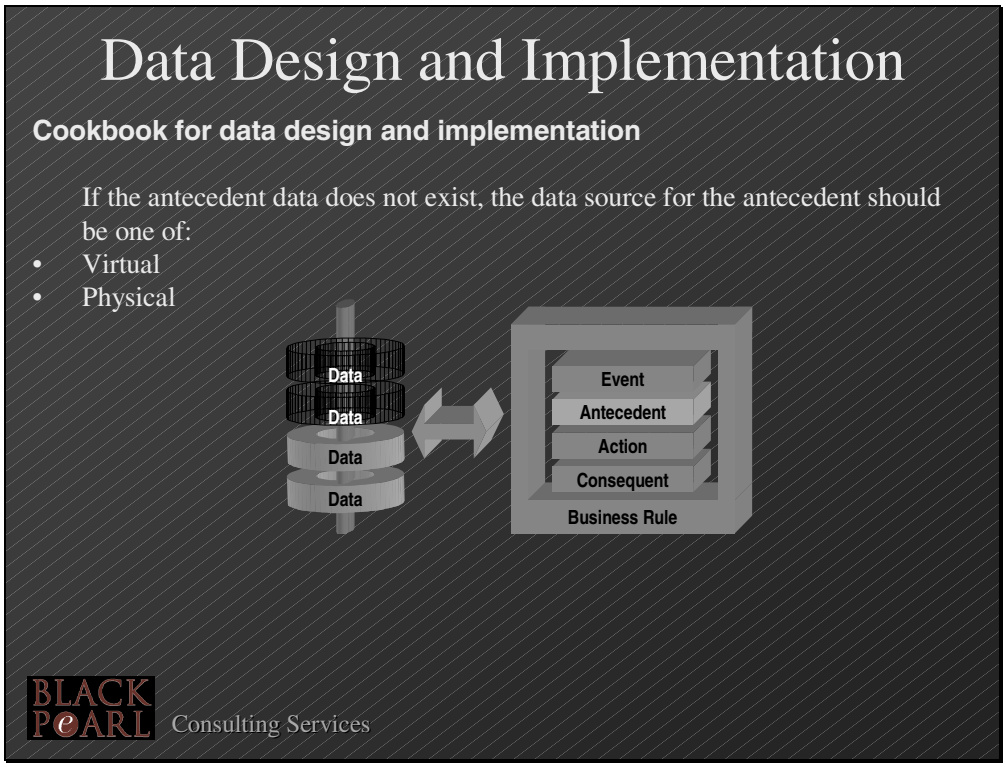
15. Data Design and Implementation



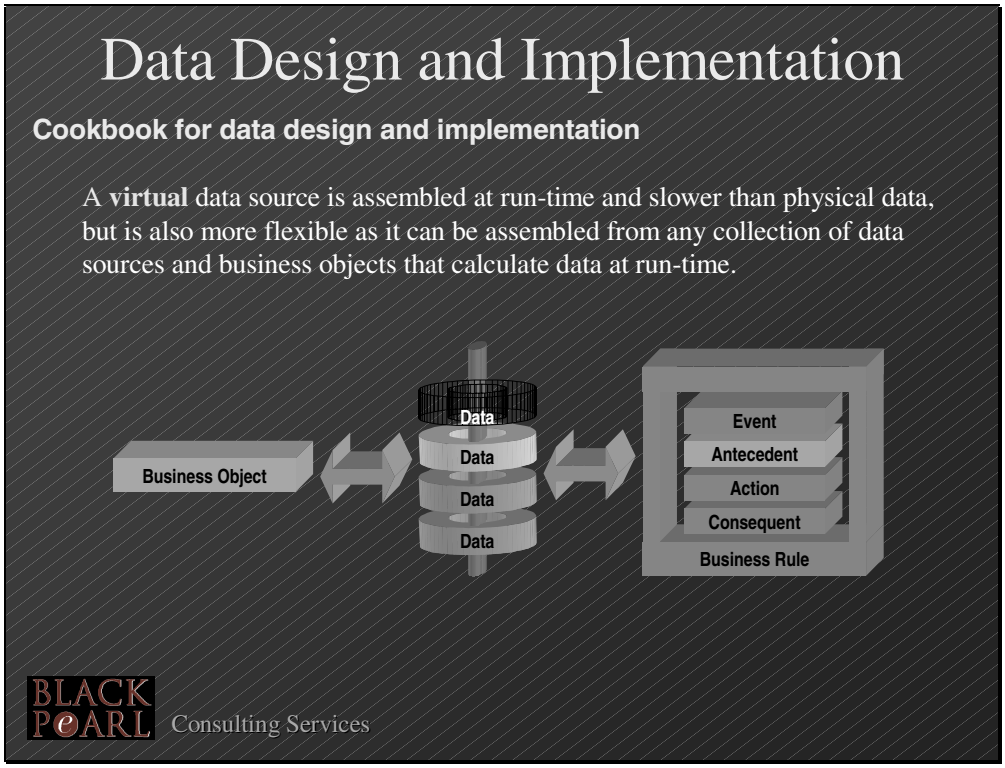
16. Data Design and Implementation



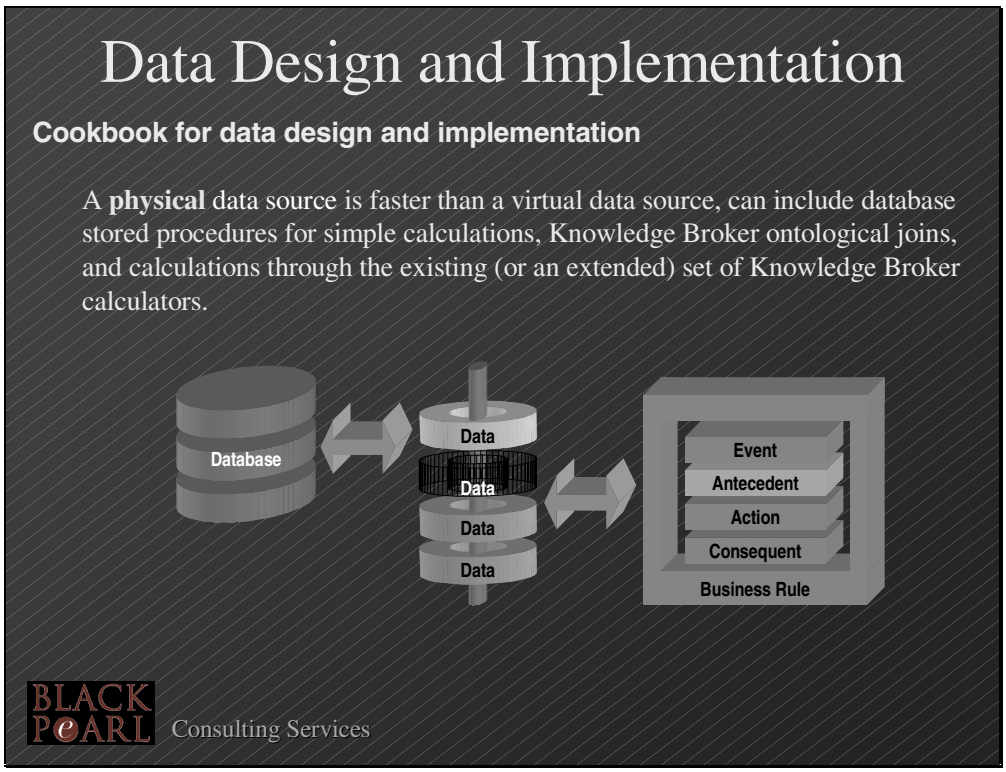
17. Data Design and Implementation



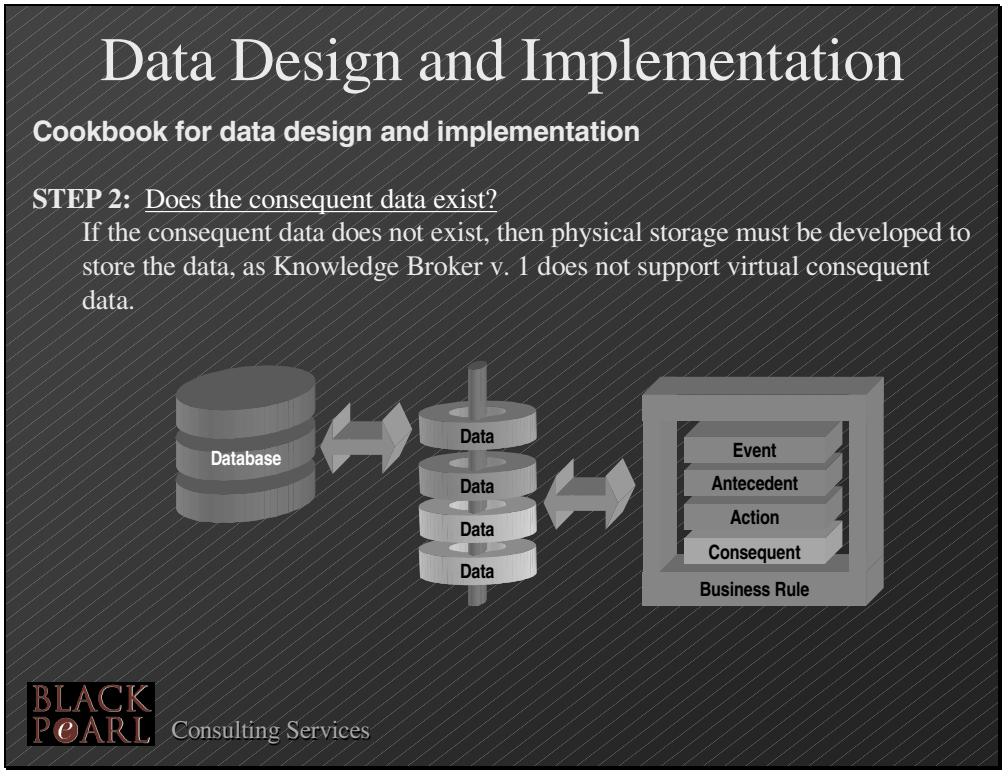
18. Data Design and Implementation



19. Data Design and Implementation

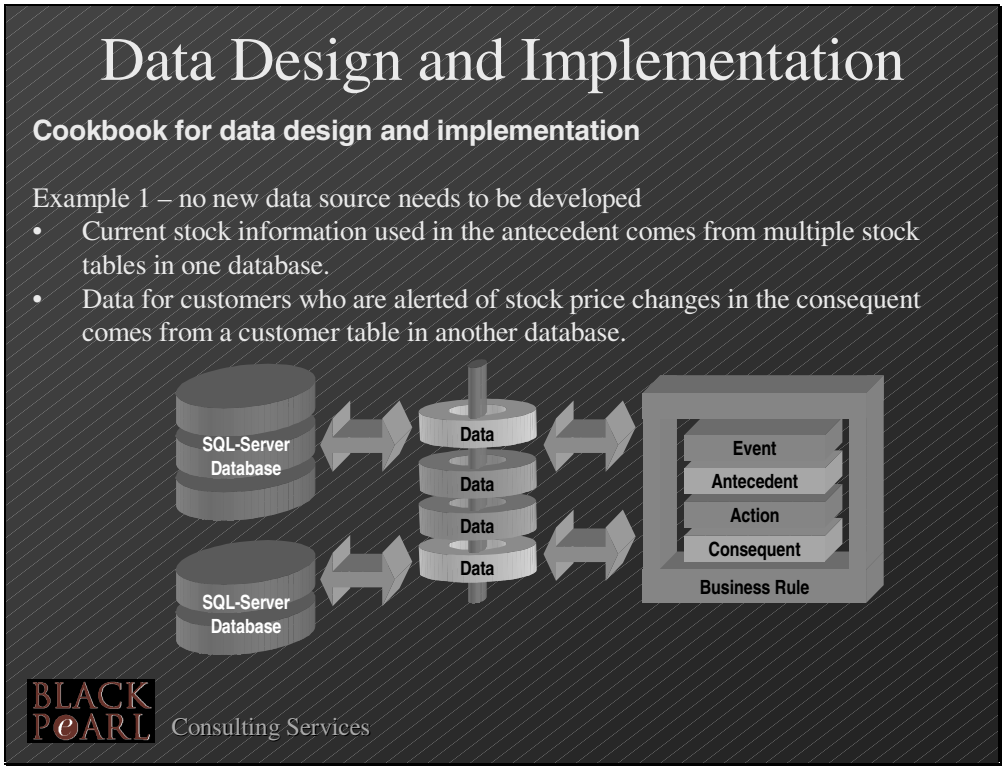


20. Data Design and Implementation

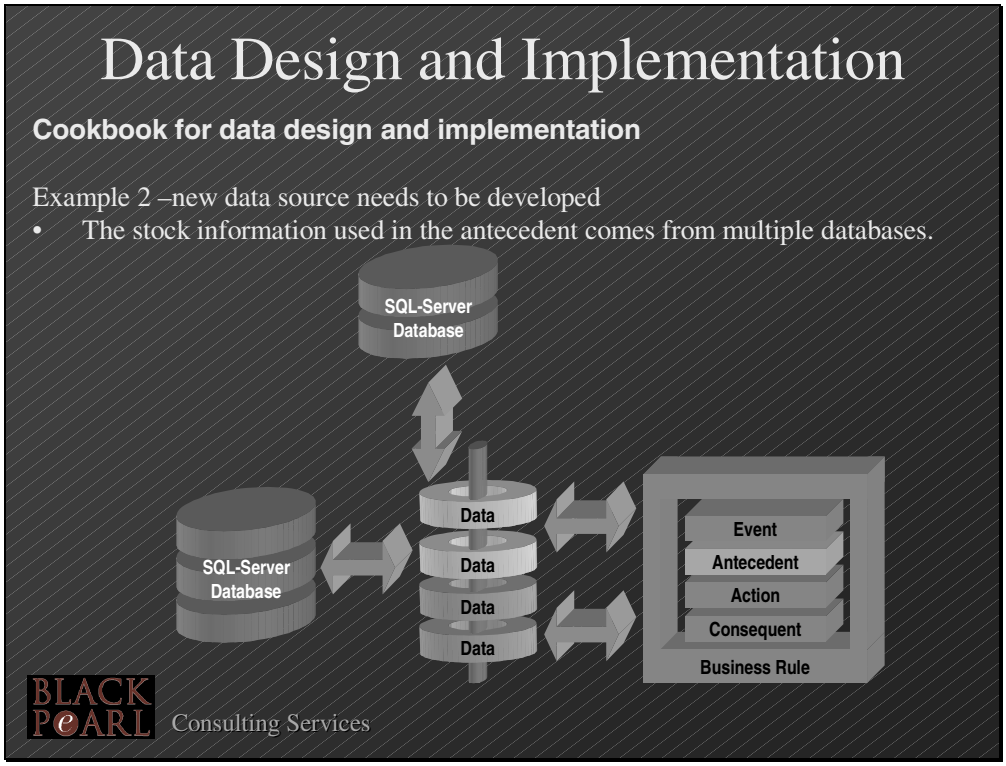


21. Data Design and Implementation

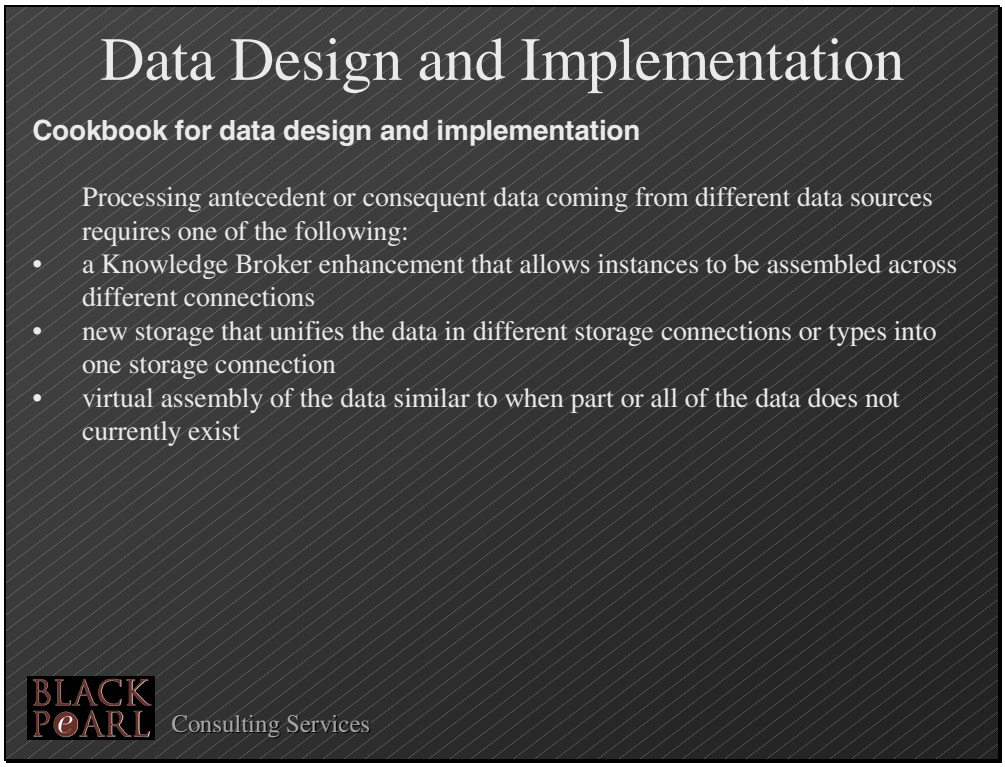
22. Data Design and Implementation



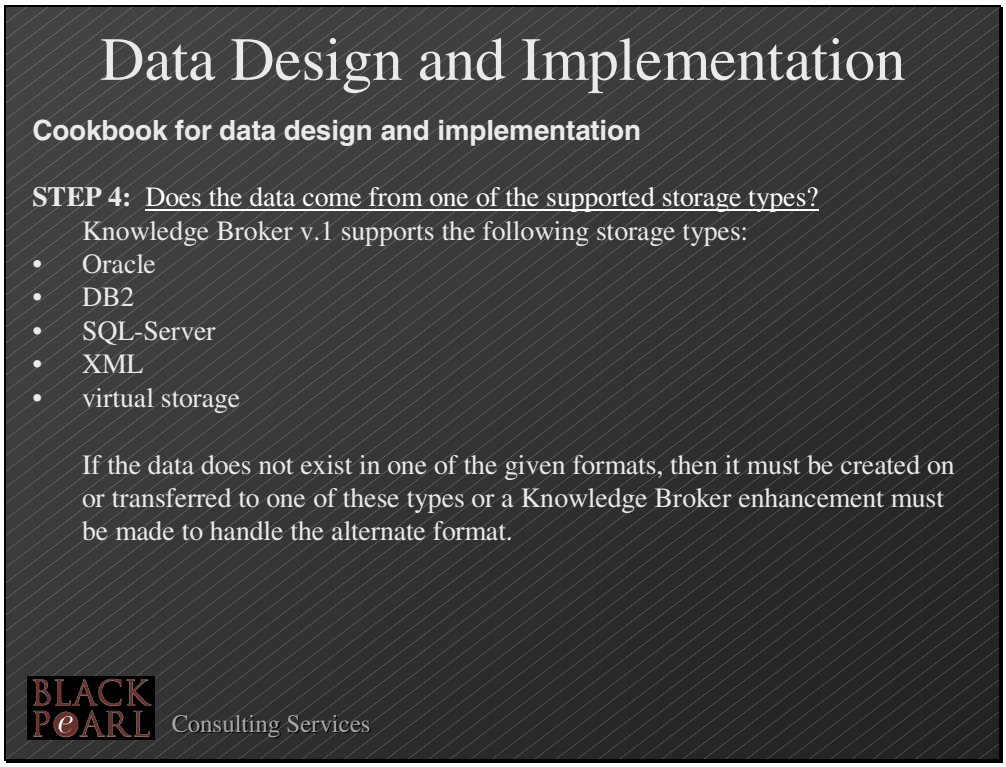
23. Data Design and Implementation



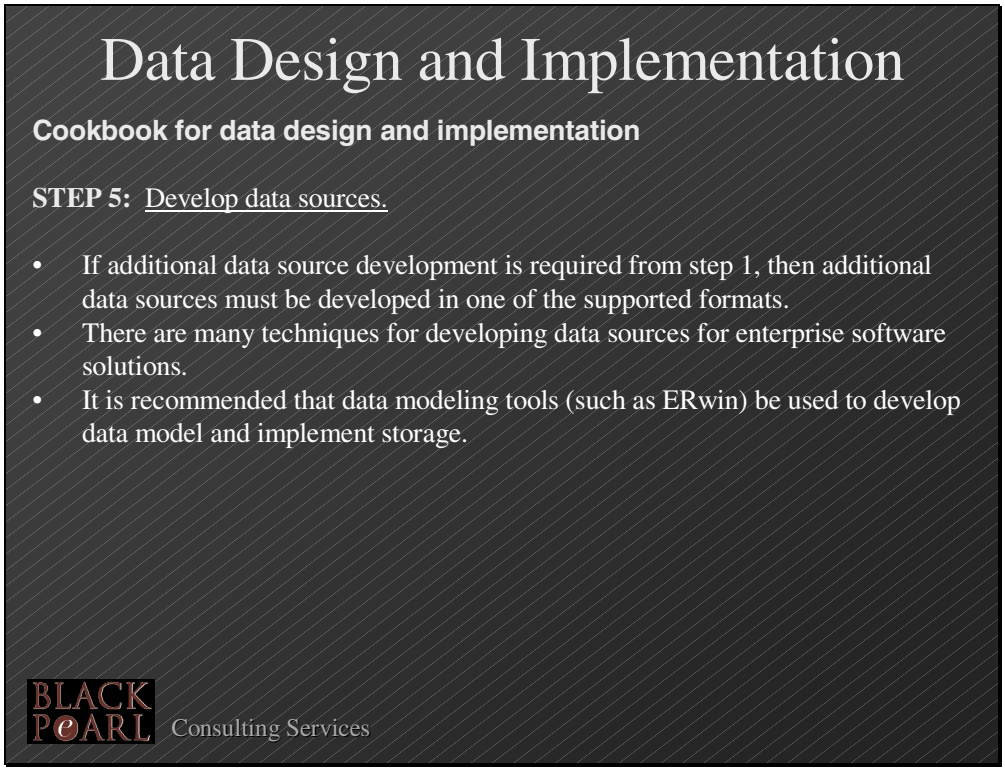
24. Data Design and Implementation



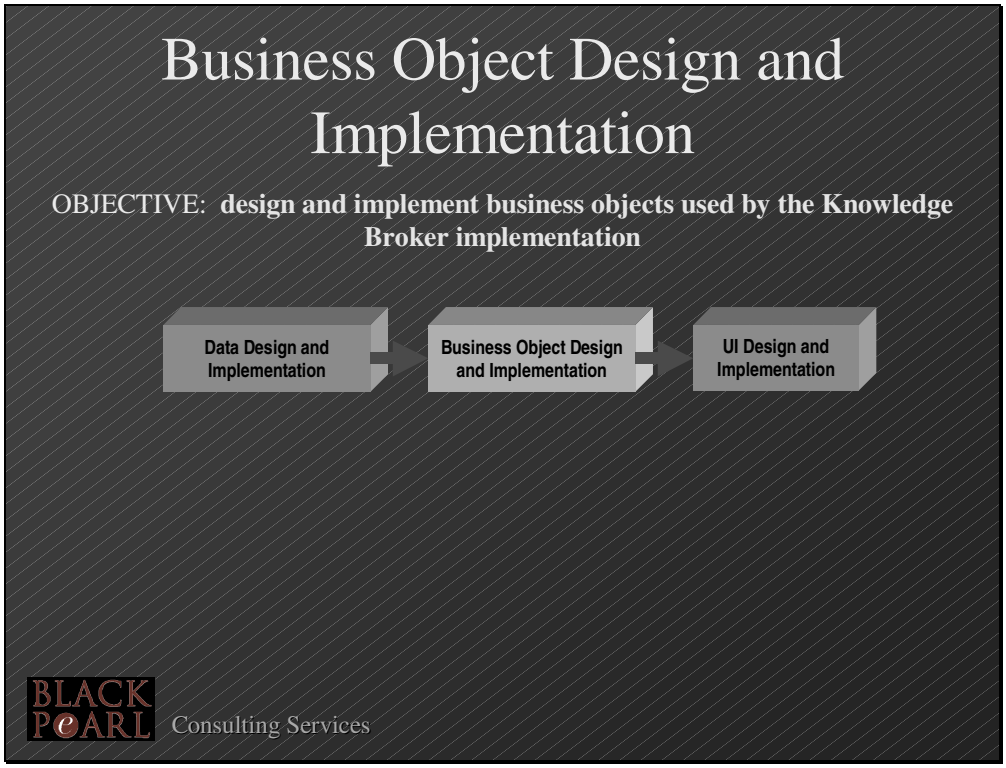
25. Data Design and Implementation



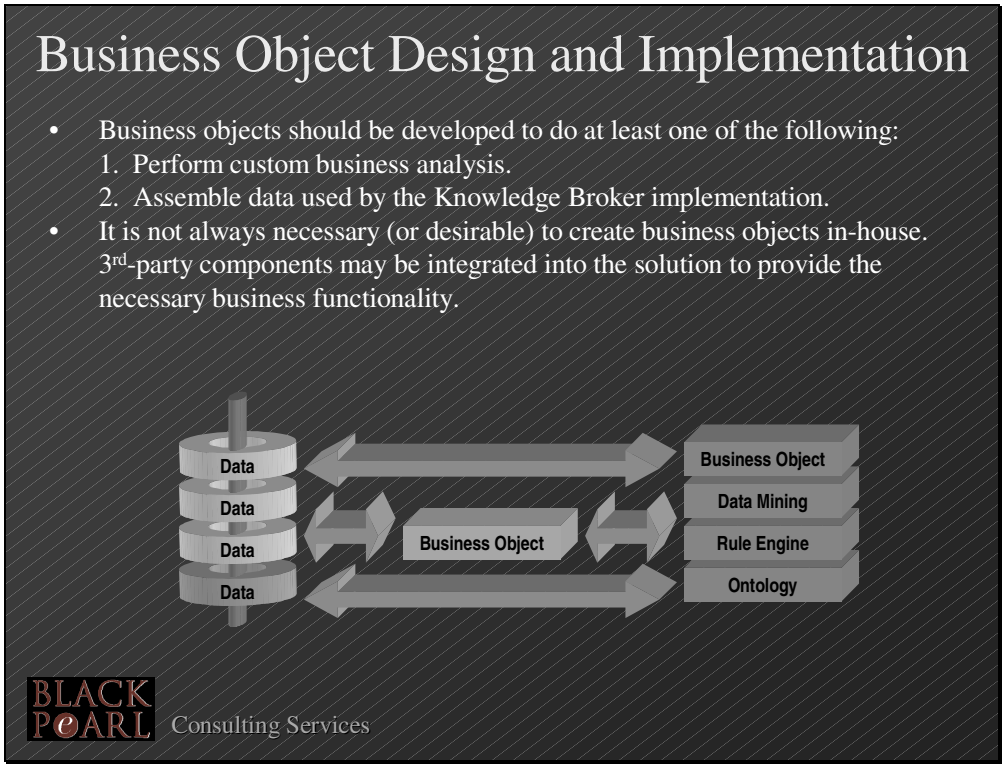
26. Data Design and Implementation



27. Business Object Design and Implementation



28. Business Object Design and Implementation



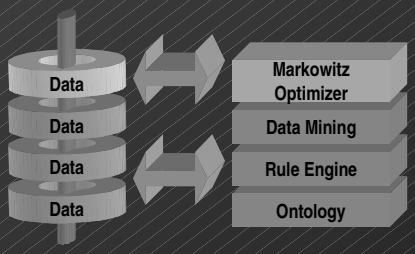
29. Business Object Design and Implementation

Business Object Design and Implementation

Cookbook for business object design and implementation

STEP 1: Is custom business analysis required for the solution?

Custom business analysis that lie outside the scope of Knowledge Broker require custom object development.



BLACK PEARL Consulting Services

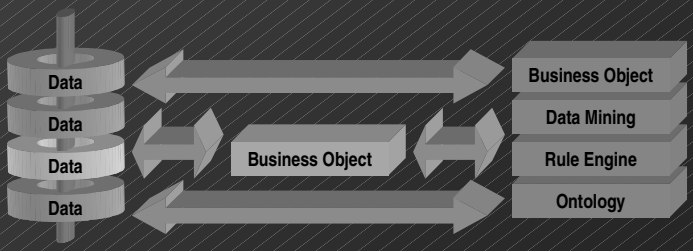
30. Business Object Design and Implementation

Business Object Design and Implementation

Cookbook for business object design and implementation

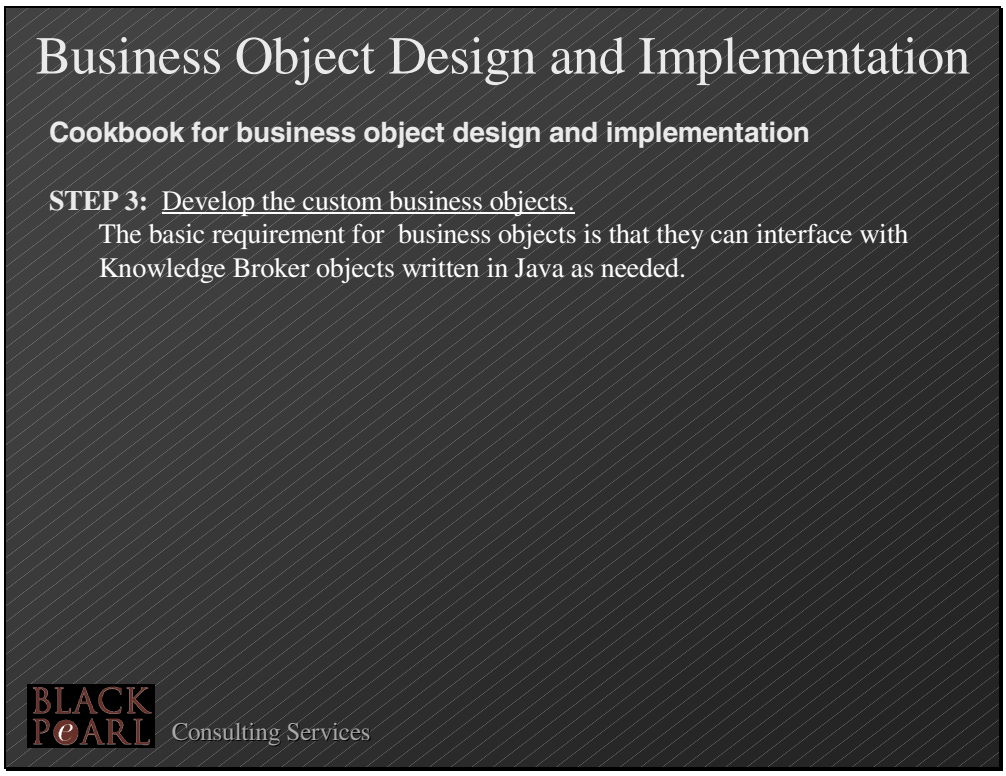
STEP 2: Does the Knowledge Broker implementation need to maintain data used as input into the Knowledge Broker?

Business objects that maintain data used exclusively as input to the Knowledge Broker must be developed, such as EJBs (Enterprise JavaBeans) that manage data traffic to and from a database.

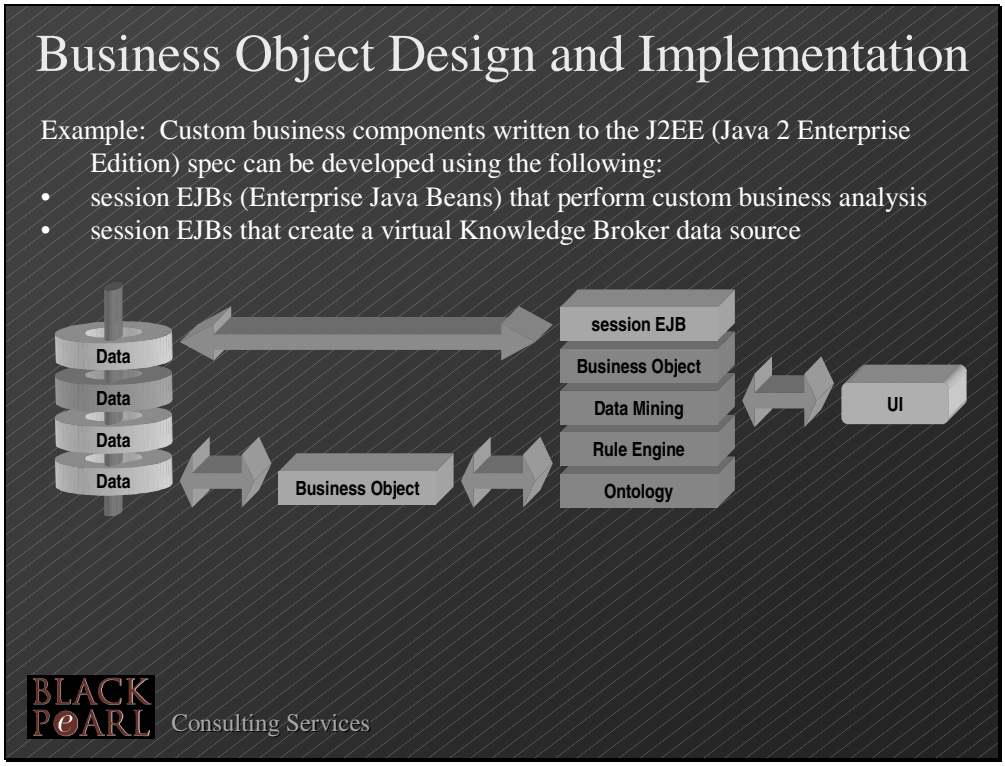


BLACK PEARL Consulting Services

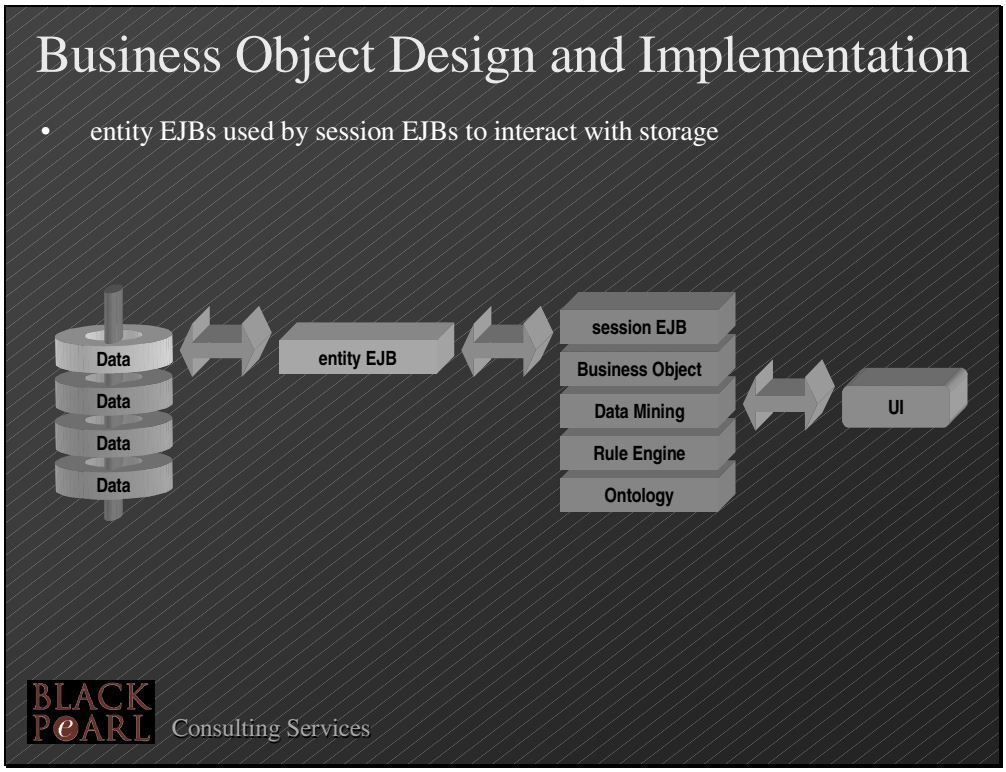
31. Business Object Design and Implementation



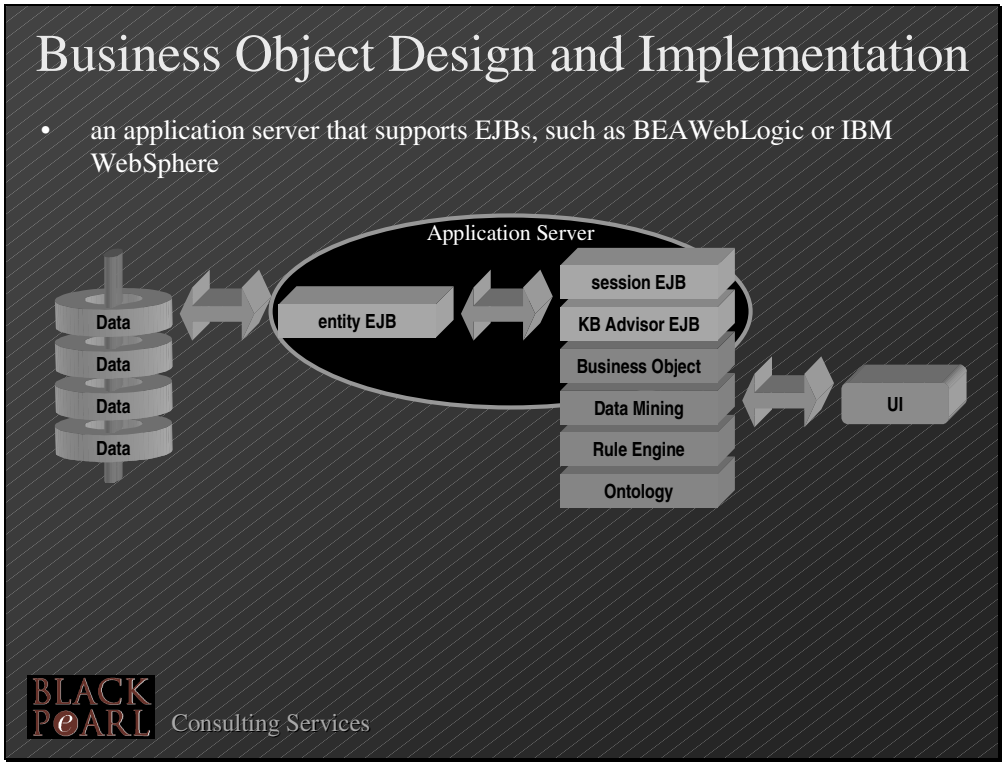
32. Business Object Design and Implementation



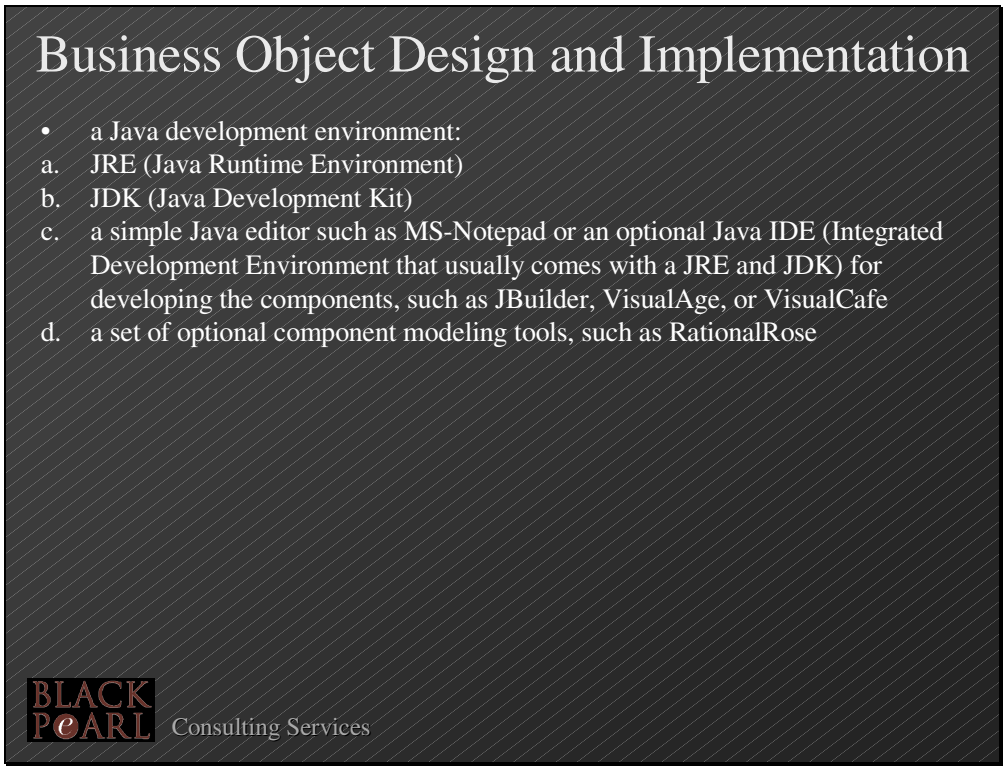
33. Business Object Design and Implementation



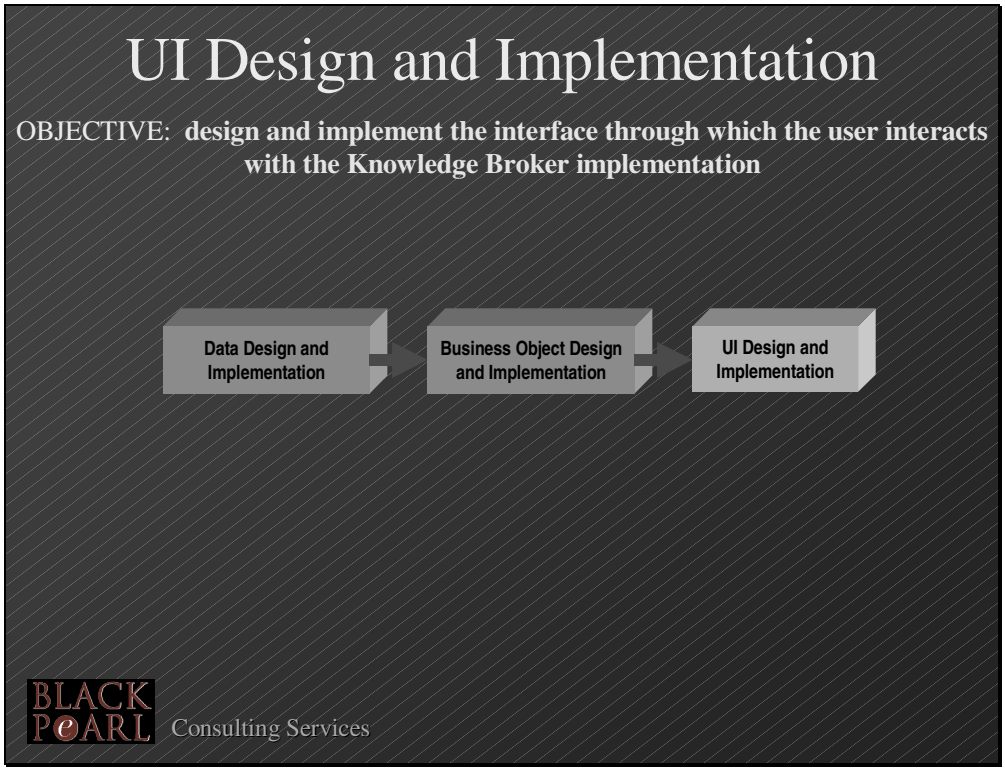
34. Business Object Design and Implementation



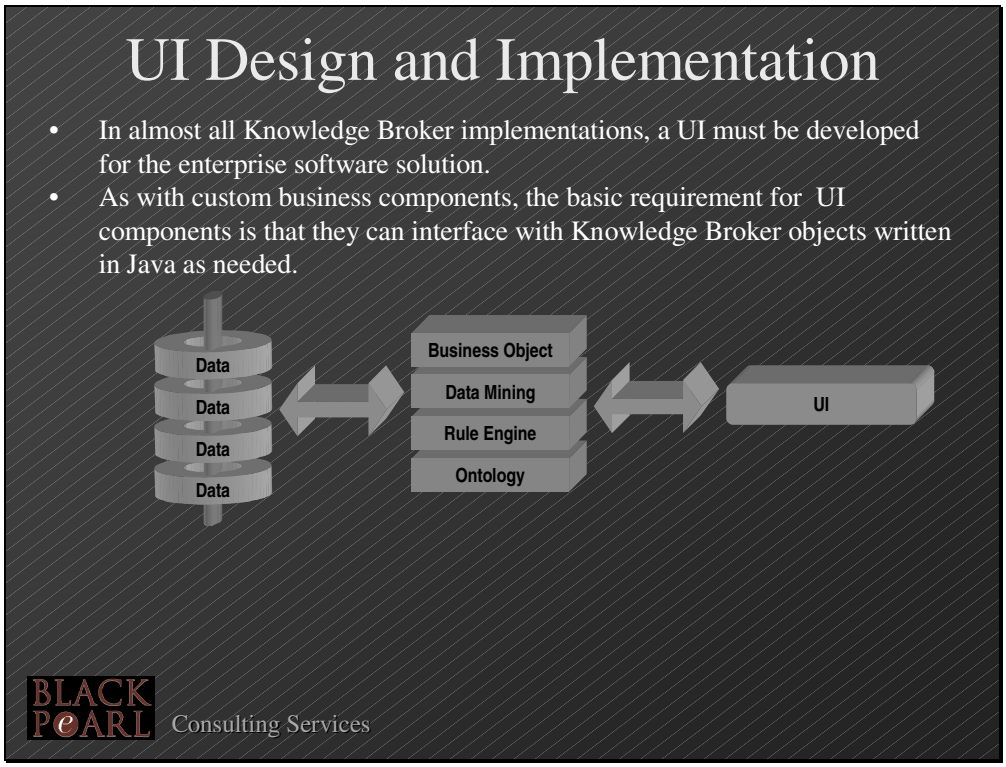
35. Business Object Design and Implementation



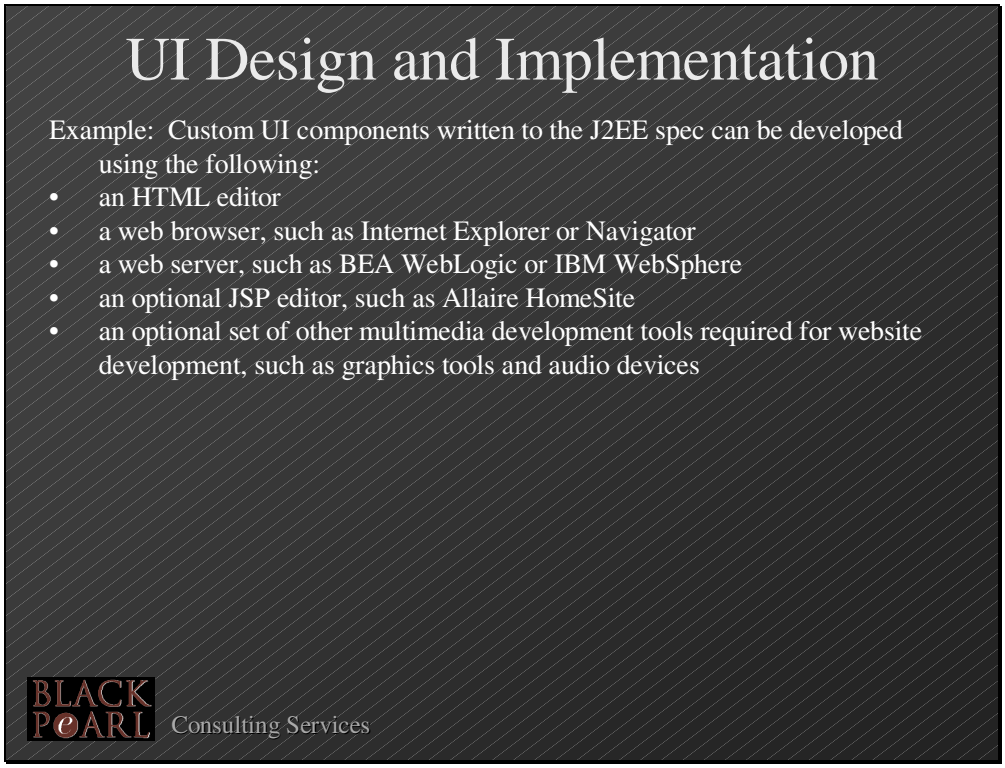
36. UI Design and Implementation



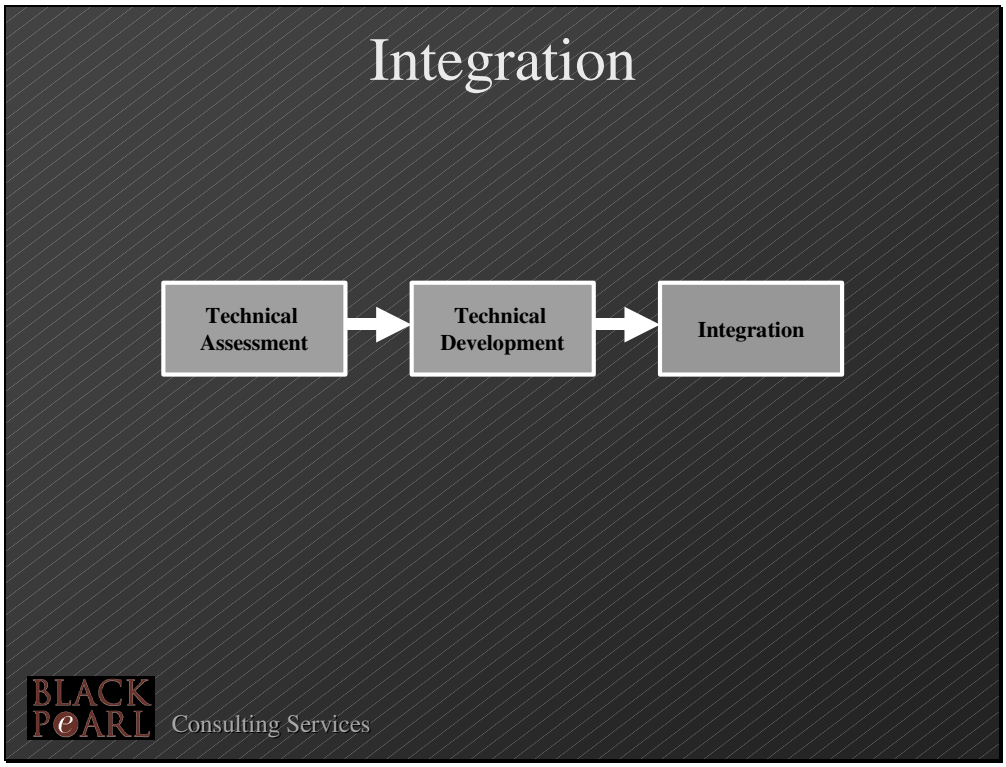
37. UI Design and Implementation



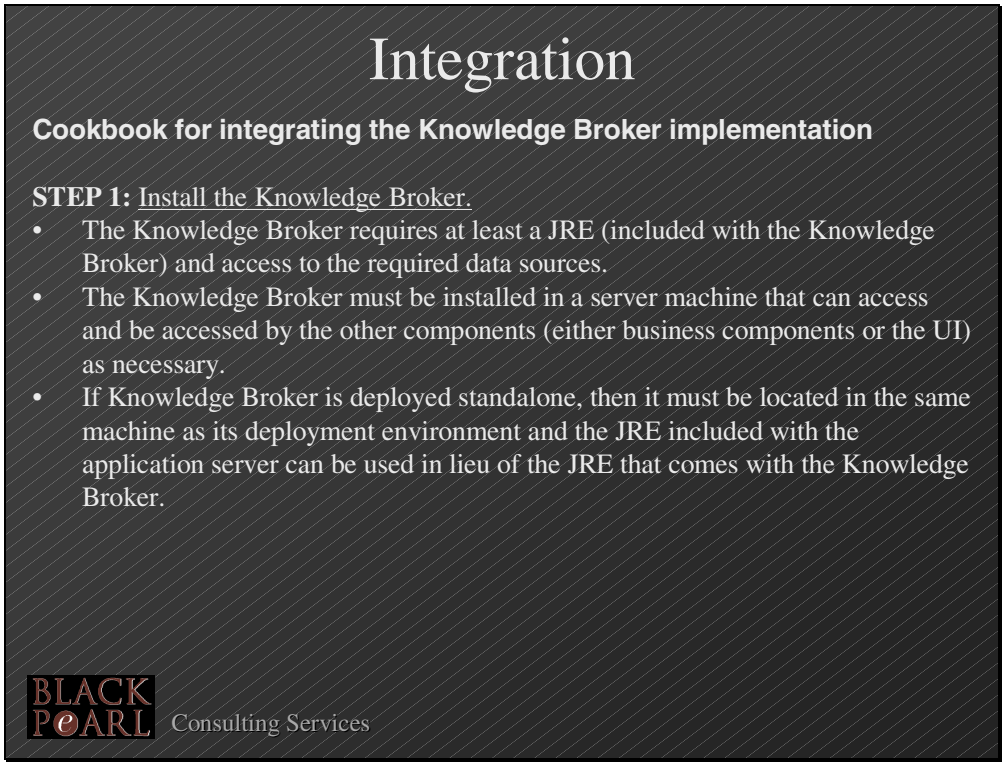
38. UI Design and Implementation



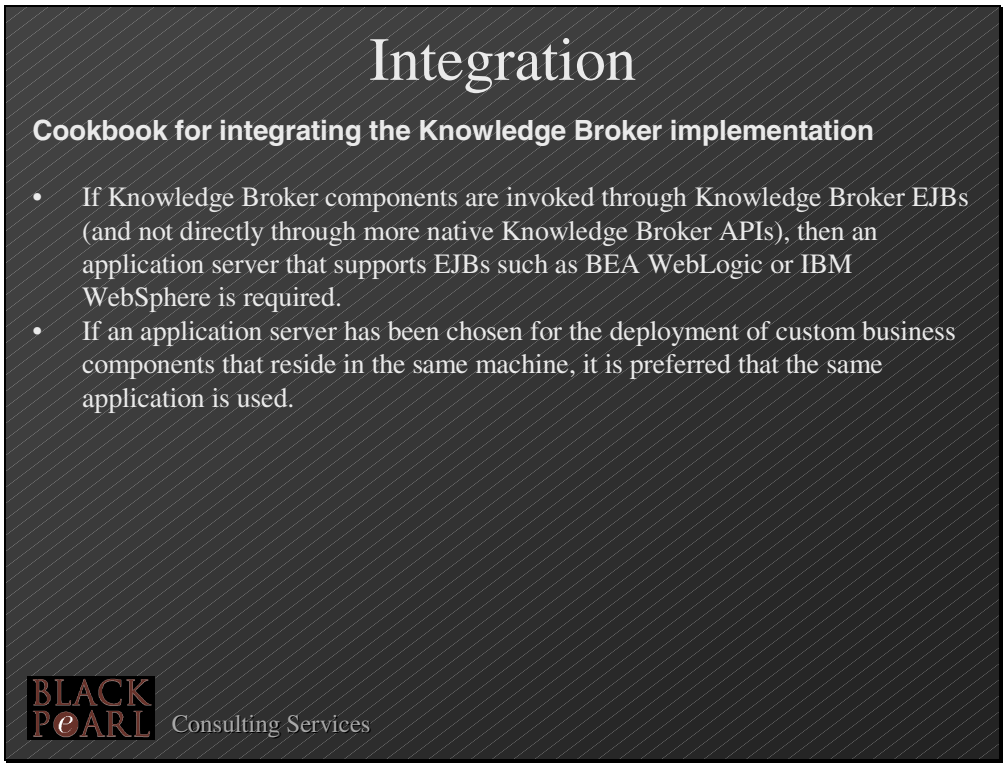
39. Integration



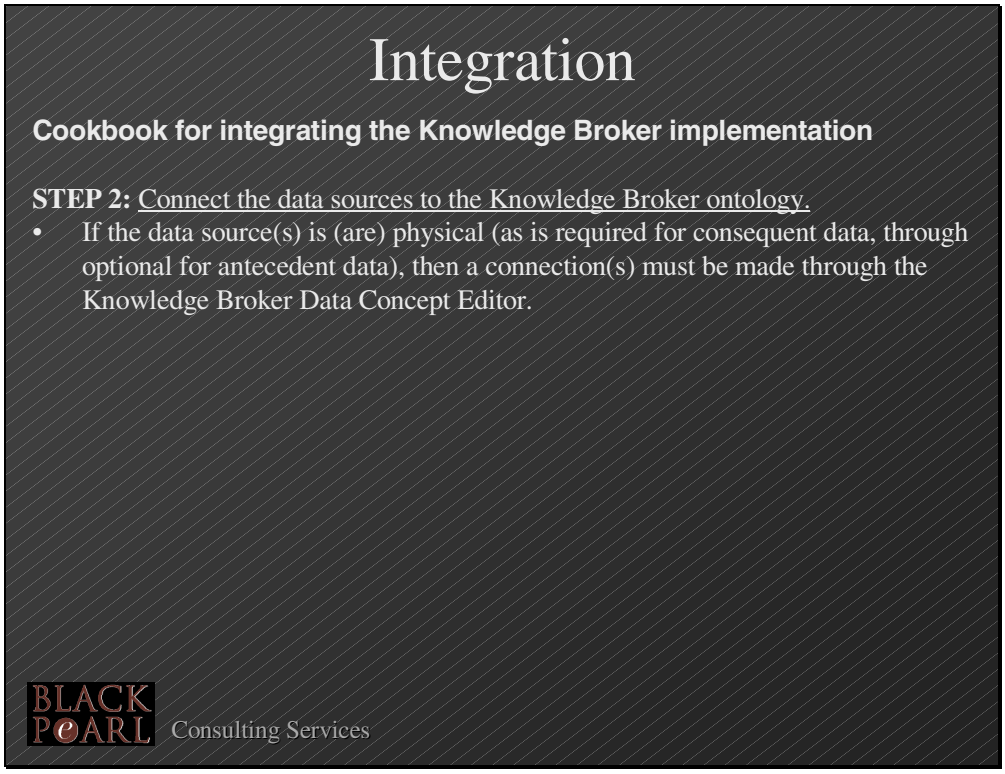
40. Integration



41. Integration

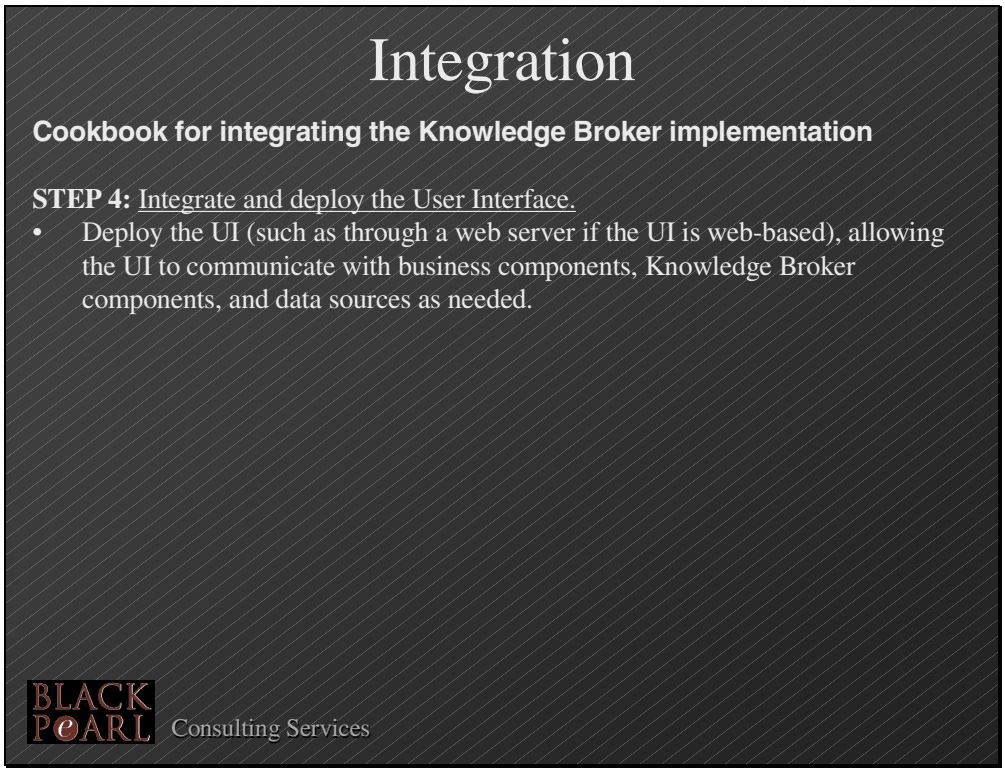


42. Integration

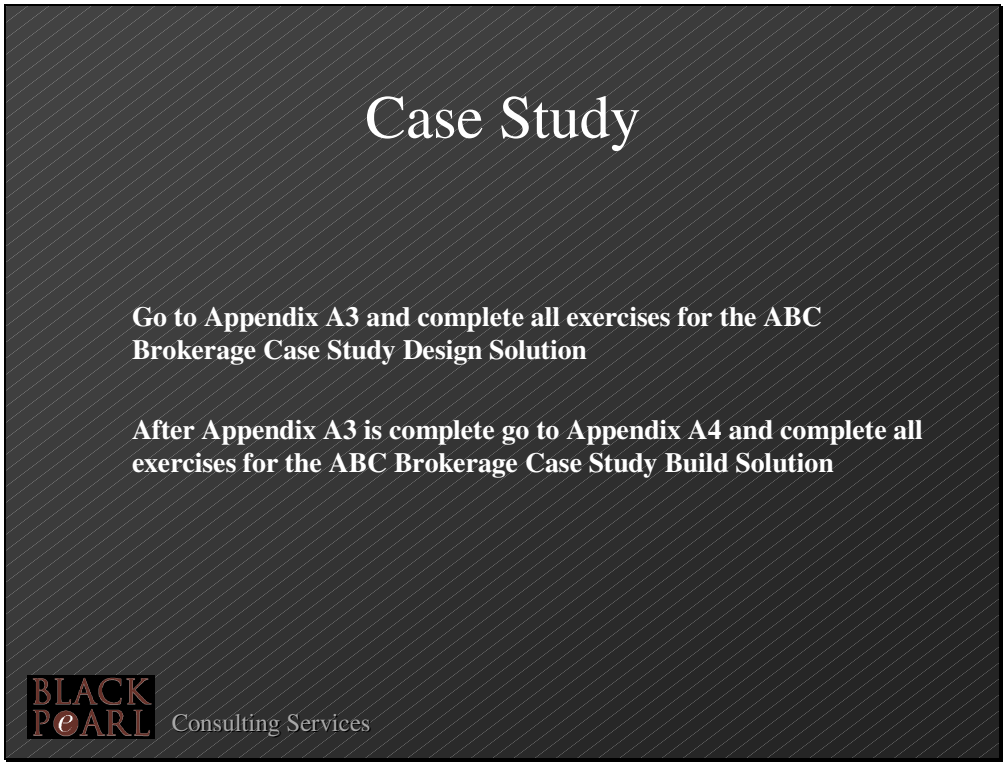


43. Integration

44. Integration



45. Case Study



Appendices A: Case Study

Black Pearl Software, Inc.
November 2, 2000

Note to participant

The names, characters, places, and incidents in this workbook are fictitious. Any resemblance to actual businesses, products, persons, events or locales is entirely coincidental.

Your organization has signed a contract and non-disclosure entitling your participation in this training program. This entitles you to participate in this training and to use and retain these materials. Neither this program nor any part thereof may be reused or reproduced in any form or medium now known or hereafter invented or created without the express written permission of Black Pearl, Inc.

BLACK PEARL

Appendix A1: ABC Brokerage Case Study

Knowledge Broker Proposal and Positioning

Black Pearl Software, Inc.
November 2, 2000

Note to participant

The names, characters, places, and incidents in this workbook are fictitious. Any resemblance to actual businesses, products, persons, events or locales is entirely coincidental.

Your organization has signed a contract and non-disclosure entitling your participation in this training program. This entitles you to participate in this training and to use and retain these materials. Neither this program nor any part thereof may be reused or reproduced in any form or medium now known or hereafter invented or created without the express written permission of Black Pearl, Inc.

000X

Contents

1. Selling The Knowledge Broker

1.1 ABC Brokerage History

ABC Brokerage has a long history of offering informative and accurate financial management advice to the investment community. John Albert and Douglas Belmont first introduced themselves to investors in 1953 when they published the *Investing Advisor*, a weekly investment newsletter aimed at middle to advanced investors. At its height in 1958, the weekly newsletter had over 2000 subscribers, each paying $50 per year for their valuable financial recommendations and insights into the financial markets.

In 1959, Albert and Belmont established the AlbertBelmont Company, a traditional brokerage house that provided clients with additional financial services as well as advice and insight. Within 10 years, AlbertBelmont had grown to serve almost 30,000 clients in the Northeastern United States.

The decades of the 60's, 70's, and 80's saw AlbertBelmont expand their client base and add branch offices with investment experts from all over the Eastern portion of the United States. Real growth, however, for AlbertBelmont, did not occur until the early 1980's when they merged with their rival investment company, Campbell Investment Management, and became known as ABC Brokerage.

In the early 1990s, ABC Brokerage was positioning itself for further growth. In 1992, ABC Brokerage broadened its service offerings to satisfy the needs of the growing potential market place, which was primarily dominated by novice investors. The program proved to be a huge success. Assets had grown from less than $1 billion from the inception of the program to over $100 billion by the year 2000.

1.2 ABC Brokerage Service Offerings

ABC Brokerage is an established investment bank that offers a variety of asset management services and advice to its private clients. The following three levels of service are offered to its clients:

Silver: A trade execution only service, which allows the client to place instructions for the acquisition or disposal of assets by phoning a broker at the bank.

Gold: An financial advisory service, which provides clients an opportunity to discuss their portfolio with an investment manager who will make recommendations as to the appropriate risk weighting and asset selection.

Platinum: A fully managed service, providing day-to-day tracking and trading advice based on the stated investment aims of the client.

1.3 ABC Brokerage Problem/Opportunity

ABC Brokerage is looking to capitalize on the recent increased interest in investing. They are looking to aggressively grow the amount of assets under management by increasing the number of new clients. They feel that their company is well positioned and that new investors would benefit greatly from the advice and services they could provide. ABC Brokerage's current concern is that in order to continue providing the high value advice and services that their clients demand, ABC Brokerage will need to either increase the number of investment managers employed by the bank, or increase the number of accounts each investment manager is in charge of.

1.4 ABC Brokerage Solution

ABC Brokerage is interested in capturing some of the expertise of the investment managers in a software solution. The solution would have to ensure that much of the filtering of possible investment advice has been performed automatically with the investment manager only having to process a small number of options. A solution such as this would not only enable each investment manager to process more accounts, but would also allow the role to be performed by managers with less experience.

1.5 Case Study Exercises

Exercise 1: Create a Sales Presentation

Create a sales presentation for ABC Brokerage that includes the following points:

- Why Knowledge Broker is an appropriate fit for the issues that ABC Brokerage is currently facing.
- A high level description of a Knowledge Broker solution that addresses the issues facing ABC Brokerage.

Exercise 2: Present to ABC Brokerage

Present the sales presentation produced in Exercise 1 to the Vice President of Equity Markets at ABC Brokerage.

Appendix A2: ABC Brokerage Case Study

Assessment and Planning

Black Pearl Software, Inc.
November 2, 2000

Note to participant

The names, characters, places, and incidents in this workbook are fictitious. Any resemblance to actual businesses, products, persons, events or locales is entirely coincidental.

Your organization has signed a contract and non-disclosure entitling your participation in this training program. This entitles you to participate in this training and to use and retain these materials. Neither this program nor any part thereof may be reused or reproduced in any form or medium now known or hereafter invented or created without the express written permission of Black Pearl, Inc.

000X

Contents

1. Assessment and Planning

1.1 Transition from Sales to Solution Delivery

Congratulations, ABC Brokerage has accepted your proposal to implement the Knowledge Broker as part of the overall solution that will deliver portfolio recommendations to the investment manager using the bank's intranet as the delivery mechanism. They are willing to free up the necessary resources and would like to start the project immediately.

1.2 ABC Brokerage Interviews

Project Sponsor
Thomas Smith
Senior Vice President of Equity Markets
Here at ABC Brokerage, one of the foundations of our company is to provide the appropriate advice and money management in accordance with our clients profile and risk tolerance levels. We are successful because of the valuable advice and personalized attention that we provide our clients, not because we are able to churn through millions of transactions in the blink of an eye. We are not a discount brokerage. Our Investment Managers spend the majority of their day ensuring that our client's are aware of all of the valuable investing opportunities available to them, according to the investment risk levels that our clients designate that they were comfortable with.

Every morning I create a list of what I call "Hot Stocks:" These are stocks that our institution and research have identified as being good opportunities for our clients. Our Investment Managers then review this list and identify which opportunities would be appropriate for the clients they manage, based on their clients risk tolerance and profile. If our Investment Managers feel that one of their clients may be interested in one of the opportunities that we have proposed, they will call them up to discuss these with them. The knowledge and expertise that our Investment Managers bring to our clients is what differentiates us from our competition.

Project Expert Resource
Harold Williams
Investment Manager of Equity Markets
As an Investment Manager, I am responsible for ensuring that my clients are informed with the most promising and up to date opportunities available to them in the market that meet their risk tolerance and personal profile. Every

day I receive a list of "Hot Stocks," which our research group has determined are good opportunities. I review the list and try to identify if any of my clients would be interested in these stocks based on their risk tolerance and profile information. Each client's preferences and situations are different. Hence, I typically spend a lot of time requesting profile and product queries from the Information Services department to ensure I have all of the relevant information necessary to make the right decision. Since all of the Investment Managers are making similar requests, it usually takes a couple of hours for them to get back to me. Once I am sent the results of the queries, I analyze the information, and based on my experience and knowledge of the type of client and the current market information I have available, I decide whether this opportunity is worth contacting the client. If it appears to be a good fit, I call the client to approve the transaction. After I have walked through their potential options and they feel comfortable with the recommendations, they generally tell me to go ahead and place the order. I then send all of the stock transactions to the Order Execution department. The department responsible for deciding how to time, group, and execute each order to ensure our clients get the best possible prices.

Project Technical Resource
Janet Jones
Chief Information Officer
We start off each morning by flagging all of Thomas' "Hot Stocks" in the Product database. Once all of the stocks have been flagged, we start the Hot Stock Reporter. This program queries the Product database and creates a report of all "Hot Stocks" of the day, which are then sent directly to each Investment Manager. Typically, the "Hot Stocks" takes only about 15 minutes at the beginning of our day. The majority of our time is actually spent running queries for the Investment Managers. All of the requested information from the Investment Managers' reports is based on the information in the Profile database and the Product database. Most of the information in the Profile database is from the Investment Profile. All of our clients fill out an Investment Profile when they open an account. The customer service reps process all of the applications and enter the Investment Profile information into the system for all of our clients. Our customer service reps are usually accurate and most of the data is fairly good. The Product database is updated once a day from information we receive from an outside vendor. The data in the Product database is always clean. We've never had any issues with that information.

1.3 Case Study Exercises

Exercise 1: Define Context and Scope

Using the information above, document the context workflow diagram for the project. The context workflow should include the entire process, including inputs, decision points, and outputs. This can be done as a group on a whiteboard to expedite the process and allow for input from the entire group.

After the process has been white boarded, it can be entered into the modeling tool and the appropriate fit and scope for the Knowledge Broker can be identified.

Appendix A3: ABC Brokerage Case Study

Design Solution

Black Pearl Software, Inc.
November 2, 2000

Note to participant

The names, characters, places, and incidents in this workbook are fictitious. Any resemblance to actual businesses, products, persons, events or locales is entirely coincidental.

Your organization has signed a contract and non-disclosure entitling your participation in this training program. This entitles you to participate in this training and to use and retain these materials. Neither this program nor any part thereof may be reused or reproduced in any form or medium now known or hereafter invented or created without the express written permission of Black Pearl, Inc.

000X

Contents

1. Design Solution

1.1 ABC Brokerage Requirements Overview

The following information was gathered during requirements sessions with the Investment Managers at ABC Company. The requirements have been split into two sections:

- **1.2 ABC Brokerage Ontology/Rule Base Requirements** – The information in this section will be necessary to complete the Knowledge Engineering portion of the project. This section contains both existing ABC Brokerage documentation and information gathered through requirements workshops with the client.
- **1.3 ABC Brokerage Technical Solution Requirements** – The information in this section will be necessary to complete the Technical Solution portion of the project. All of the requirements in this section were gathered through requirements workshops with the client.

1.2 ABC Brokerage Ontology/Rule Base Requirements

Buy recommendations will only be made to private clients who meet the following conditions:

- Resident of the US
- US citizen

In order to make a buy recommendation, the investment must be on the buy or strong buy list.

Clients with a moderate or conservative risk profile may not purchase high-risk investments.

Suitability for specific investments is listed below:

Client Criteria	Recommend
Low income or Unemployed, Retired, or Student	Fund and Deposit
Within 5 years of retirement	Fund, Deposit, and Bond

Tax bracket 28% or less	Bond
Officer of employer company and tax bracket more than 36% or High Income and Married with 3 children	Stock
Aggressive investor and has been client for at least 3 years	Stock Index
Employed (including self employed), male, single, high income and young	Future, Option, Swap
Corporate client registered and resident in the USA	Fund

Source Data Tables

Client Table

Column_Name	Data Type	Code Value	Description
Client_Id	Long Integer		Unique identifier of Client
Client_Type	Char(3)		
		IND	Individual Client
		ORG	Organization Client
Citizenship_Status	Char(3)		
		US	US Citizen
		FR	Foreign Citizen
Client_Effective_Date	Date		Has been client since this date.
Client_Years	Integer		
Client_Name	Char(100)		
Client_Status	Char(3)		
		ACT	Active Client
		INA	Inactive Client
Date_Of_Birth	Date		
Age	Integer		
Desired_Retirement_Age	Integer		
EMail_Address	Char(100)		

Employee_Status	Char(3)		
		OFF	Officer
		SHR	Shareholder
Employment_Status	Char(1)		
		1	Employed
		2	Self Employed
		3	Unemployed
		4	Retired
		5	Student
Federal_Tax_Bracket	Char(1)		
		1	15% Tax Bracket
		2	28% Tax Bracket
		3	31% Tax Bracket
		4	36% Tax Bracket
		5	39.6 Tax Bracket
		6	Other Tax Bracket
Gender	Char(1)		
		M	Male
		F	Female
Annual_Income	Long Integer		
Individual_Status	Char(1)		
		0	Alive
		1	Deceased
Investor_Risk_Level	Char(1)		
		1	Aggressive Investor

		2	Moderate Investor
		3	Conservative Investor
Marital_Status	Char(1)		
		1	Single
		2	Married
		3	Domestic Partner
		4	Divorced
		5	Separated
		6	Widowed
Net_Asset_Value	Long Integer		
Number_Of_Children	Integer		
Number_Of_Dependents	Integer		
Number_Of_Employees	Integer		
Org_Business_Size	Char(1)		
		1	Large Business
		2	Medium Business
		3	Small Business
Org_Citizenship	Char(1)		
		1	US Corporation
		2	Foreign Corporation
Organization_Status	Char(1)		
		1	Active Organization
		2	Defunct Organization
Organization_Type	Char(1)		
		1	Trust Client
		2	Government Client

		3	Corporate Client
Phone_Number			
Residence_Status	Char(1)		
		1	US Resident
		2	Foreign Resident
Tax_Status	Char(1)		
		1	Tax Deferred
		2	Tax Exempt
		3	Taxable
		4	Not Tax Deferred

Product Table

Column_Name	Data Type	Code Value	Description
Product_Id	Long Integer		
Current_Bid_Price	Double		
Current_Offer_Price	Double		
Cusip	Char(15)		
Symbol			
Beta	Single		
Name	Char(100)		
Issue_Date	Date		
Payment_Frequency	Integer		
Previous_Bid_Price	Double		
Previous_Offer_Price	Double		
Price_Type	Char(1)		

		O	Opening Price
		C	Closing Price
		I	Intra day Price
Investment_Nature	Char(1)		
		1	Growth Investment
		2	Income Investment
		3	Speculative Investment
Investment_Ranking	Char(1)		
		1	Rank Strong Buy
		2	Rank Buy
		3	Rank Neutral
		4	Rank Sell
		5	Rank Strong Sell
Investment_Risk_Type	Char(1)		
		H	High Risk Investment
		L	Low Risk Investment
Product_Type	Char(1)		
		1	Financial Instrument
		2	Fund
Instrument_Type	Char(1)		
		1	Bond
		2	Stock
		3	Future
		4	Swap

		5	Option
		6	Stock Index
		7	Currency Instrument
		8	Deposit Instrument
Trading_Type	Char(1)		
		1	OTC Traded Instrument
		2	Exchange Traded Instrument

1.3 ABC Brokerage Technical Solution Requirements

ABC Brokerage's Investment Managers must be able to view real-time Knowledge Broker recommendations for a particular client and the client profile (taken from the client table) on a single web page. Assume that Knowledge Engineering requirements have determined that recommendations are generated from rules of the form: "if <CLIENT> recommend buy <PRODUCT>". That is, the rule components are as follows:

1. Event: none
2. Antecedent: CLIENT
3. Action(s): recommend buy
4. Consequent: PRODUCT

The Client and Product tables are stored in Excel files, *client.xls* and *product.xls*. There is an additional Code table stored in *code.xls* that translates the codes in the Client and Product tables into values. There is no custom business analysis required, though the solution must maintain data for both the Client and Product tables ultimately used by the Knowledge Broker.

1.4 Case Study Exercises

1.4.1 Knowledge Engineering Exercises

Exercise 1: Build Ontology Data Elements

Determine the data that is required within the decision making process and set up the ontology within m2. *Hint*: Create key data concepts Party and Product/Service. Create Data Concepts Client and Investment. Note that m2 sets the correct case automatically for object names in the ontology. See <Edit> <Special Editing Functions> <Edit Object Name Conventions>

This exercise requires opening m2 and setting up the ontology that is necessary to support the recommendations and associated rules for the case study.

Exercise 2: Load Source System Data
Load the source system data definitions into m2. This is performed by taking the data definitions as supplied in Excel and creating a CSV file in a form suitable for export into m2. The format of the CSV file is:

<Data Model name>,<Data Area name>,<Entity name>,<Attribute name>

Use DB2 for the Data Model name.

Exercise 3: Complete Ontology
Set up filter and calc expressions as required.
Map to source data items where required.

Exercise 4: Define Rule Base
Create the rule workflows in m2 based on the information gather during the requirements session.

1.4.2 Technical Solution Exercises

Exercise 1: Determine whether or not a data source has to be built.

Apply data source development cookbook to answer the following questions:

1. *Does the antecedent data exist?*
2. *Does the consequent data exist?*
3. *If the data currently exists, how many data sources does the data come from?*
4. *Does the data come from one of the supported storage types?*

Exercise 2: Determine whether or not custom business objects have to be built and if so, design a valid J2EE implementation of the custom business objects.
Apply custom business object development cookbook to answer the following questions:

1. *Is custom business analysis required for the solution?*

2. *Does the Knowledge Broker implementation need to maintain data used as input into the Knowledge Broker?*

3. *If custom business development is necessary, design a valid set of custom business components in J2EE.*

Exercise 3: Determine if a UI has to be built and if so, design a valid J2EE implementation of the UI.

Appendix 4: ABC Brokerage Case Study

Build Solution

Black Pearl Software, Inc.
November 2, 2000

Note to participant

The names, characters, places, and incidents in this workbook are fictitious. Any resemblance to actual businesses, products, persons, events or locales is entirely coincidental.

Your organization has signed a contract and non-disclosure entitling your participation in this training program. This entitles you to participate in this training and to use and retain these materials. Neither this program nor any part thereof may be reused or reproduced in any form or medium now known or hereafter invented or created without the express written permission of Black Pearl, Inc.

000X

Contents

1. Build Solution

1.1 Build and Integrate ABC Brokerage Solution

During this phase the Knowledge Engineering track and the Technical Solution track start to merge and assemble the final solution. Assume that the Client and Product tables are stored in an MS-Access database with the business components and JSPs deployed on WebLogic.

1.2 Case Study Exercises

1.2.1 Knowledge Engineering Exercises

Exercise 1: Populate Knowledge Broker
Populate the Knowledge Broker with the rule base and the ontology for ABC Brokerage. (Hint: Generate the BPOntology.xml and the BPRuleBase.xml from m2)

Exercise 2: Map Data Sources in Knowledge Broker
Map all of the data sources used in the ABC Brokerage solution into the Knowledge Broker.

1.2.2 Technical Solution Exercises - Build

Exercise 1: Develop the data source in MS-Access.
Export the Excel tables into an MS-Access database "training.mdb" with tables of the same name. Assume that "Client ID" is the primary key in the Client table, "Product ID" is the primary key in the Product table, and "Code ID" is the primary key in the code table.

Exercise 2: Review/Develop and deploy the custom business objects.
After storage has been built for the Client, Product, and Code tables, custom business objects must be built that manage data for these tables. The J2EE spec suggests that entity EJBs can be used for this purpose. For the Case Study, either develop these custom entity EJBs or review and use the included container-managed entity EJBs and build files:

a. Client EJB
    i. Client.java (the Client remote interface)
    ii. ClientBean.java (the Client bean implementation)

   iii. ClientHome.java (the Product home object)
b. Product EJB
   i. Product.java (the Product remote interface)
   ii. ProductBean.java (the Product bean implementation)
   iii. ProductHome.java (the Product home object)
c. Code EJB
   i. Code.java (the Code remote interface)
   ii. CodeBean.java (the Code bean implementation)
   iii. CodeHome.java (the Code home object)
d. deployment files
   i. build.cmd (the build file)
   ii. ejb-jar.xml (the EJB deployment descriptor)
   iii. weblogic-ejb-jar.xml (the weblogic deployment descriptor)
   iv. weblogic_cmp_rdbms_client.xml (the client entity bean descriptor)
   v. weblogic_cmp_rdbms_product.xml (the product entity bean descriptor)
   vi. weblogic_cmp_rdbms_code.xml (the code entity bean descriptor)

If you want to review the code, a VisualCafe project, training.vep, has been setup for viewing the EJBs' component Java files.

The build file has also been setup to facilitate bean deployment, along with the necessary deployment descriptors (ejb-jar.xml, weblogic-ejb-jar.xml, weblogic_cmp_rdbms_client.xml, weblogic_cmp_rdbms_product.xml, weblogic_cmp_rdbms_code.xml). The deployment descriptors contain miscellaneous deployment information used by the EJB server. To build the files, run the build file. A DOS screen comes up showing build progress, and should generate a training.jar file under your build/lib directory.

Exercise 3: Review/Develop the sample HTML and JSPs.

After custom business components are built, the ABC Brokerage Knowledge Broker solution UI can be built using the custom business components and the solution data source. For the Case Study, either develop these UI components or review and use the included JSPS, index.jsp and detail.jsp.

a. simple JSP example – index.jsp

Open *index.jsp* through HomeSite. A JSP is any file with a .jsp extension and written in a mark-up language that looks just like plain HTML plus a few extended tags. JSP tags are equivalent to other HTML extensions written for VBScript and javascript.

Unlike normal HTML requests that return static pages to browsers, JSP requests are routed to a special server-side program called the *page-compile servlet*. The page-compile servlet converts the JSP into a JSP-servlet that creates responses displayed in the browser. JSPs with no special tags (i.e., files that end in JSP but written wholly in HTML tags) are converted into JSP-servlets that always return the same HTML page for different requests. JSPs with special tags instruct the page-compile servlet to convert the JSPs into JSP-servlets that often return different HTML pages for different requests.

For example, *index.jsp* uses the 3 types of tags as follows

```
<%@
    page import="javax.naming.*,training.ejbeans.client.*"
%>

<%
  try {
    // get the system context
    Context context = new InitialContext(System.getProperties());
    // use the system context to lookup the client home object
    ClientHome clientHome = (ClientHome) context.lookup("training.client");
    // get all the clients from the home object (i.e., retrieve all the rows from
    // the client table in the database)
    Enumeration clientEnum = clientHome.findAllClients();

    while (clientEnum.hasMoreElements()) {
            Client client = (Client) clientEnum.nextElement();
            // get the client name and ID
            String name = client.getClientName();
            String id = client.getClientID();
```

```
%>


 <tr>
  <td width="100%">
    <font size="2" face="Arial Narrow">
    <a href="detail.jsp?clientID=<%=id%>"><%=name%></a>
    </font>
  </td>
 </tr>


<%
    }
    } catch (Exception e) {
            e.printStackTrace();
            out.println("<p><font color='#FF0000' face='Arial Narrow'>");
            out.println("A system error has occurred.  Please try again later.");
            out.println("</font></p>");
    }
%>
```

These tags are called:

1. <%@ … %> - directive
2. <% … %> - scriptlet
3. <%= … %> - expression

A *directive* instructs the page-compile servlet to perform special functions, such as module includes. A *scriptlet* instructs the page-compile servlet to embed the enclosed Java script directly into the JSP-servlet. An *expression* is computed and transformed at run-time into a String used directly in the HTML. For more details on

how to use WebLogic JSP (including a complete list of extended tags supported by WebLogic server), please refer to the WebLogic JSP documentation on: http://www.weblogic.com/docs51/classdocs/API_jsp.html#setup.

b. complex JSP example – detail.jsp

Open *detail.jsp* through HomeSite. The complex programming logic in *detail.jsp* is as follows:

```
<%
    // get a handle to the Ontology with the given name
    Ontology ontology = OntologyAPI.getOntology("BlackPearlOntology");
    String conceptName = "Client";
    // construct a filter for the Ontology concept that will retrieve only the instance required
    // filters are Strings of the form:  ['@FEATURE_NAME$OP$'VALUE']*, where
    // FEATURE_NAME := the name of the Feature in the Ontology
    // OP := =|>=|<=|!=
    // VALUE := the value for the comparison, with special characters delimited by a /
    String filter = "[@CLIENT ID$=$'" + clientID + "']";

    // retrieve the instance from the Ontology
    Iterator instances = ontology.getInstances(conceptName, null, filter);
    // the instance list should not be empty
    if (!instances.hasNext()) {
            throw new Exception("No instances found for concept " + conceptName
                                    +  " with filter " + filter + ".  One instance required.");
    }

    // get an advisor object from the container
    AdvisorHome advisorHome = (AdvisorHome) context.lookup("blackpearl.advisor");
    // the advisor key must be included as a special concept in BPInit.xml
    Advisor advisor = advisorHome.create(new
    AdvisorKey("BlackPearlOntology/Advisors"));

    // create a Cursor to the instance retrieved from the Ontology
    DataObjectsCursor facts = new DataObjectsCursor(instances);

    // get recommendations from the Advisor bean, which returns a Cursor to the
    // recommendation list
    Cursor cursor = advisor.getRecommendations(facts);

    // create a handler that processes the objects in the Cursor
    MyHandler handler = new MyHandler(codeHome);
    cursor.setObjectHandler(handler);

    // iterate through the Cursor
    while(cursor.next()){}
    // print the handler output
    out.println(handler.getOutput());

    …
```

```
%>

<%!
	/**
	 * Custom handler that processes the objects in the Cursor.  The handler converts
	 * objects in the cursor into HTML Strings pooled in an instance variable.  The
	 * handler provides a method that extract that pool and deliver the HTML to the
	 * browser.
	 *
	 */
	public class MyHandler implements ObjectHandler
	{
		// contains all the output returned to the browser
		String output = "";
		// used as a counter in the recommendation pool
		int count = 1;
		// used to translate codes to the appropriate values
		CodeHome codeHome = null;

		/**
		 * @param codeHome the code Home object used to translate codes to
		 *		the appropriate values
		 */
		public MyHandler(CodeHome codeHome) {
			super();
			this.codeHome = codeHome;
		}

		/**
		 * @return the HTML output String displayed in the browser
		 */
		public String getOutput() {
			return output;
		}

		/**
		 * called by the Advisor bean when a recommendation is given with
		 * with the type and attributes
		 *
		 * @see ObjectHandler
		 */
		public void startObject(String type, AttributeList attributes) {
			String productType = this.getAttribute("PRODUCT TYPE", attributes);

			if (productType.equals("1")) {
				// the product is an investment instrument
				// get all the investment instrument attributes that will
				// be displayed in the UI
				String symbol = translate(codeHome, "Symbol",
					getAttribute("PRODUCT NAME", attributes));
				…
				String finding = getFinding(attributes);

				// construct the HTML that displays the product recommendation
				this.output += "  <tr>";
				…
```

```
            this.output += "  </tr>";
} else if (productType.equals("2")) {
            // the product is a fund
            // get all the investment instrument attributes that will
            // be displayed in the UI
            String symbol = getAttribute("PRODUCT NAME", attributes);
            …
            String finding = getFinding(attributes);

            // construct the HTML that displays the product recommendation
            this.output += "  <tr>";
            …
            this.output += "  </tr>";
                                    }
}

/**
 * Hook that does nothing.
 *
 * @see ObjectHandler
 */
public void endObject(String type) {
}

/**
 * @return the value for the given attribute, or "" when the attribute doesn't
 *                      exist or either the attribute name or attribute list is null
 *
 * @param attributeName the attribute being queried
 * @param attributes the attribute list that contains the attribute value
 */
String getAttribute(String attributeName, AttributeList attributes) {
            if ((attributeName != null) && (attributes != null)) {
                        int index = attributes.find(attributeName);

                        if (index >= 0) {
                                    return (String) attributes.getValue(index);
                        }
            }
            return "";
}

 /**
 * @param explanation String from the Advisor bean
 *
 * @return the detailed explanation within the explanation String,
 *         which starts with "DETAILED EXPLANATION" within the
 *                      full explanation String returned by the Advisor bean
 */
String getFinding(AttributeList attributes) {
            String explanation = getAttribute("explanation", attributes);

            int index = explanation.indexOf("DETAILED EXPLANATION:");
            if (index > 0) {
                        return explanation.substring(index + 21,
```

```
                                                    explanation.length() - 1);
                        } else {
                                return explanation;
                        }
                }

        };
%>

<%!
        /**
         * @return the translation of the given code for the given column
         */
        public String translate(CodeHome codeHome, String column, String codeString) {
                // if any of the parameters is null, return the code
                if ((codeHome == null) || (column == null) || (codeString == null)) {
                        return codeString;
                }

                // translate the code into a value
                try {
                        // query the Code home object for a translation
                        Enumeration codes = codeHome.findByColumnCode(column,
                                        codeString);

                        if (!codes.hasMoreElements()) {
                                // there were no translations found, DB error
                                System.out.println("Code " + codeString +
                                        " for column " +  column + " not found.");
                                return codeString;
                        }

                        Code code = (Code) codes.nextElement();

                        if (codes.hasMoreElements()) {
                                // there were multiple translations found, DB error
                                System.out.println("Code " + codeString +
                                        " for column " + column +
                                        " has multiple values.");
                                return codeString;
                        }

                        // there was exactly one translation found, return the translation
                        return code.getDecodedValue();
                } catch (FinderException e) {
                        System.out.println("Error finding code " + codeString +
                                        " for column " + column + ".");
                        return codeString;
                } catch (RemoteException e) {
                        System.out.println("RemoteException finding code " +
                                        codeString + " for column " + column + ".");
                        return codeString;
                }
        }
%>
```

Detail.jsp uses another special tag, <%! … %>, called a *declaration*. A *declaration* instructs the page-compile servlet to perform interpret the Java included as a declaration within the JSP-servlet, either as an instance variable or (in this case) an inner class.

Detail.jsp also uses the Knowledge Broker API as follows:

1. retrieve the Ontology
2. retrieve an instance from the Ontology
3. use the instance as the antecedent to the Rule Engine, wrapped inside the Advisor EJBean
4. invoke the Rule Engine through the Advisor EJBean
5. extract the recommendations from the Cursor returned by the Advisor EJBean

For more details on how to use the Knowledge Broker API (including a complete list of all Knowledge Broker functions), please refer to the Knowledge Broker Javadoc.

1.2.3 Technical Solution Exercises - Integration

Exercise 1: Install Knowledge Broker.
Install Knowledge Broker in d:/BlackPearl and use the JDBC-ODBC bridge (sun.jdbc.odbc.JdbcOdbcDriver.class) as the default driver and substitute the Knowledge Broker setup files with the files developed by in Knowledge Engineering. Modify the Advisor properties file to refer to recommendations of the form “if <CLIENT> then recommend buy <PRODUCT>".

Exercise 2: Connect the data sources to their concept representation in the Knowledge Broker Ontology.
HINT: If the case study data source is registered as a Windows ODBC data source called “Training”, the Knowledge Broker Data Concept Editor connection should be “jdbc:odbc:Training”.

Exercise 3: Integrate and deploy the business components.
Deploy the custom business components and Knowledge Broker Advisor bean in WebLogic 5.1.

Exercise 4: Integrate and deploy the UI.
Deploy the UI in WebLogic 5.1.

BLACK PEARL

Appendix B: Knowledge Broker Installation

Black Pearl Software, Inc.
November 2, 2000

Note to participant

The names, characters, places, and incidents in this workbook are fictitious. Any resemblance to actual businesses, products, persons, events or locales is entirely coincidental.

Your organization has signed a contract and non-disclosure entitling your participation in this training program. This entitles you to participate in this training and to use and retain these materials. Neither this program nor any part thereof may be reused or reproduced in any form or medium now known or hereafter invented or created without the express written permission of Black Pearl, Inc.

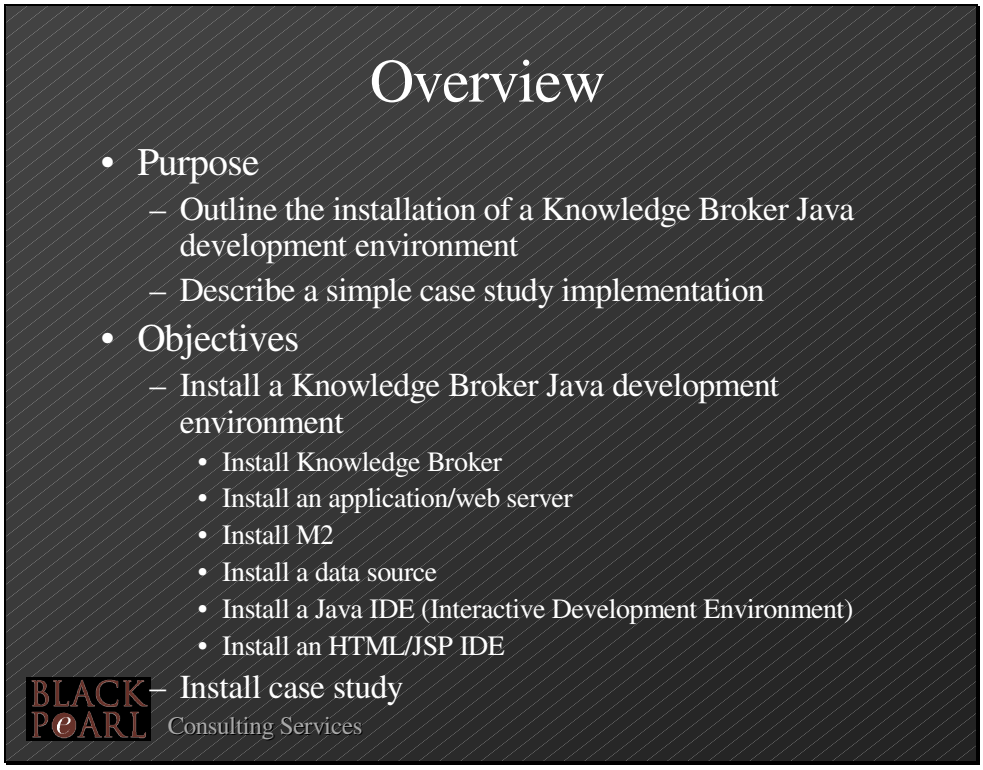
000X

Contents

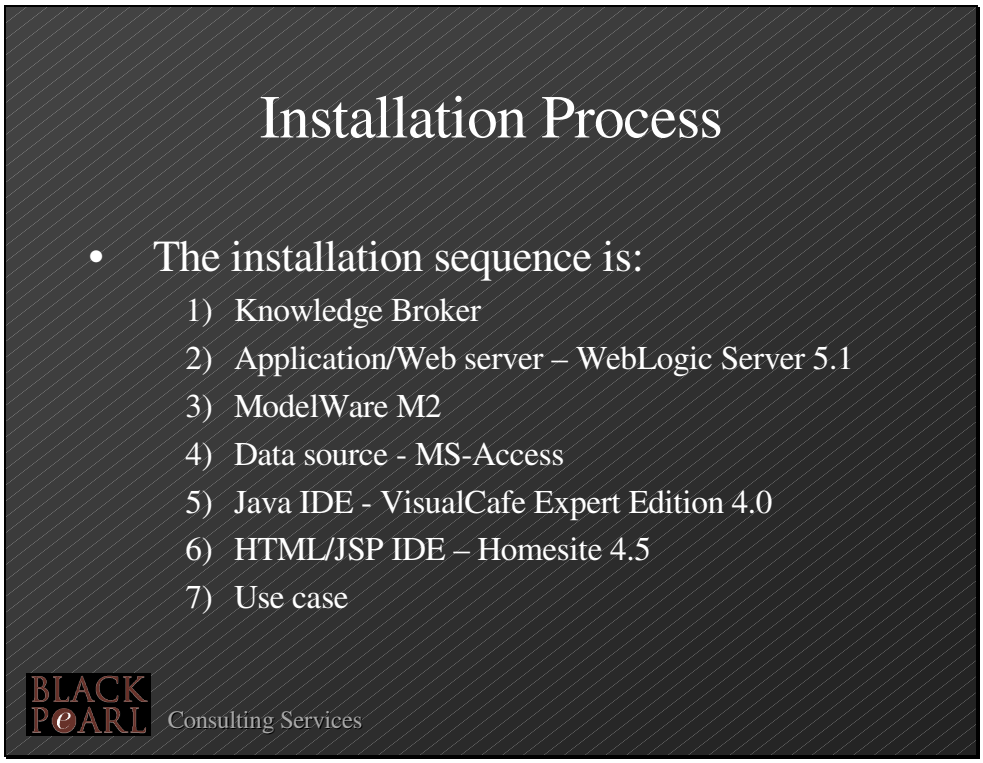
Contents

1. Installation

2. Overview


Overview
• Purpose
– Outline the installation of a Knowledge Broker Java development environment
– Describe a simple case study implementation
• Objectives
– Install a Knowledge Broker Java development environment
• Install Knowledge Broker
• Install an application/web server
• Install M2
• Install a data source
• Install a Java IDE (Interactive Development Environment)
• Install an HTML/JSP IDE
– Install case study
BLACK PEARL
Consulting Services

3. Installation Process

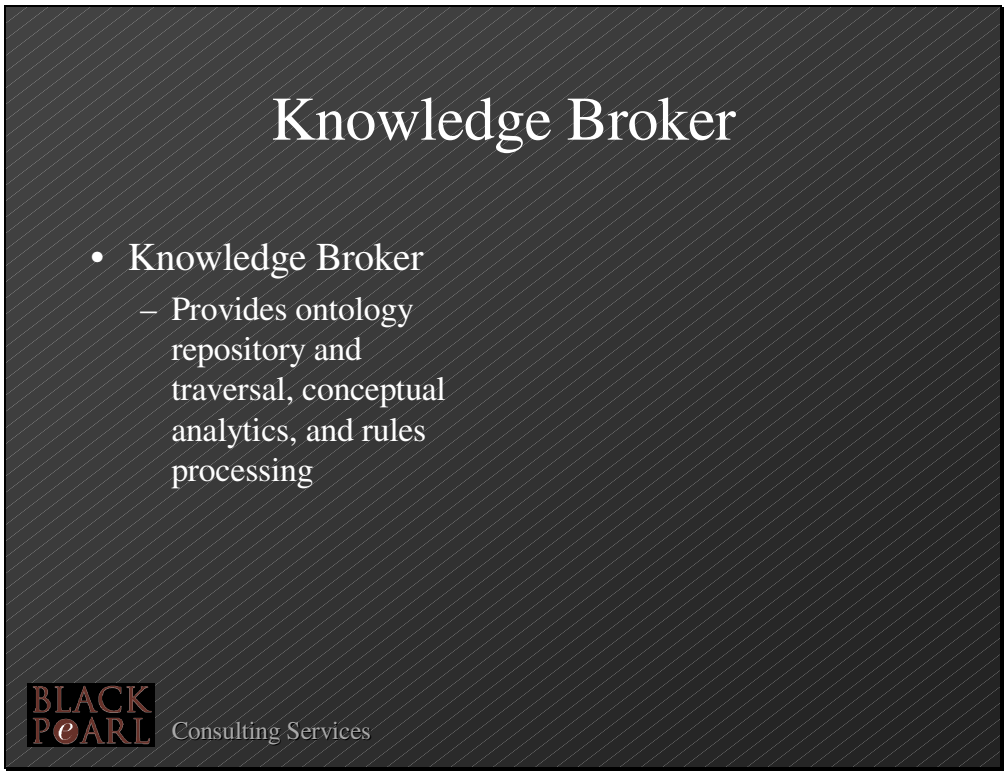


Black Pearl assumes that the hardware has been at least minimally specified and configured according to the information contained in the Knowledge Broker User Guide.

4. Knowledge Broker

Install Standalone version

5. Knowledge Broker



6. Application/Web server

7. Application/Web server

Document worth reading:

http://www.weblogic.com/docs51/install/install_shield.html

License data is stored in .XML file in:

D:\weblogic51\license

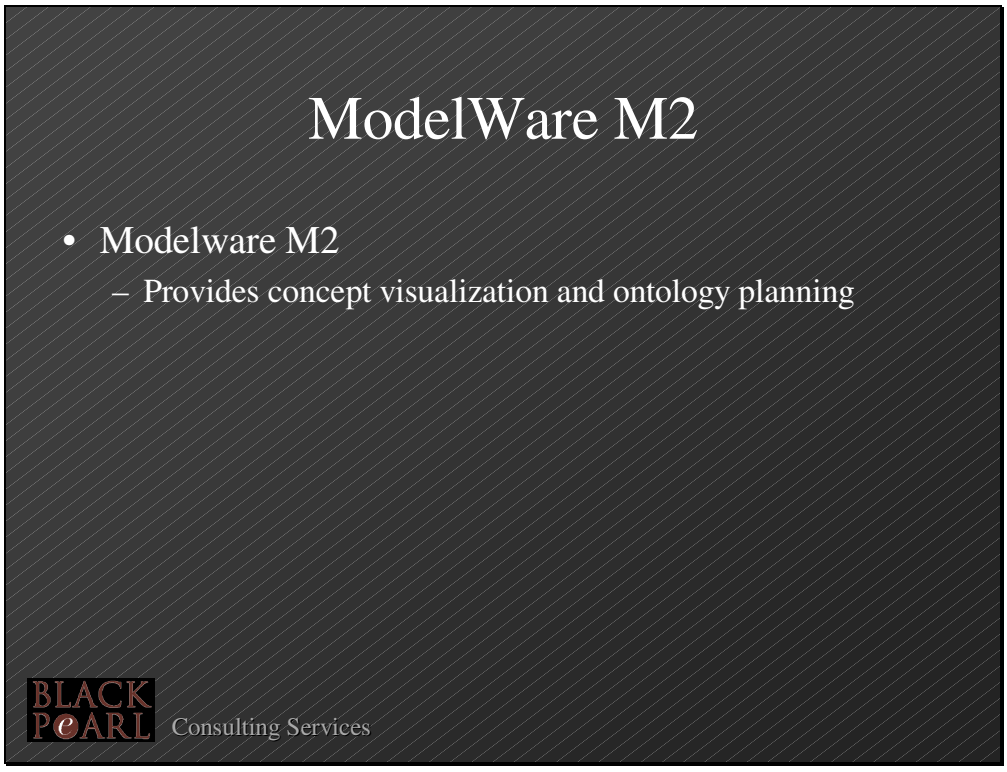
8. Application/Web server

Document to read:

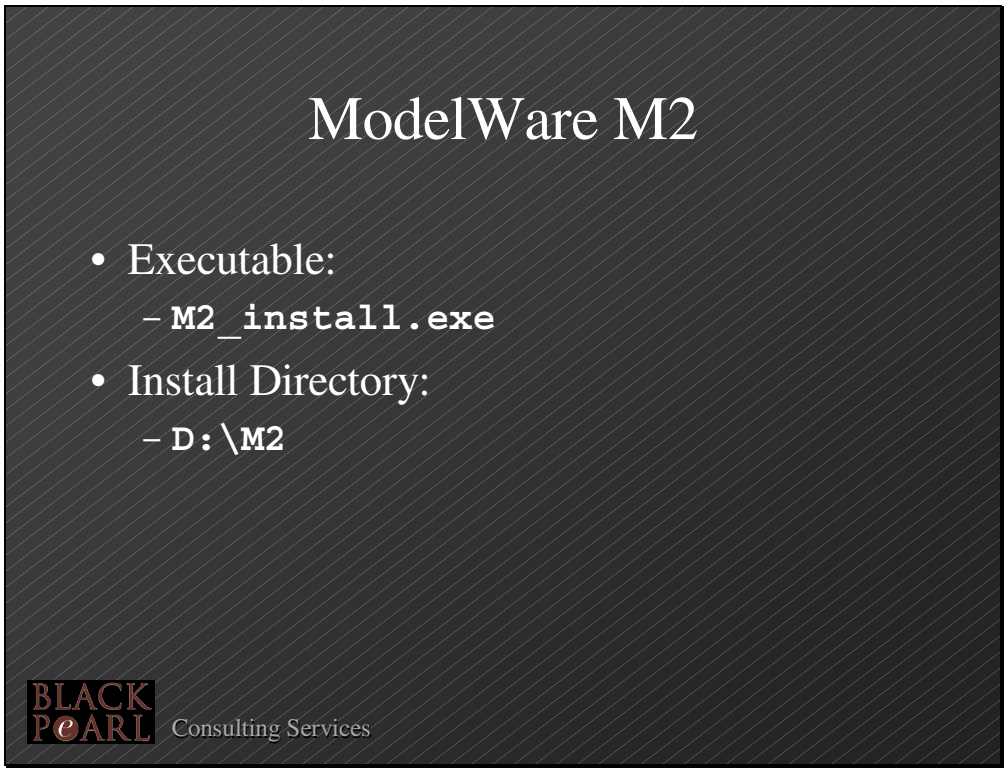
http://www.weblogic.com/docs51/install/startserver.html

Do NOT use MS SDK for Java (Jview)

9. ModelWare M2

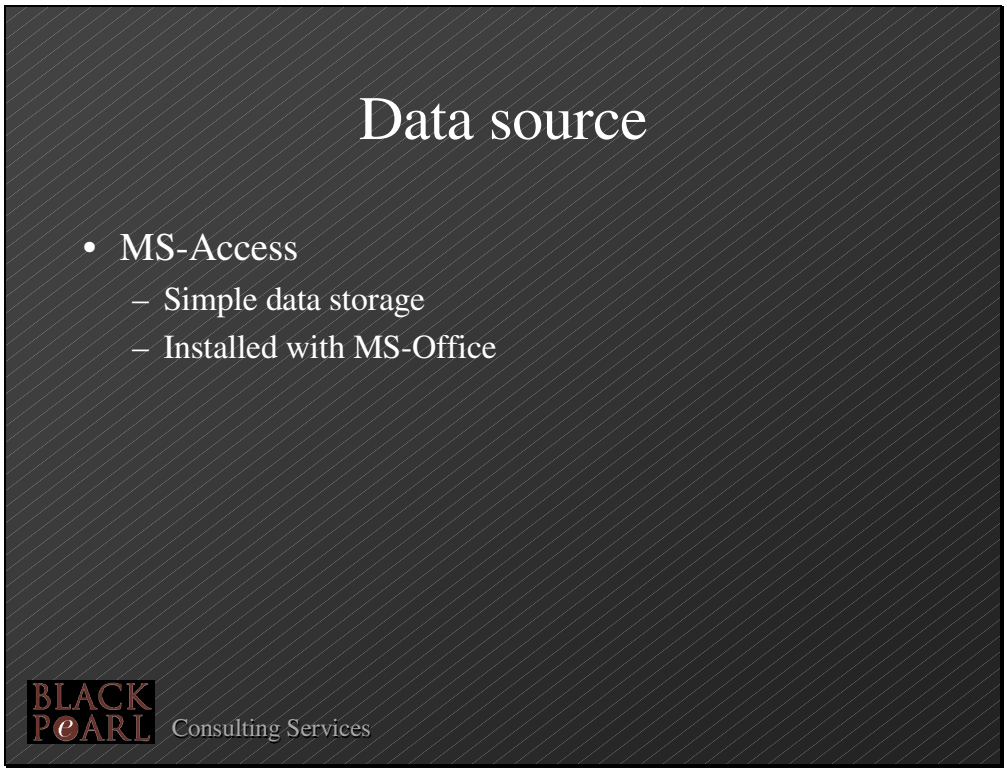


10. ModelWare M2

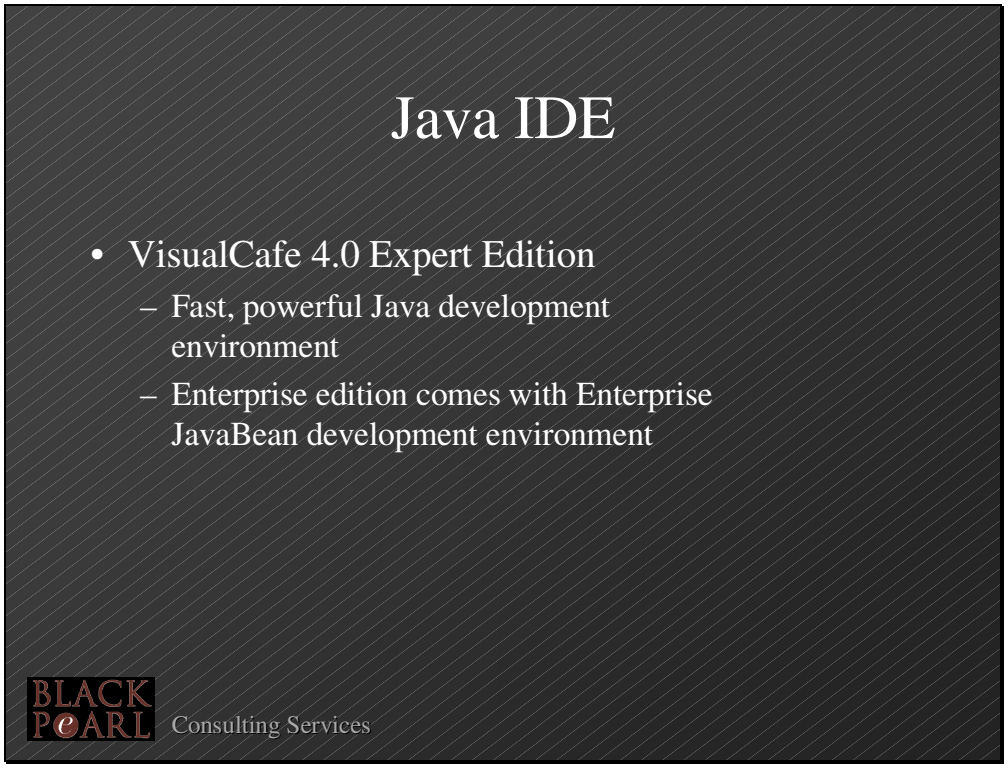


Install into D:\M2, **not** D:\M1 (as recommended)

11. Data Source



12. Java IDE

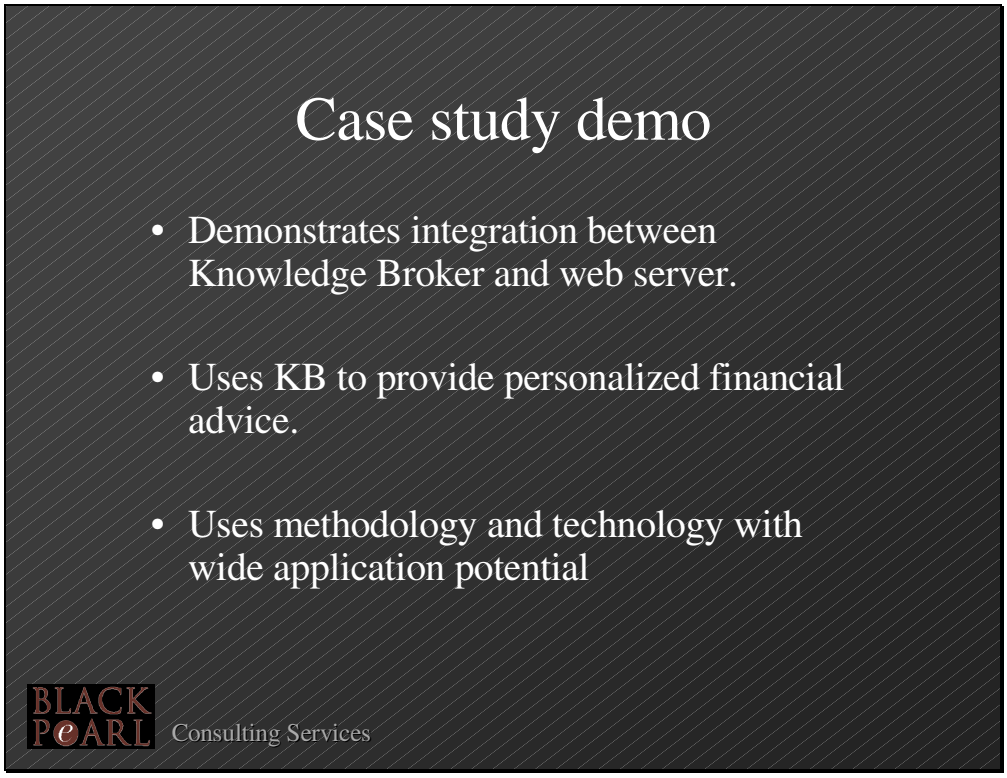

Java IDE
• VisualCafe 4.0 Expert Edition
– Fast, powerful Java development environment
– Enterprise edition comes with Enterprise JavaBean development environment
BLACK PEARL
Consulting Services

13. Java IDE

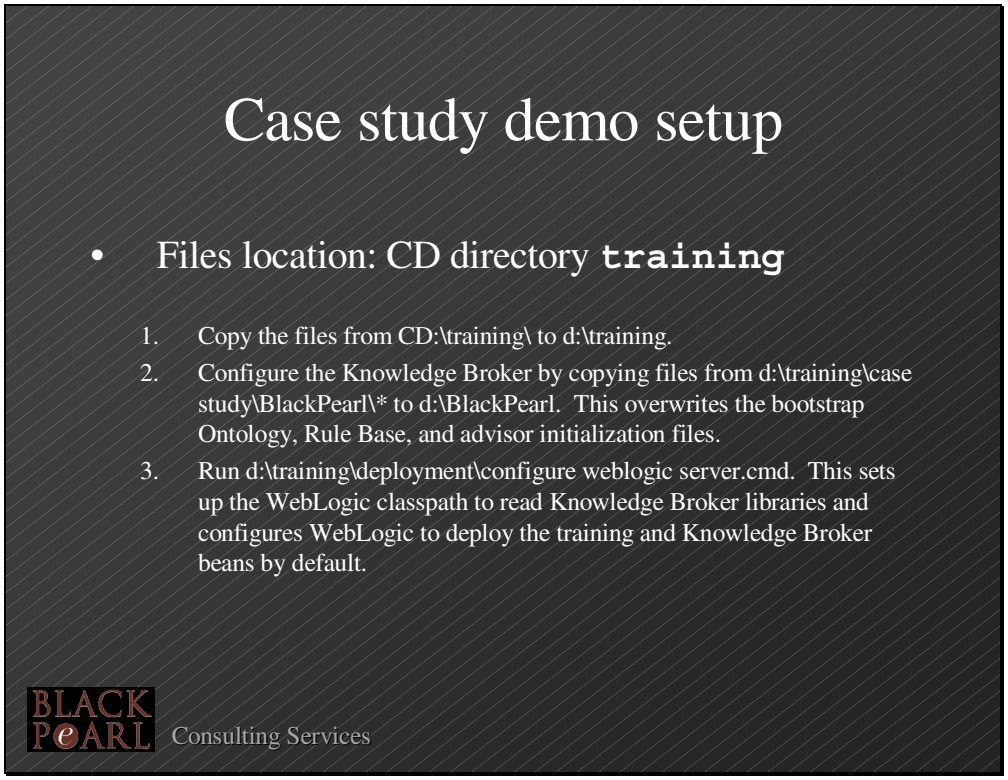

Java IDE
• Executable:
– Visual Cafe Expert 4.0\setup.exe
• Install Directory:
– D:\VisualCafe
BLACK PEARL
Consulting Services

14. HTML/JSP IDE

15. Case Study Demo



16. Case Study Demo Setup



17. Case Study Demo Setup

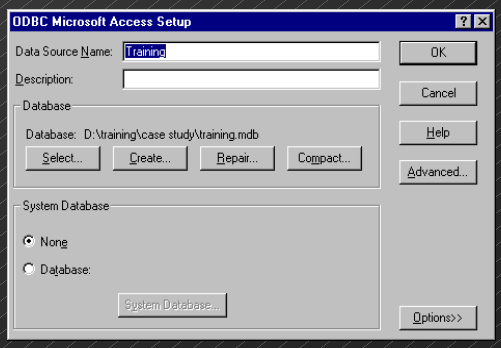
Case study demo setup

- Need to setup MS-Access Data Source

1. Select this program:

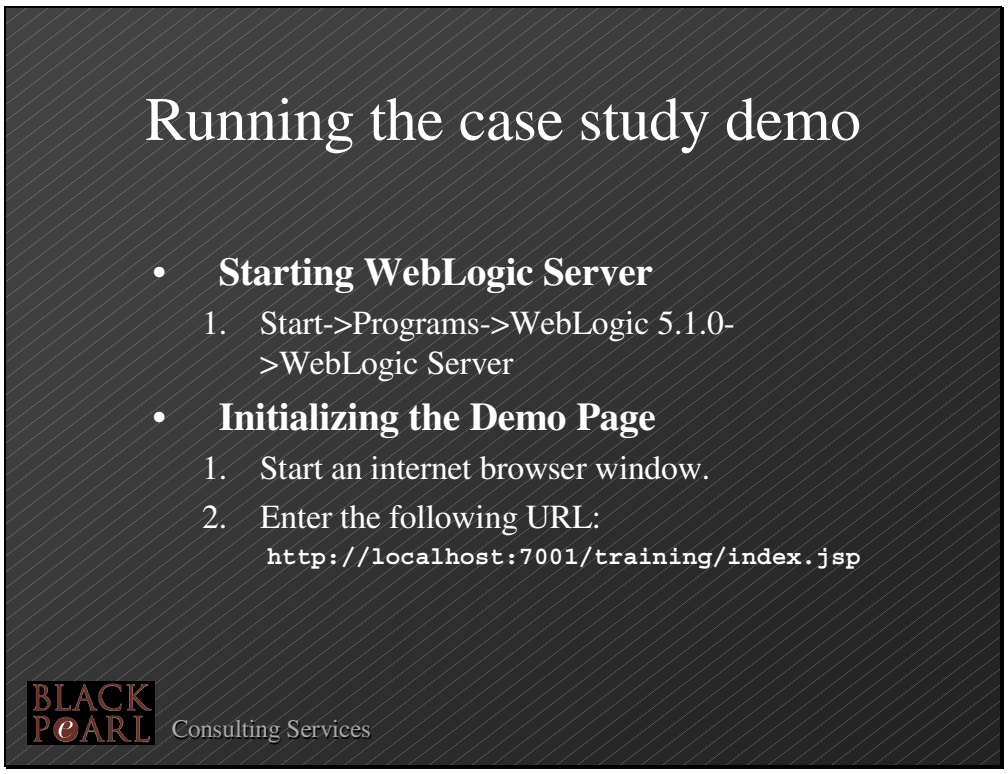
   Start->Settings->Control Panel->Data Sources (ODBC)

2. On the System DSN Tab in the ODBC Data Source Administrator, add a new Access data source using the driver "Driver do Microsoft Access (*.mdb)" and the following settings:



BLACK PEARL Consulting Services

18. Running the Case Study Demo



Appendix C: M2 Overview

Black Pearl Software, Inc.
November 2, 2000

Note to participant

The names, characters, places, and incidents in this workbook are fictitious. Any resemblance to actual businesses, products, persons, events or locales is entirely coincidental.

Your organization has signed a contract and non-disclosure entitling your participation in this training program. This entitles you to participate in this training and to use and retain these materials. Neither this program nor any part thereof may be reused or reproduced in any form or medium now known or hereafter invented or created without the express written permission of Black Pearl, Inc.

000X

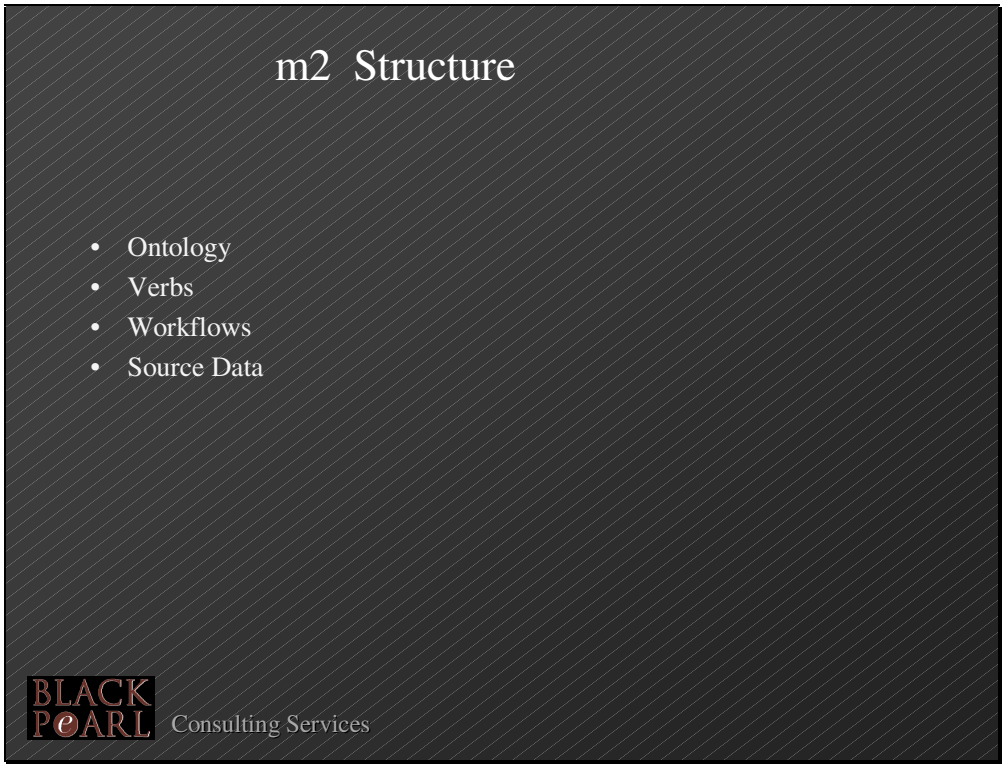
Contents

Contents

1. Appendix C: m2 Overview

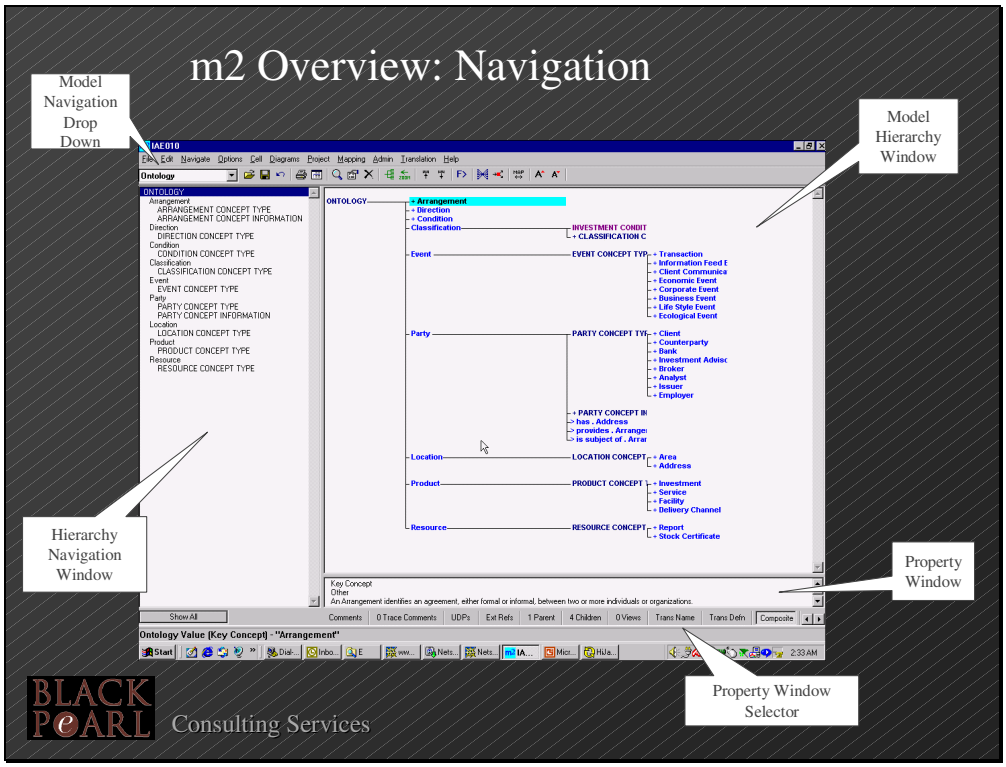
The m2 modeling tool provides a mechanism for modeling the components of a rule base (i.e. the ontology), the list of verbs (or recommendation types), the workflows, and the data sources for the ontology.

2. m2 Structure

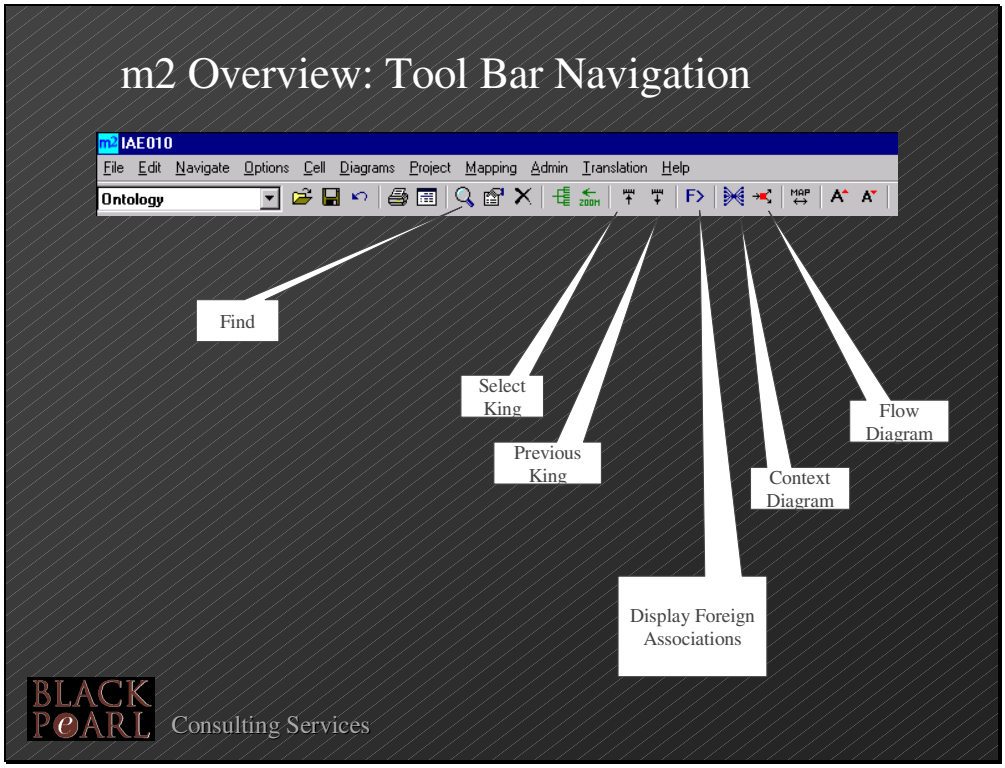


The m2 modeling tool provides a mechanism for modeling the components of a rule base (i.e. the ontology), the list of verbs (or recommendation types), the workflows, and the data sources for the ontology.

3. m2 Overview: Navigation



4. m2 Overview: Tool Bar Navigation



[Invoke m2 to demonstrate the functionality shown on this slide]

Find provides a means of searching for an object based on its name (or part of name) and its object type. For example, find all activities starting with "Recommend." This is very useful when dealing with large models (over 1000 objects).

Select King—Within a hierarchical tree model, the top of the tree is known as the **king**. In order to assist in the navigation of a hierarchy, any node in the hierarchy may be promoted to become a temporary king. This has the effect of moving a branch of the tree to the top of the screen,

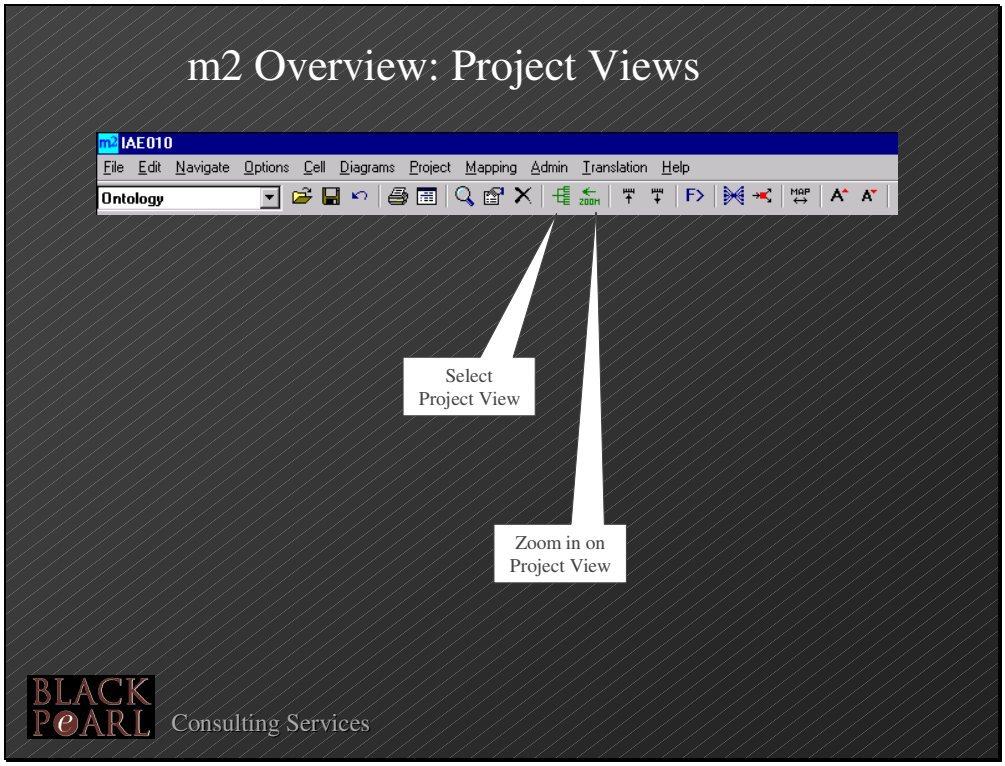
Previous King undoes the last **Select King** operation.

Display Foreign Associations—A foreign association is defined as a mapping between an object in one model hierarchy, and an object in another hierarchy. For example, "Client Type" has source data item "CLNT.CL_TYPE". This button toggles the display of child foreign associations for each object in the currently displayed hierarchy.

Context Diagram—This is a powerful alternate method of viewing the models. It displays a network diagram with the selected object at the center with parent objects to the left and child objects to the right. The diagram may be navigated by moving any displayed object into the center of the context diagram (select object, shift+click)

Flow Diagram may only be activated when an **Activity** is selected. It invokes the Flow Diagrammer, which is a graphical tool that facilitates the representation of activity sequencing.

5. m2 Overview: Project Views



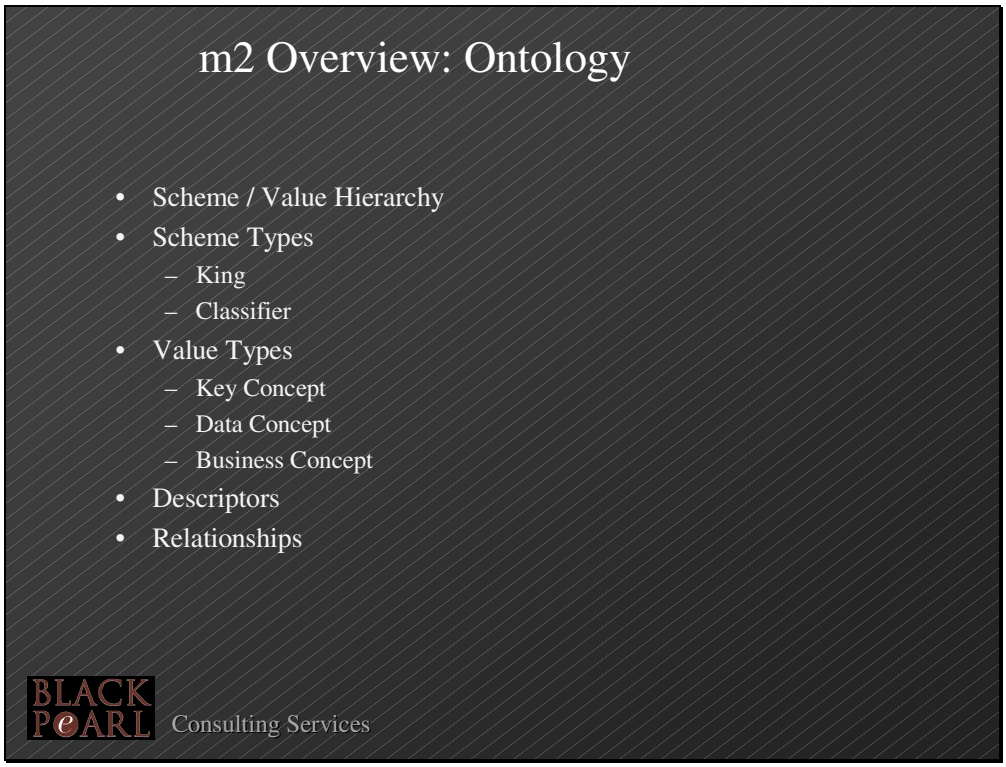
[Invoke m2 to demonstrate the functionality shown on this slide]

Project Views enable the user to create one or more subsets of the model without having to extract the required elements into a separate model. This is achieved by placing each required object into a project view. The action is very similar to taking a highlighter pen and marking objects as required.

Views can be turned on and off using the **Select Project View** button.

The model can be **zoomed** on one or more selected project views; that is, only the objects in that view (and their hierarchy parents) will be displayed. The tool bar button toggles zoom on and off.

6. m2 Overview: Ontology



[Invoke m2 to demonstrate the functionality shown on this slide]

Scheme/Value Hierarchy. The Ontology is arranged as a single hierarchy. The base hierarchy is a classification system where the classification schemes may be considered the questions, and the classification values the answers to those questions. Each classification value may then be classified by further classification schemes.

Scheme/Value Types The root (i.e. top) of the hierarchy is a fixed **King** scheme, named Ontology.

This has child classification values that are **Key Concepts**. Examples are Party, Product, Event etc.

Appendix C

Each Key Concept is classified according to its component **Data Concepts.**

Each Data Concept is classified according to its component **Business Concepts**

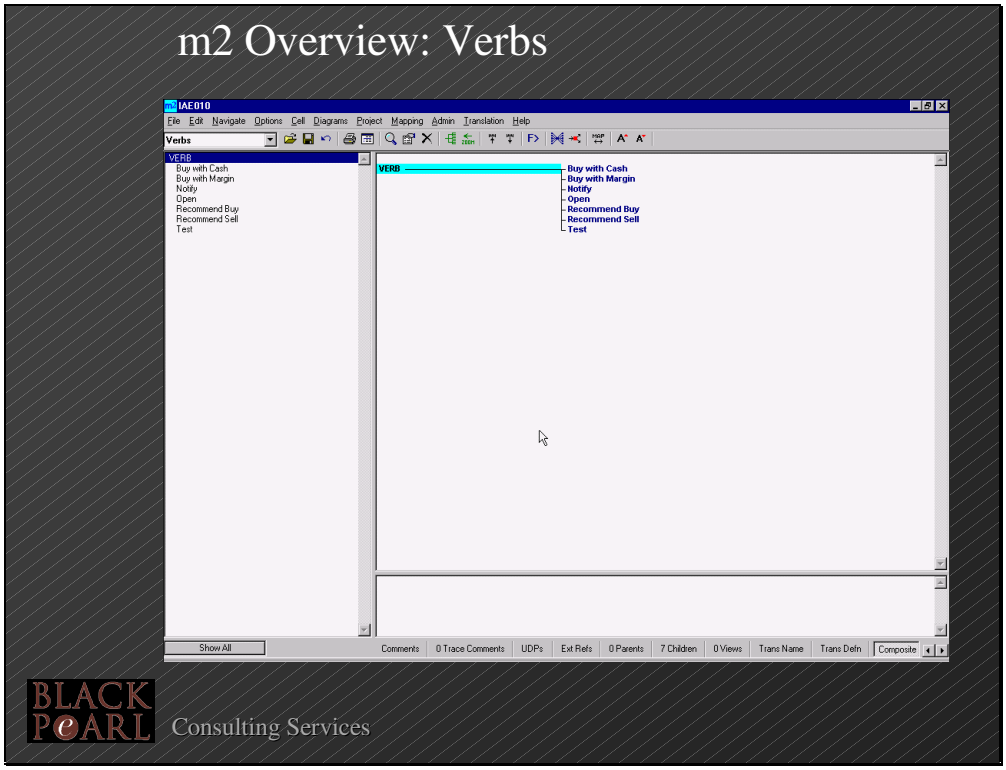
Business Concepts may be further sub classified by Business Concepts.

Descriptors may be attached to any classification value. A Descriptor is an information property of a classification value, such as Name, Age, and Date of Birth.

Relationships define a link between two classification values, for example, Client.is member of Segment.

The Ontology hierarchy implies full inheritance.

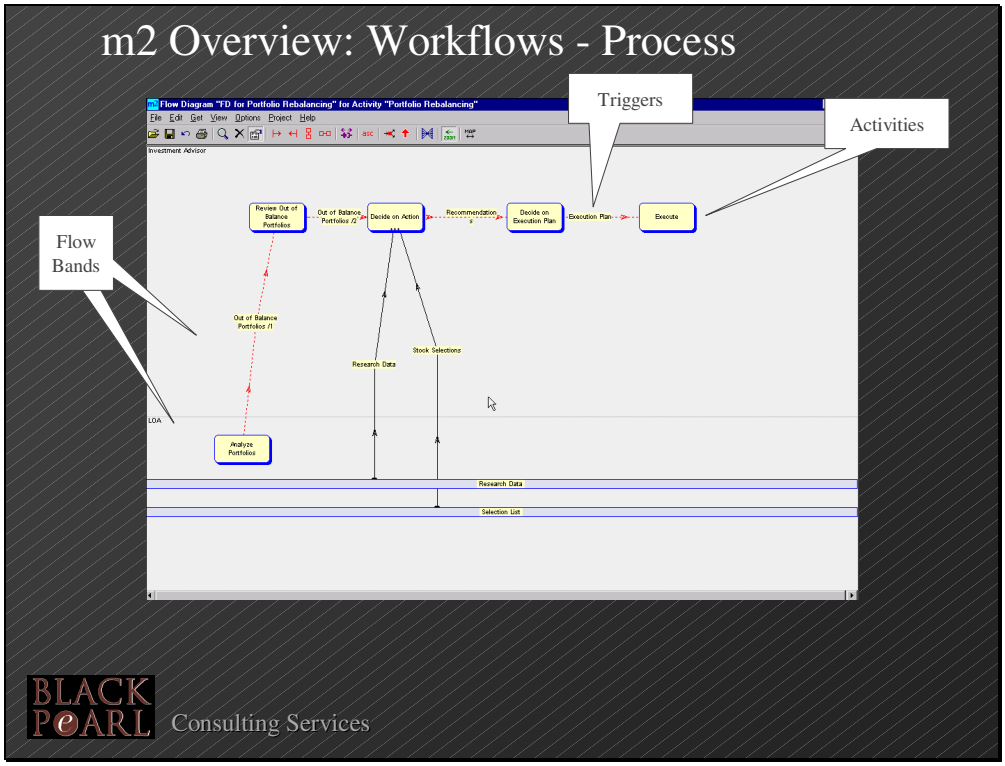
7. m2 Overview: Verbs



[Invoke m2 to demonstrate the functionality shown on this slide]

Verbs are the recommendation verbs used as the outcome activity in rule workflows. This list of verbs should be created before the workflow is created.

8. m2 Overview: Workflows - Process

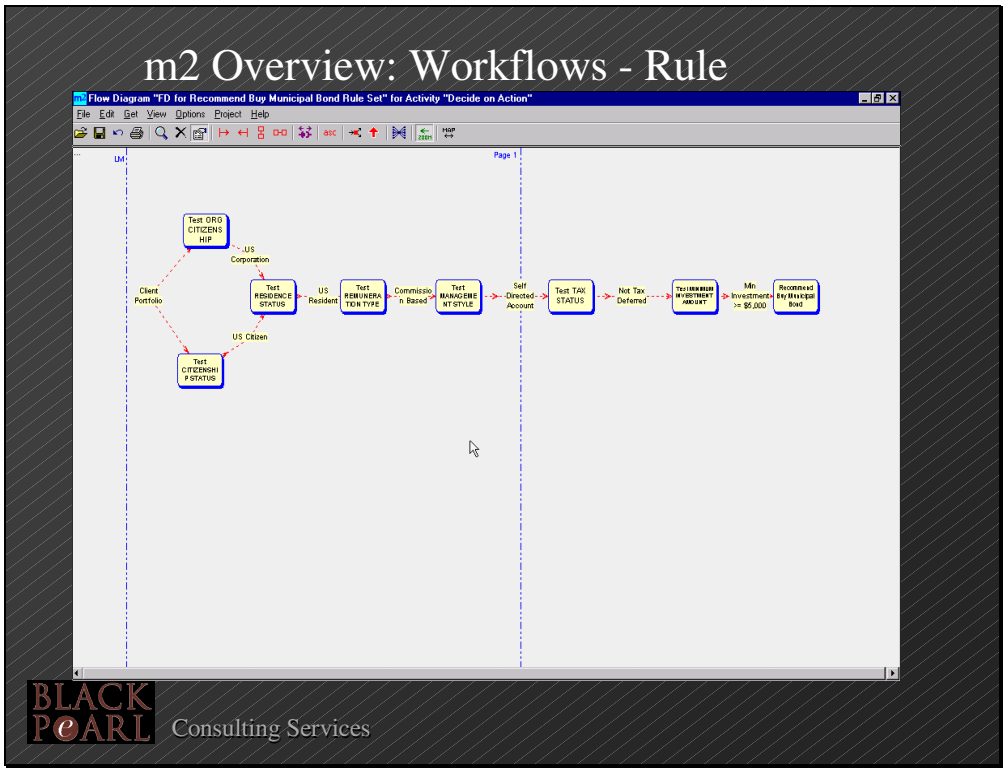


[Invoke m2 to demonstrate the functionality shown on this slide]

The process workflow provides a means of documenting a business process using a method involving laying out the process flow using **Flow Bands** to represent the person, role, department or system performing the constituent activities. The slide shows a simple flow with 2 flow bands—one for the Investment Advisor (role) and one to represent automated processes.

The example here also shows the use of Information Buses, which are used to represent data bases or systems that provide data. Data flows represent the storing and extraction of data.

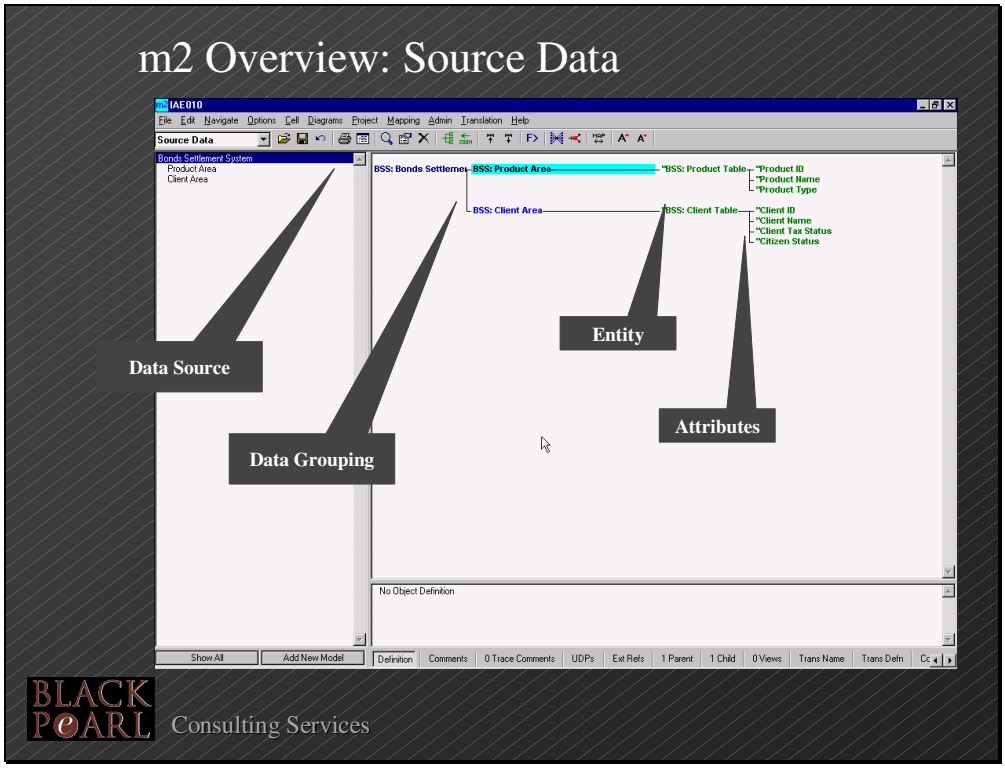
9. m2 Overview: Workflows - Rule



[Invoke m2 to demonstrate the functionality shown on this slide]

A **Rule Workflow** shows the logic that is embedded in a rule. The rule consists of an initiating trigger, a series of test activities, each having one or more result triggers which fire further tests until an outcome is reached.

10. m2 Overview: Source Data



[Invoke m2 to demonstrate the functionality shown on this slide]

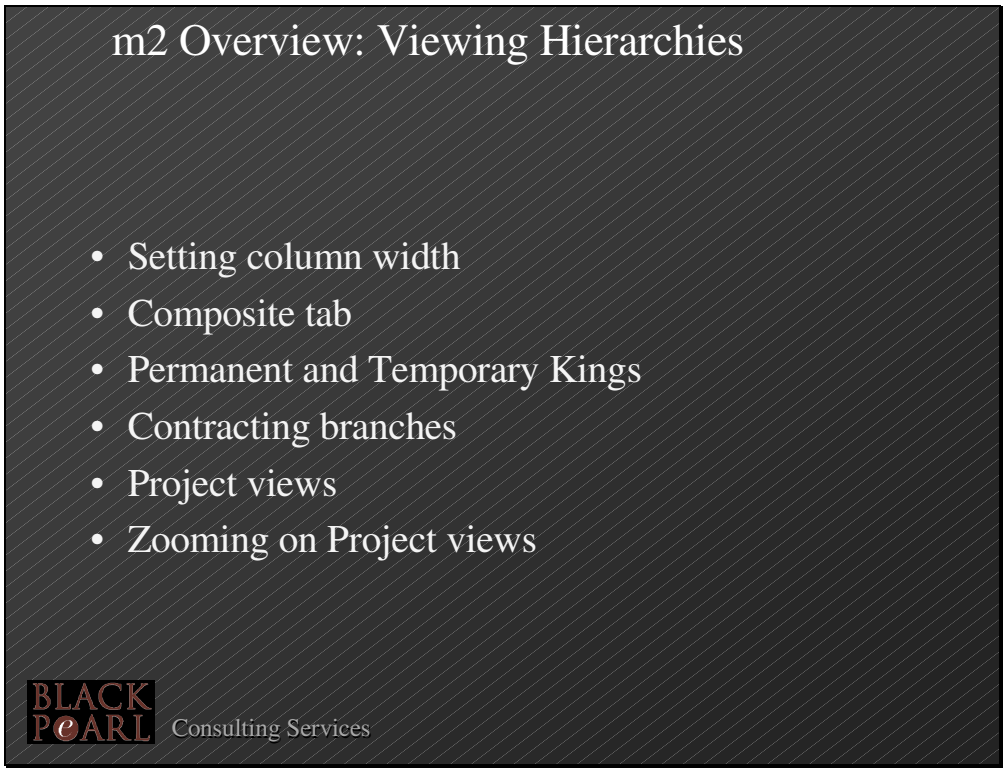
Data Source—Name of a database

Data Grouping—A collection of tables

Entity—Table

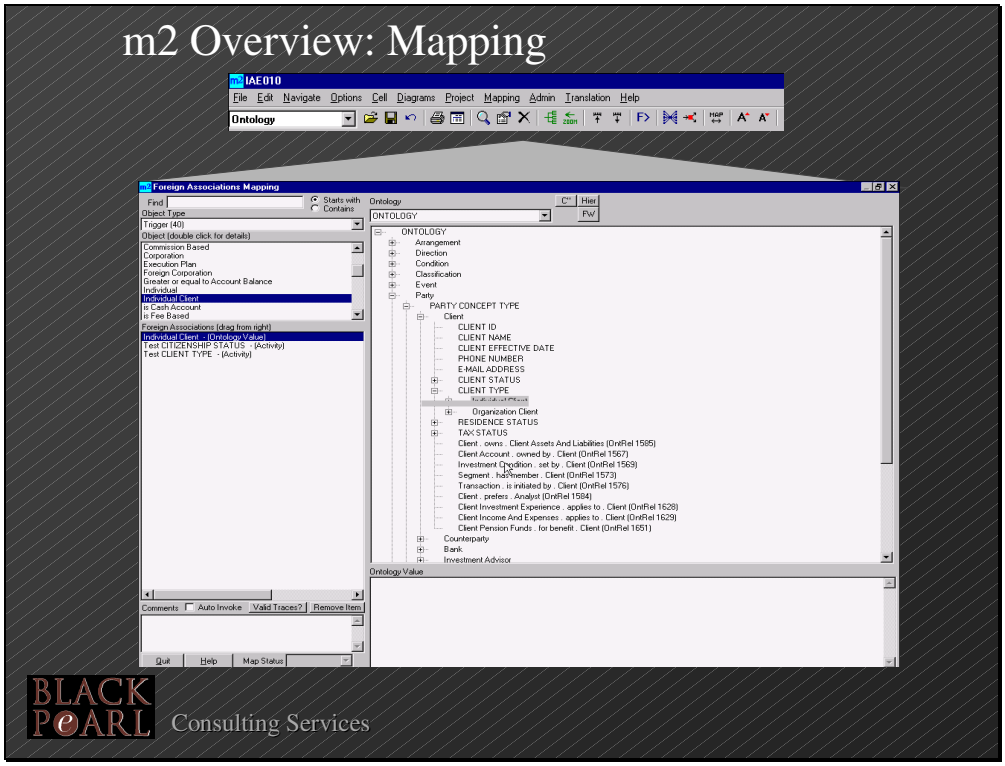
Attribute—Column

11. m2 Overview: Viewing Hierarchies



Demonstrate in m2.

12. m2 Overview: Mapping

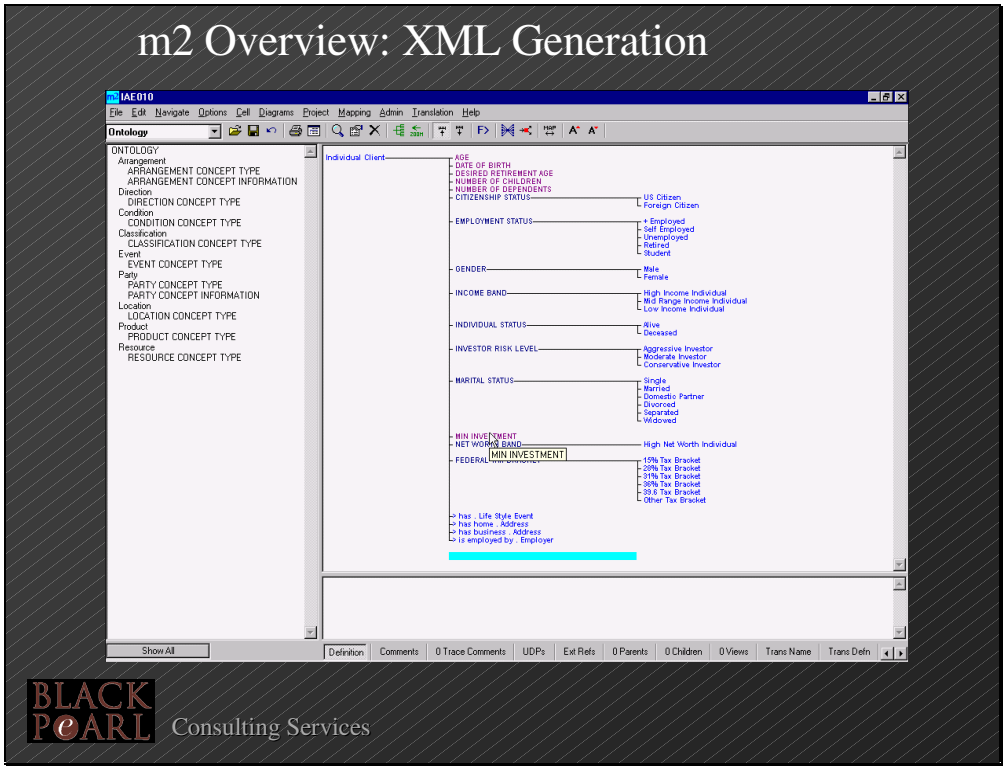


[Invoke m2 to demonstrate the functionality shown on this slide]

Mappings between objects in different hierarchies may be entered using the **Foreign Association Mapping** function. This can be invoked from the tool bar.

The left side of the screen displays object to be mapped from. The right side of the screen displays the object to mapped to. It does not matter which way the mapping is made (i.e. left to right, or right to left). M2 determines the parent-child order.

13. m2 Overview: XML Generation



[Invoke m2 to demonstrate the functionality shown on this slide]

First, the hierarchy for which the XML is to be generated is displayed.

Next, <File> <Print Text as XML>

XML file is created. This may be viewed through a browser or editor of choice such as Notepad, XML Spy or MS XML Notepad.

Appendix D: Feedback

Black Pearl Software, Inc.
November 2, 2000

Note to participant

The names, characters, places, and incidents in this workbook are fictitious. Any resemblance to actual businesses, products, persons, events or locales is entirely coincidental.

Your organization has signed a contract and non-disclosure entitling your participation in this training program. This entitles you to participate in this training and to use and retain these materials. Neither this program nor any part thereof may be reused or reproduced in any form or medium now known or hereafter invented or created without the express written permission of Black Pearl, Inc.

000X

Contents

1. Feedback

Please provide us with your feedback. In order to develop and deliver excellent training, your feedback is invaluable to Black Pearl's educational offerings in the future.

Name (optional): _____ Company (optional)_____

Score: On a scale of 1-5; 1 for "*Needs Work*", 2 for "*Fair*", 3 for "*Average*", 4 for "*Very Good*", 5 for "*Excellent*"

Courseware		
Was the courseware (training materials and slides) helpful?		
Was the case study useful to understand Knowledge Broker?		
Did the case study add educational value?		

Facilities		
Was the environment conducive to learning?		

Overall		
What is your overall satisfaction with the training?		
What is your overall satisfaction with Knowledge Broker?		

How would you like us to do it differently next time? Tell us how you would improve the training. Give as much detail as possible. Write below and on the back of this sheet.