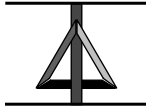


Data Model Patterns: A Metadata Map

East Coast DAMA 2006

David C. Hay
May-June, 2006



Essential Strategies, Inc.

13 Hilshire Grove Lane, Houston, TX 77055
 (713) 464-8316
 dch@essentialstrategies.com
 www.essentialstrategies.com

1

The Architecture Framework . . .

	Data (What)	Activities (How)	Locations (Where)	People (Who)	Time (When)	Motivation (Why)
Objectives / Scope (Planner's View)	List of things important to the enterprise 	List of processes the enterprise performs 	List of enterprise locations 	Organization approaches 	Business master schedule 	Business vision and mission
Enterprise model (Business Owners' Views)	Language, divergent data model 	Business process model 	Logistics network 	Organization chart 	State / transition diagram 	Business strategies, tactics, policies, rules
Model of Fundamental Concepts (Architect's View)	Convergent e/r model 	Essential data flow diagram 	Locations of roles 	The viable system, use cases 	Entity Life History 	Business rule model
Technology Model (Designer's View)	Data base design 	System design, program structure 	Hardware, software distribution 	User interface, security design 	Control structure 	Business rule design
Detailed Representation (Builder's View)	Physical storage design 	Detailed program design 	Network architecture, protocols 	Screens, security coding 	Timing definitions 	Rule specification program logic
Functioning System	<i>(Working System)</i>					
	Converted data	Executable programs	Communications facilities	Trained people	Business events	Enforced rules



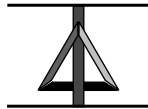
Copyright © 2005, Essential Strategies, Inc.

2

The kinds of question we ask are as many as the kinds of things which we know. They are in fact four: (1) whether the connexion of an attribute with a thing is a fact, (2) what is the reason of the connexion, (3) whether a thing exists, (4) what is the nature of the thing. Thus when our question concerns a complex of thing and attribute and we ask whether the thing is thus or otherwise qualified--whether, e.g., the sun suffers eclipse or not--then we are asking as to the fact of a connexion.

Aristotle
Posterior Analytics

Column 1: Data



Essential Strategies, Inc.

13 Hilshire Grove Lane, Houston, TX 77055
 (713) 464-8316
 dch@essentialstrategies.com
 www.essentialstrategies.com

3

The Architecture Framework . . .

	Data (What)	Activities (How)	Locations (Where)	People (Who)	Time (When)	Motivation (Why)
Objectives / Scope (Planner's View)	List of things important to the enterprise 	List of processes the enterprise performs 	List of enterprise locations 	Organization approaches 	Business master schedule 	Business vision and mission
Enterprise model (Business Owners' Views)	Language, divergent data model 	Business process model 	Logistics network 	Organization chart 	State / transition diagram 	Business strategies, tactics, policies, rules
Model of Fundamental Concepts (Architect's View)	Convergent er model 	Essential data flow diagram 	Locations of roles 	The viable system, use cases 	Entity Life History 	Business rule model
Technology Model (Designer's View)	Data base design 	System design, program structure 	Hardware, software distribution 	User interface, security design 	Control structure 	Business rule design
Detailed Representation (Builder's View)	Physical storage design 	Detailed program design 	Network architecture, protocols 	Screens, security coding 	Timing definitions 	Rule specification program logic
Functioning System	(Working System)					
	Converted data	Executable programs	Communications facilities	Trained people	Business events	Enforced rules



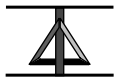
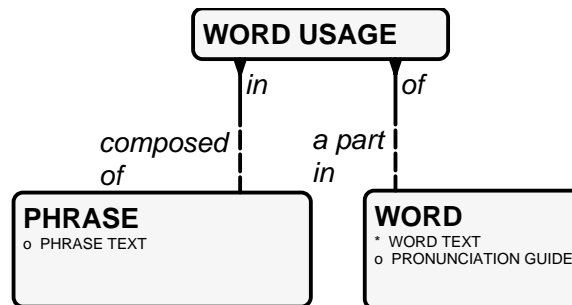
Copyright © 2005, Essential Strategies, Inc.

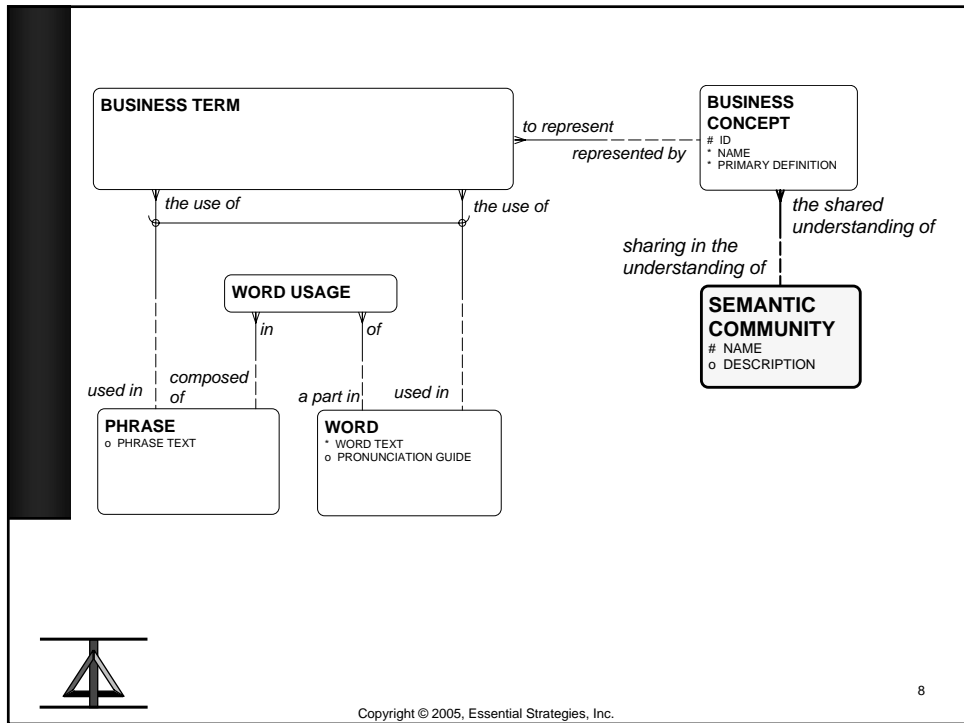
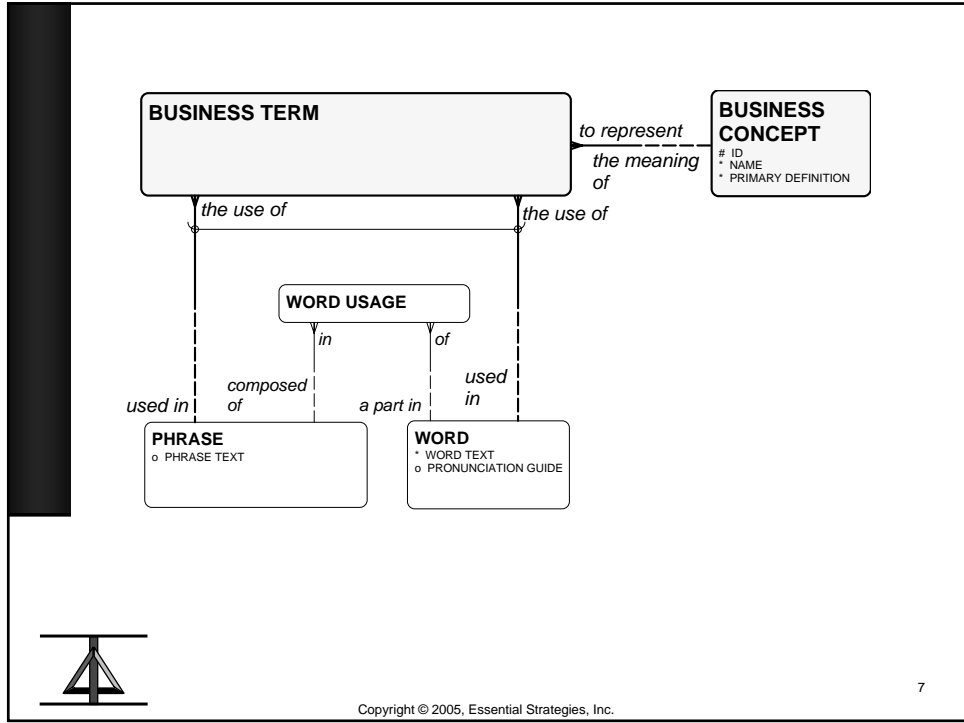
4

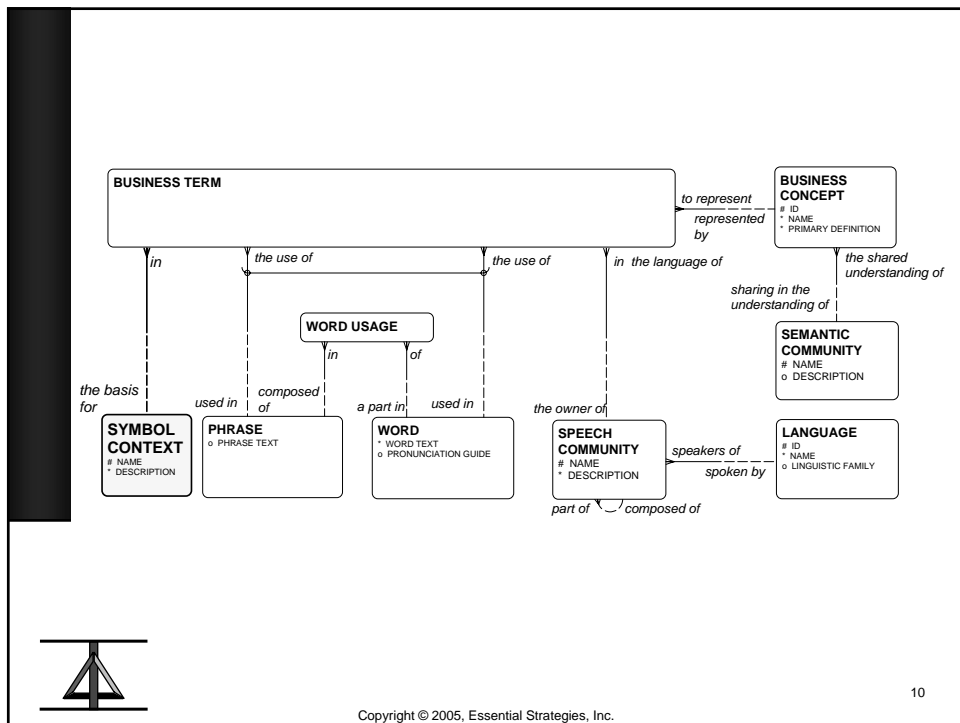
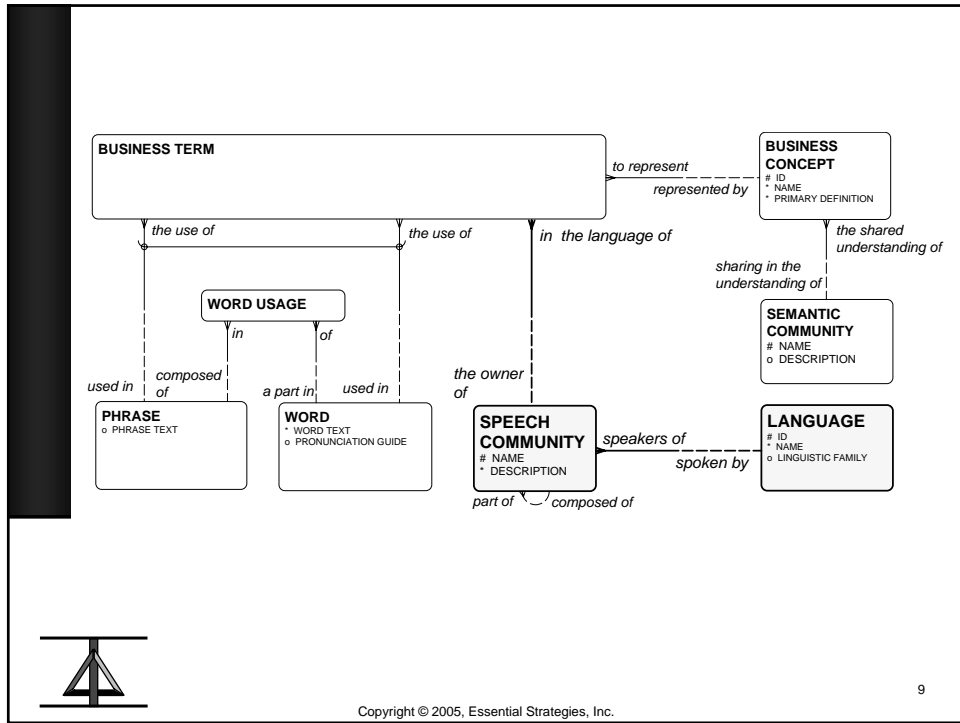
Row Two: The business owner's View

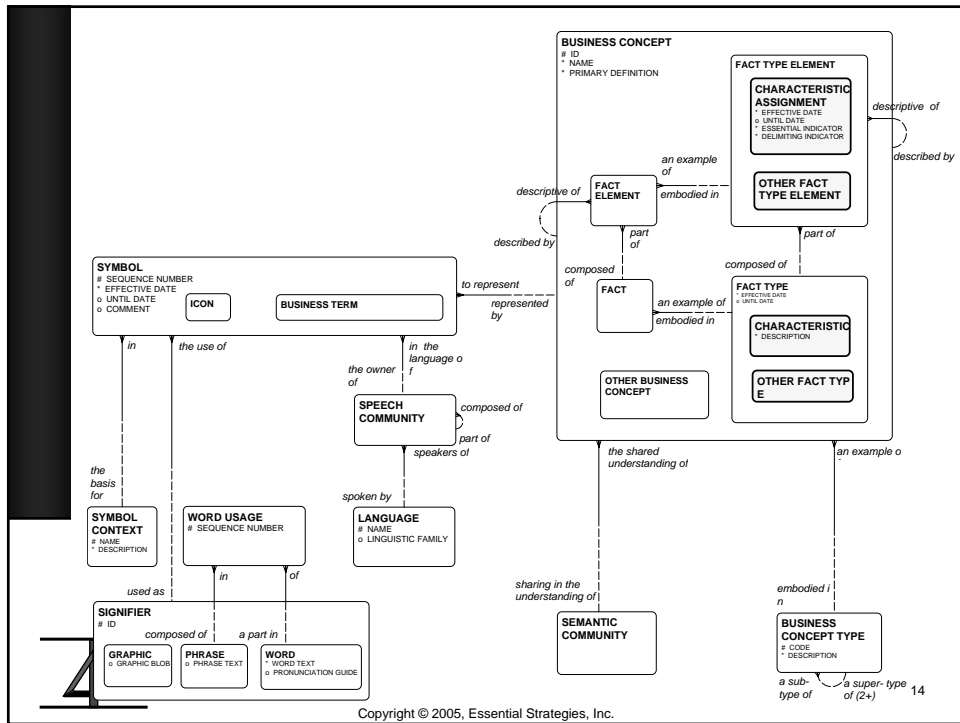
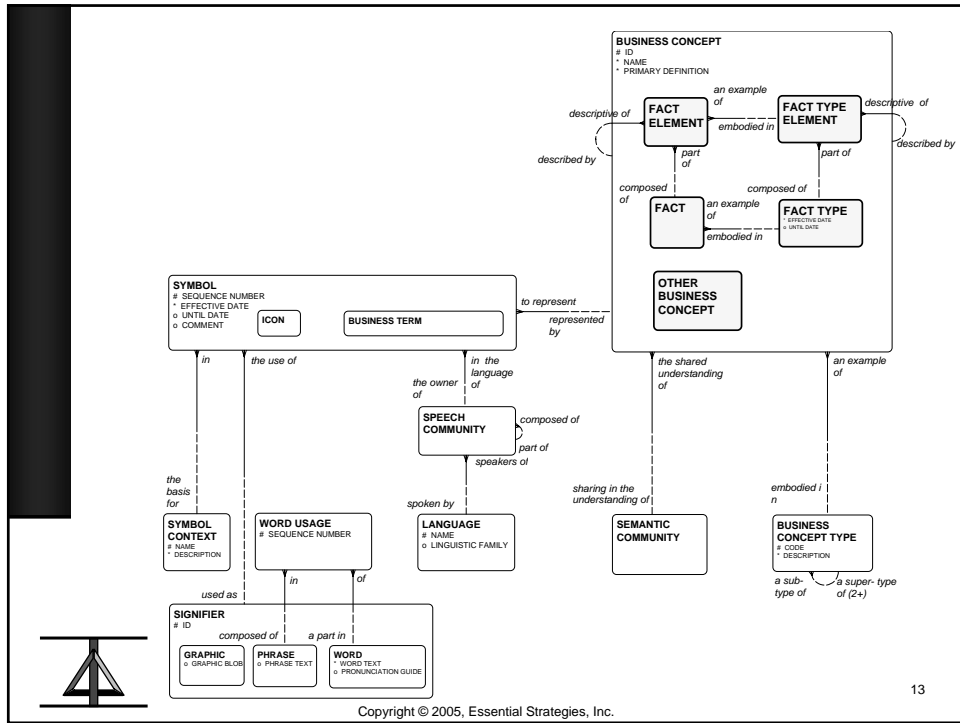


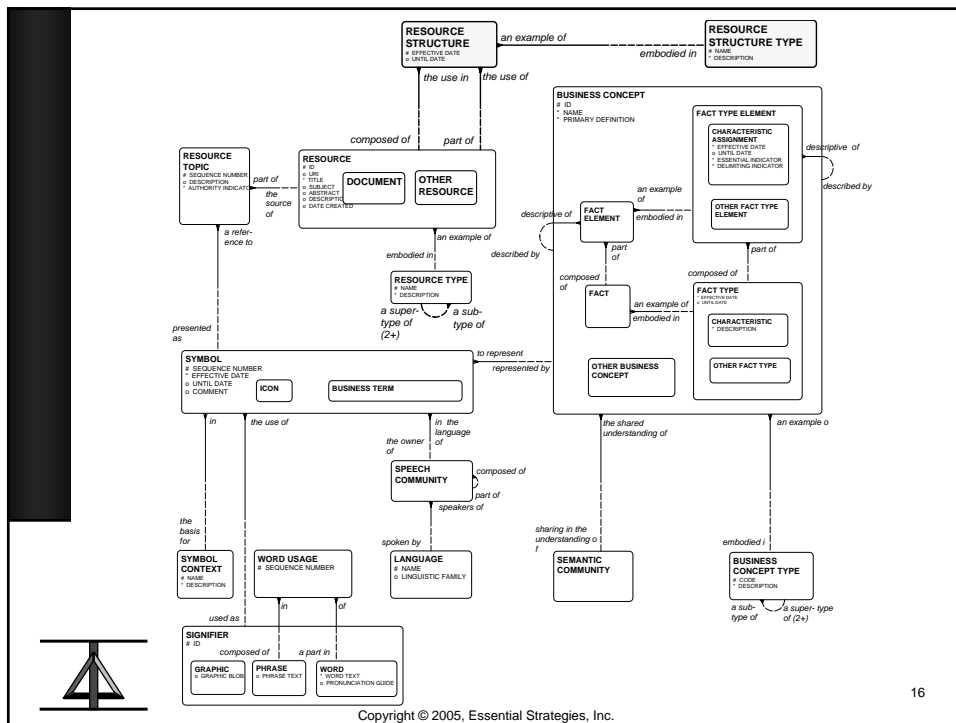
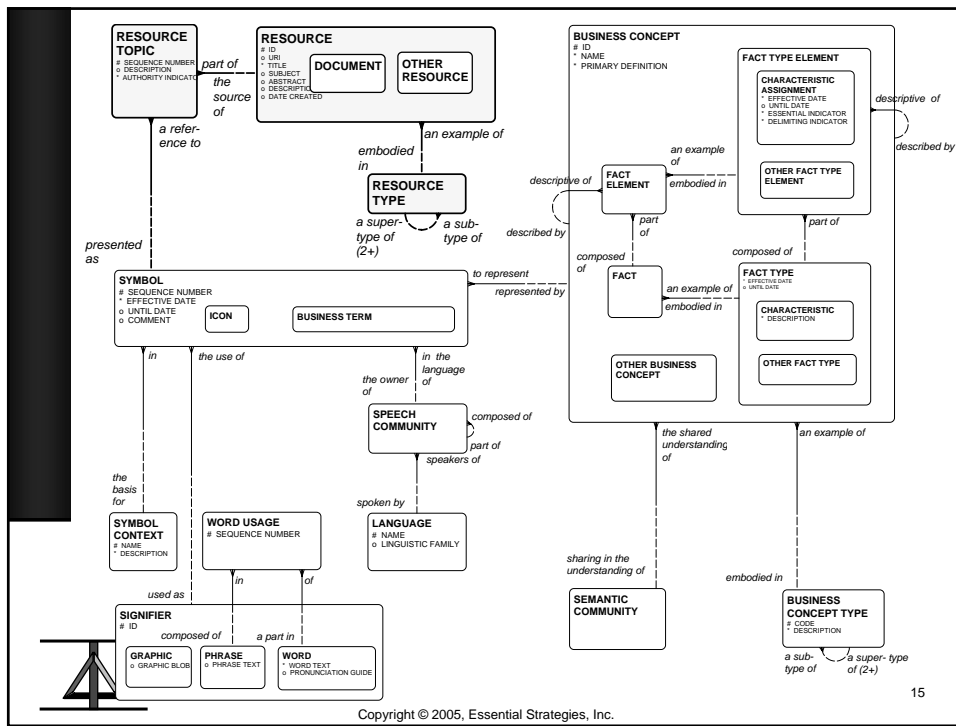
WORD USAGE

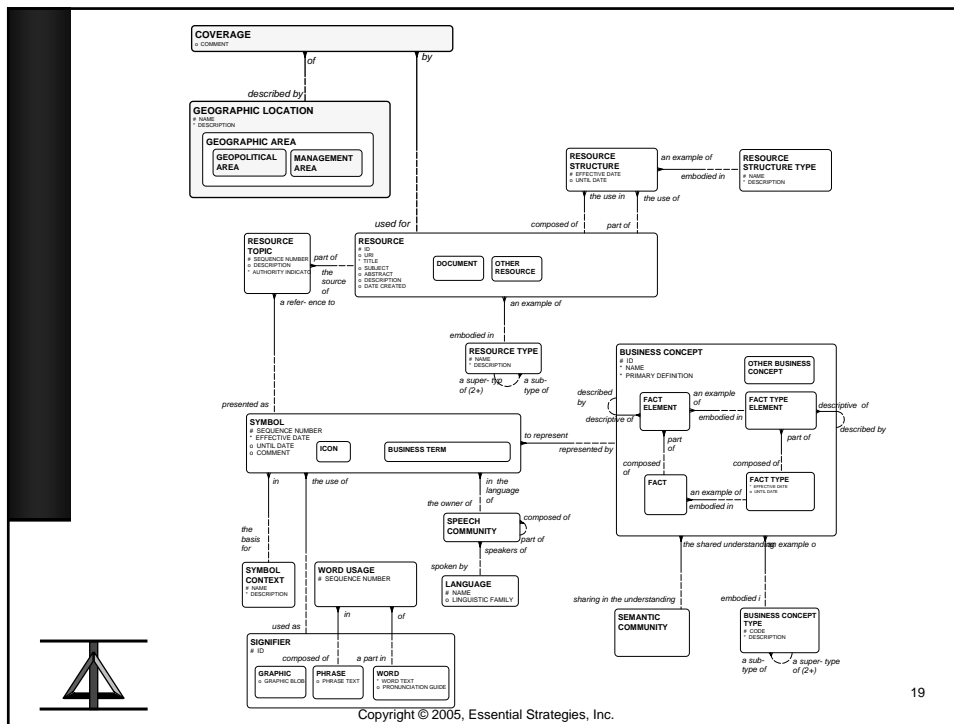








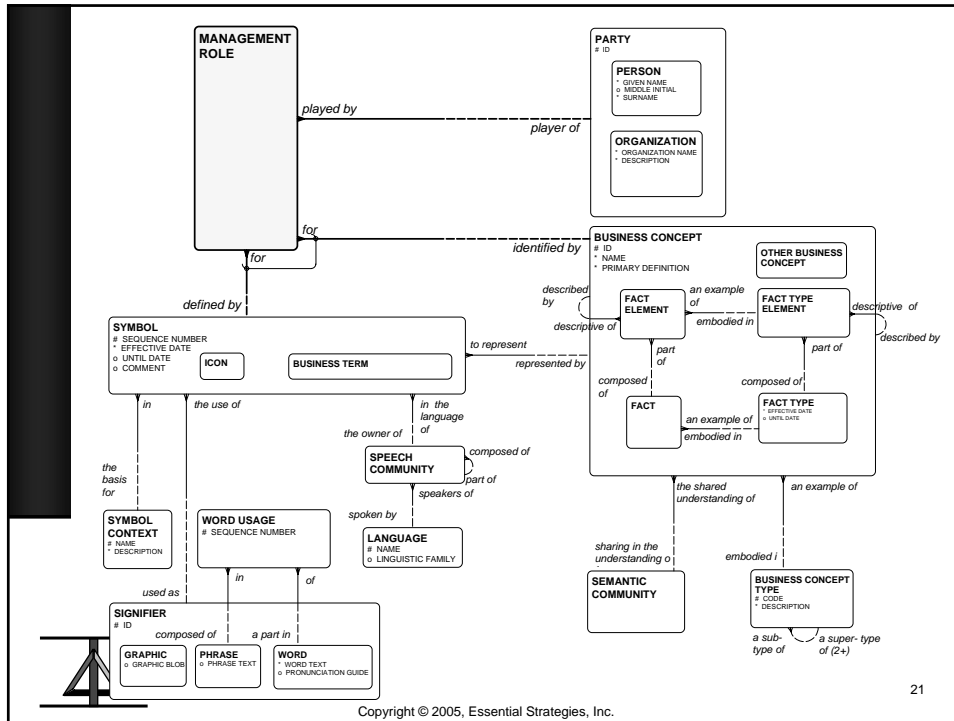




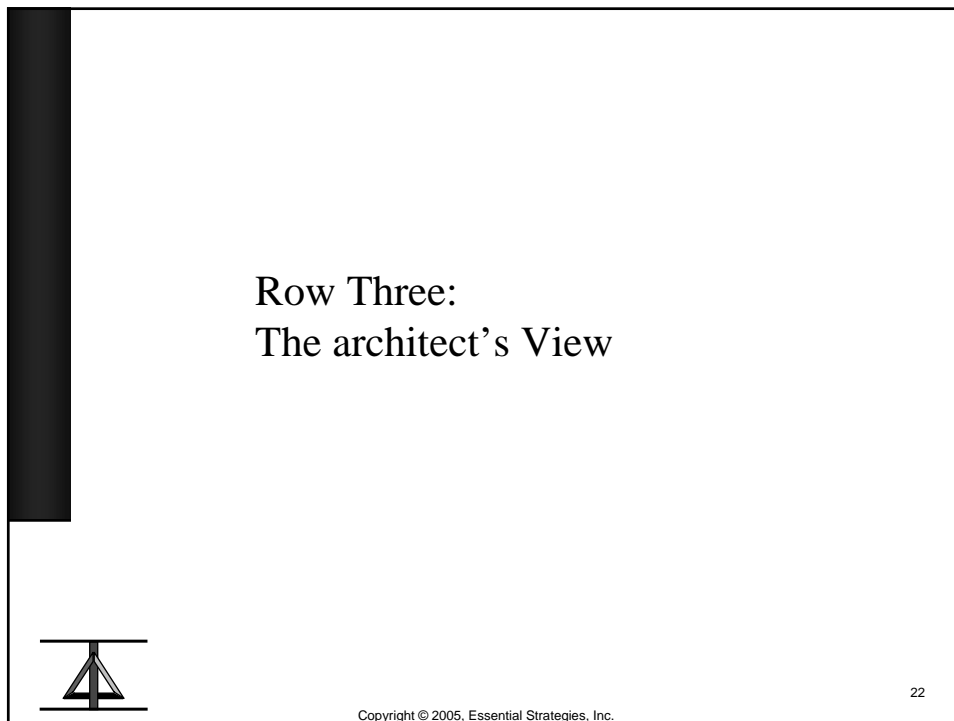
The Dublin Core . . .

Dublin Core Term	Metadata Element
Contributor	<i>created via authorship role/ an example of authorship role type "Contributor"</i>
Coverage	<i>to describe an aspect of coverages / of a geographic location.</i>
Creator	<i>created via authorship roles/ an example of authorship role type "Creator"</i>
Date	resource: "Date Created"
Description	resource: "Description"
Format	<i>embodied in copy; "Format"</i>
Identifier	resource: "ID"; resource: "URI"
Language	<i>the source of resource elements/ a reference to a symbol/ in the language of a speech community/ speaker of a language_</i>
Publisher	<i>created via authorship roles/ an example of authorship role type "Publisher"</i>
Relation	<i>part of a resource structure/ the use of another resource</i>
Rights	<i>created via authorship roles/ an example of authorship role type "Rights holder"</i>
Source	<i>part of resource structure/ an example of resource structure type "Source"</i>
Subject	resource: "Subject"
Title	resource: "Title"
Type	resource type: "Name"

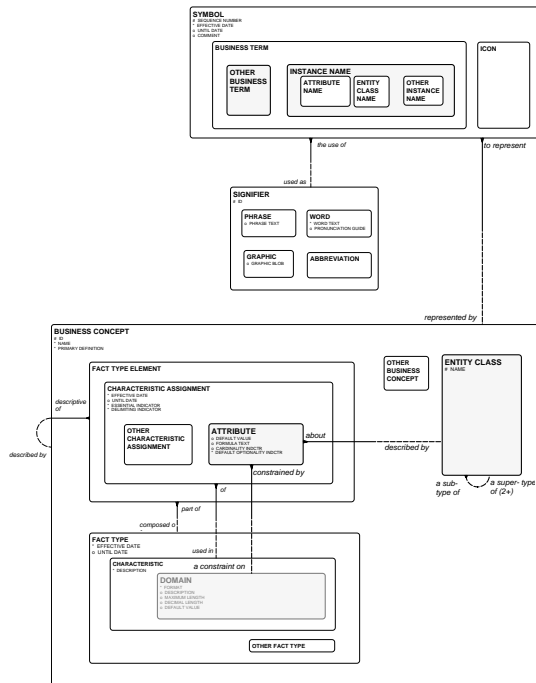
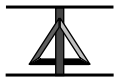
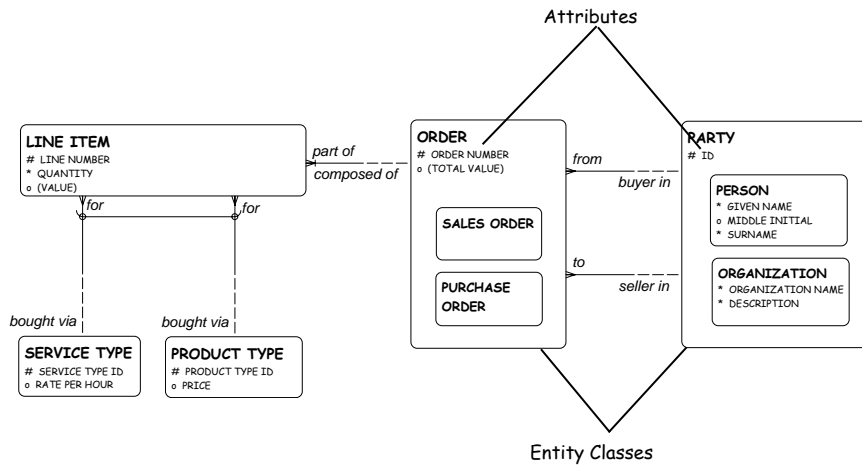
* Read "I" as "that is (are)..."



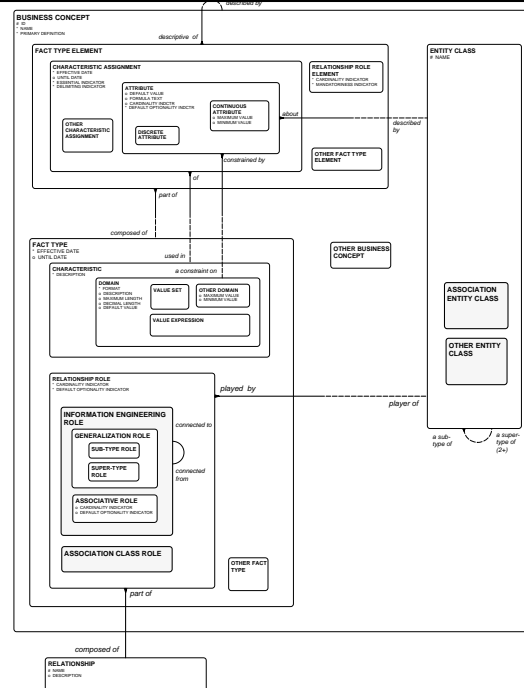
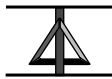
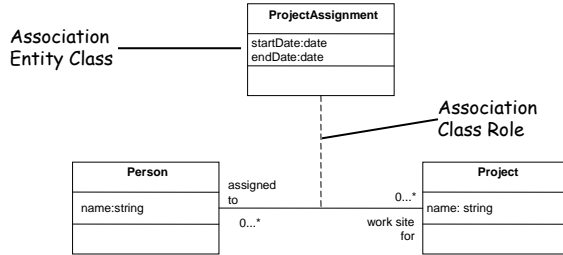
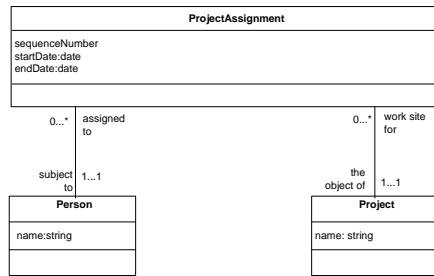
Row Three: The architect's View

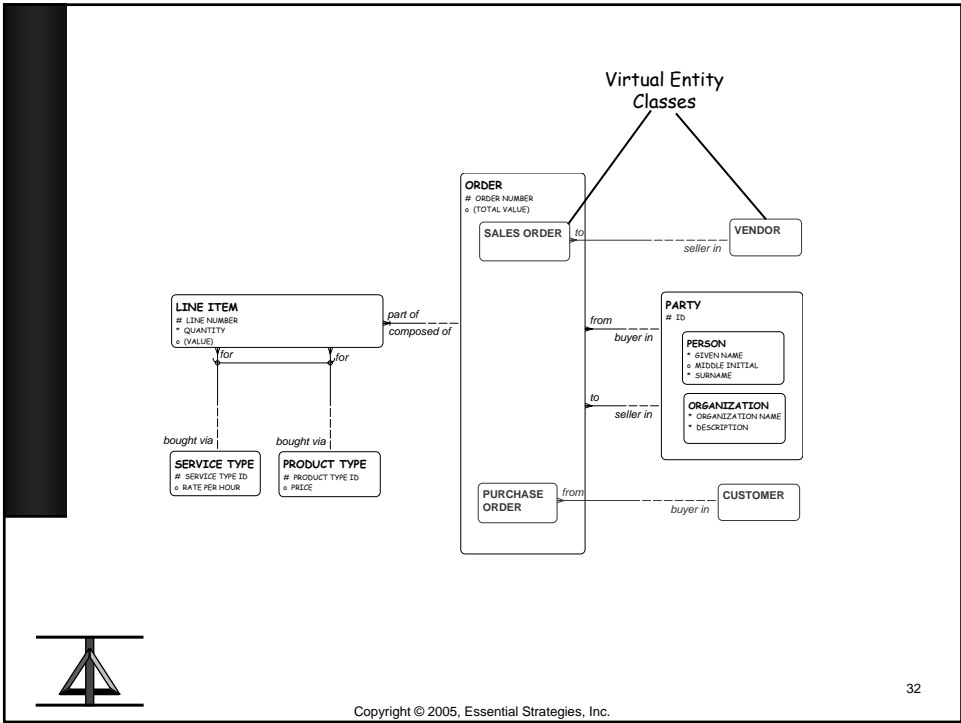
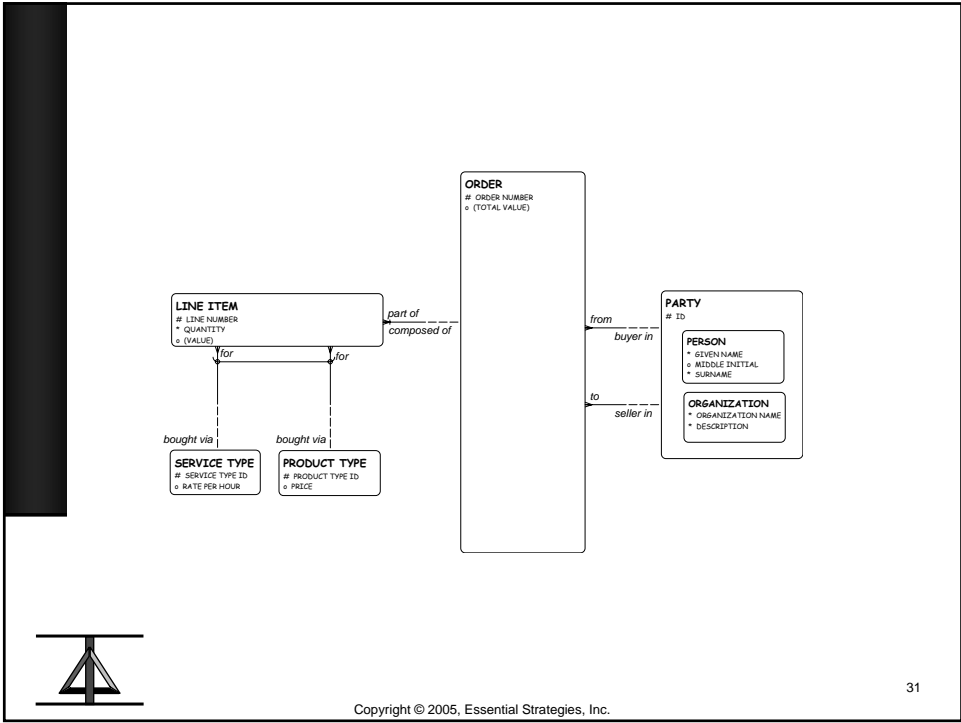


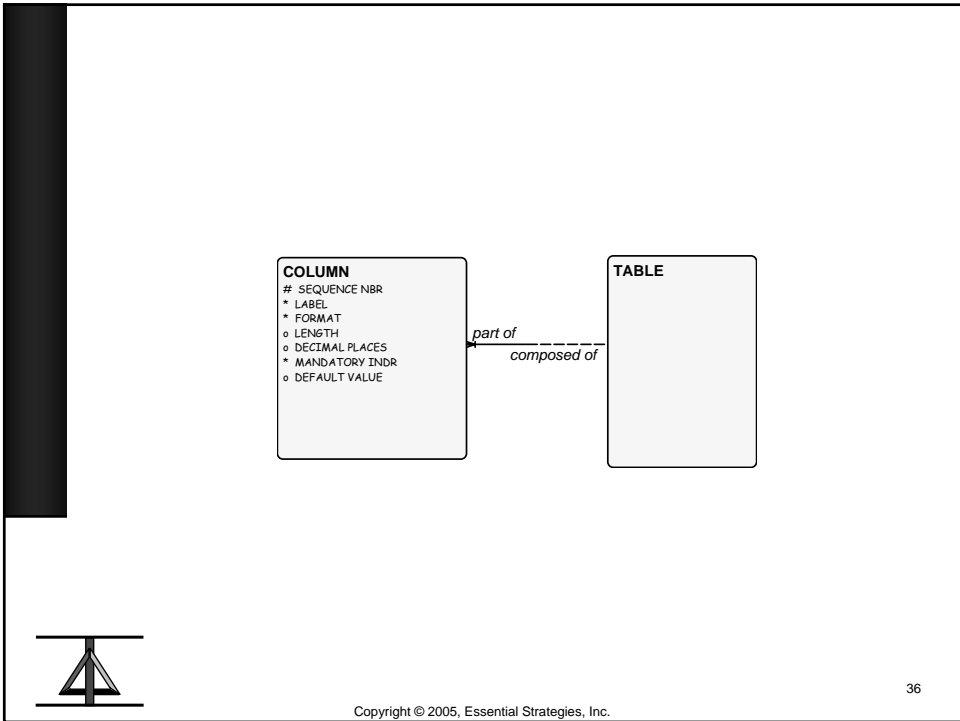
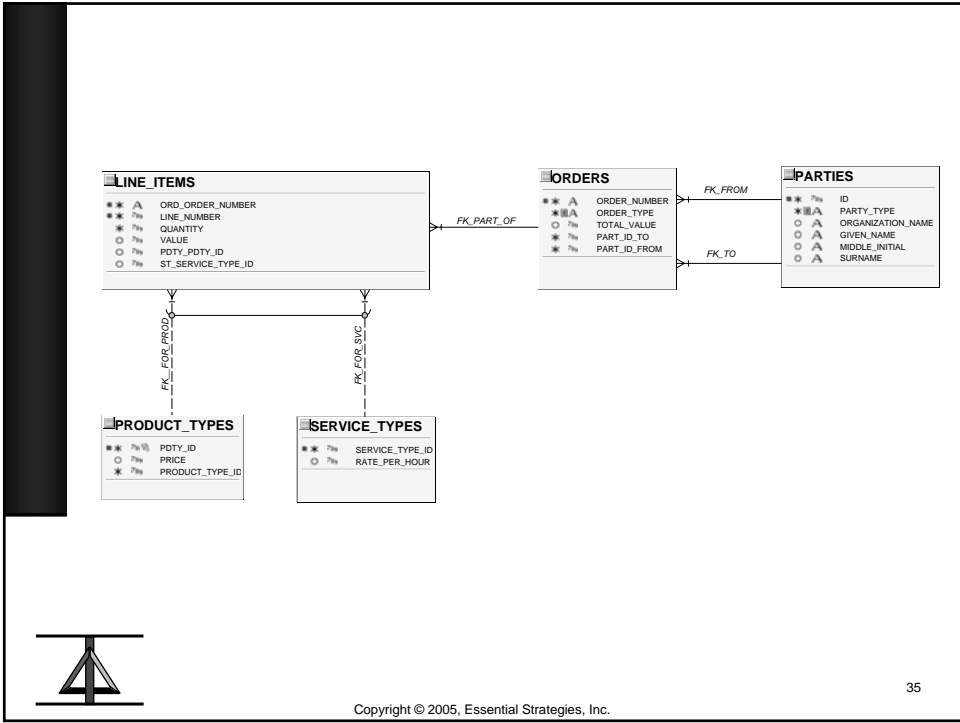
Sample Entity / Relationship Diagram . . .



Sample Intersect
Entity Classes . . .

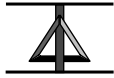






A View (in SQL) . . .

```
CREATE VIEW CUSTOMER AS
SELECT P.ID,P.PARTY_TYPE,P.ORGANIZATION_NAME,P.
      GIVEN_NAME,P.MIDDLE_INITIAL, P.SURNAME
FROM ORDERS O, PARTIES P
WHERE O.PART_ID_TO = <the 'party id' of our company>
      AND O.PART_ID_FROM = P.ID
      AND O.PART_ID_FROM <> <the 'party id' of our company>
```



**VIEW SELECTION
CONDITION**
SEQUENCE NUMBER
OPERATOR
CONSTANT

**COLUMN
SELECTION**
EFFECTIVE DATE
UNTIL DATE

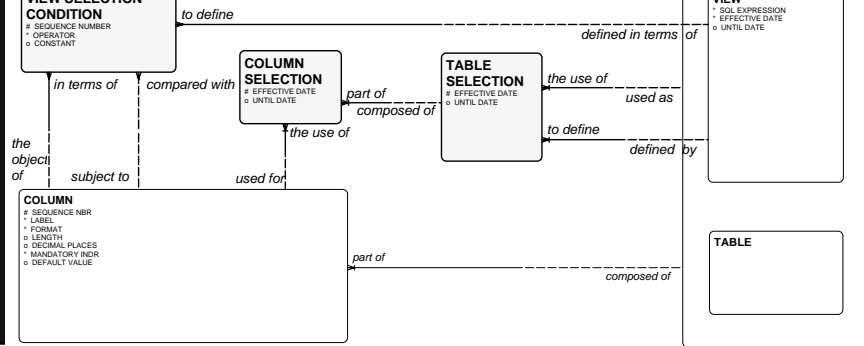
**TABLE
SELECTION**
EFFECTIVE DATE
UNTIL DATE

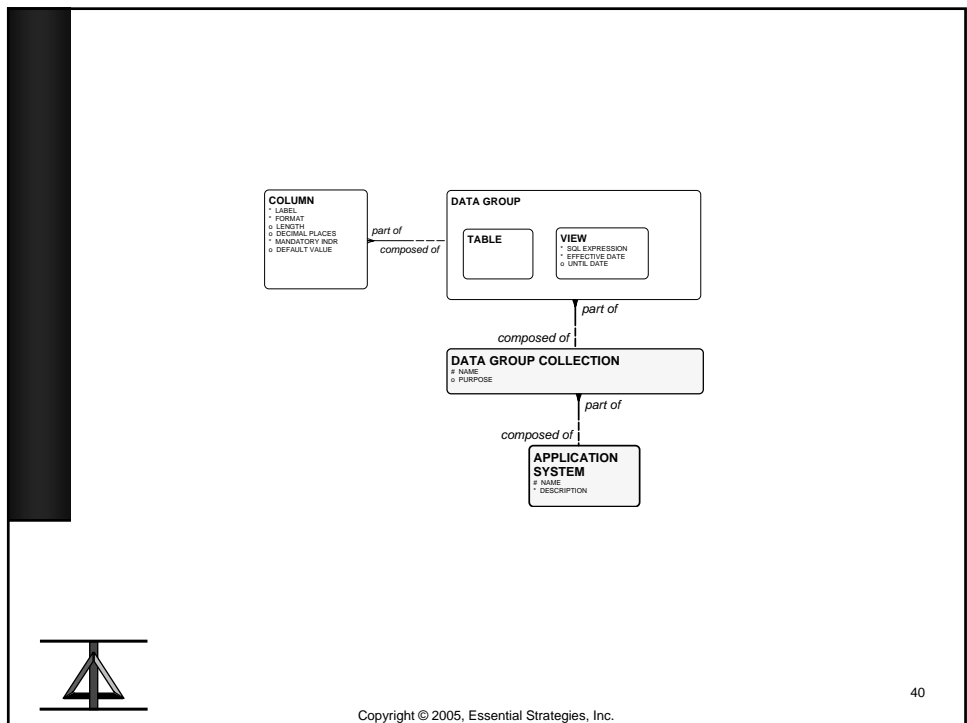
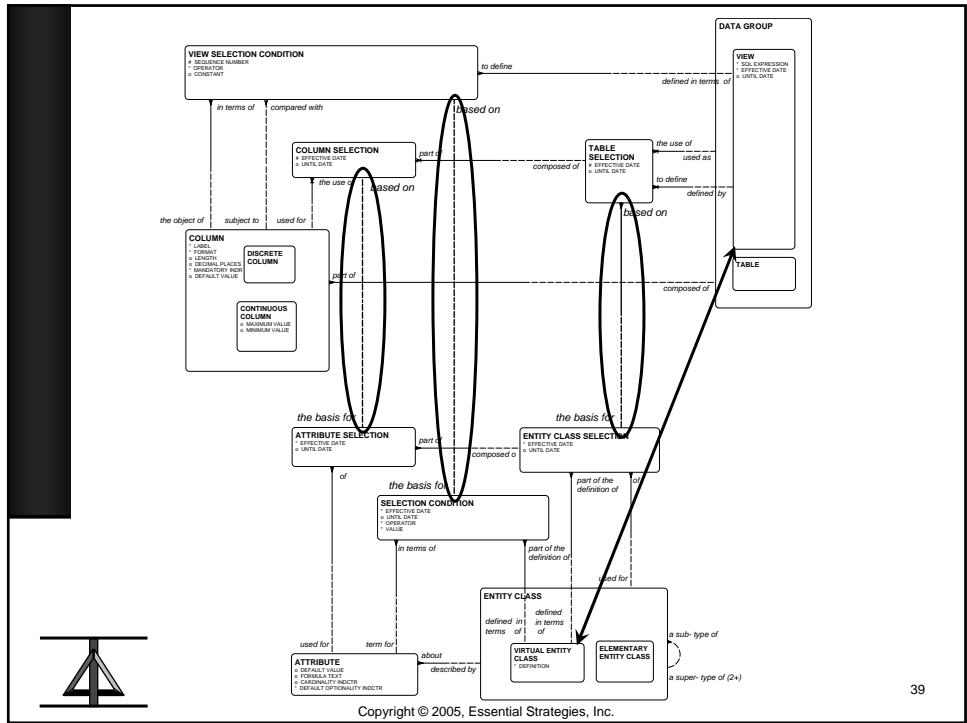
COLUMN
SEQUENCE NBR
LABEL
FORMAT
LENGTH
DECIMAL PLACES
MANDATORY INDR
DEFAULT VALUE

DATA GROUP
ID
DESCRIPTION

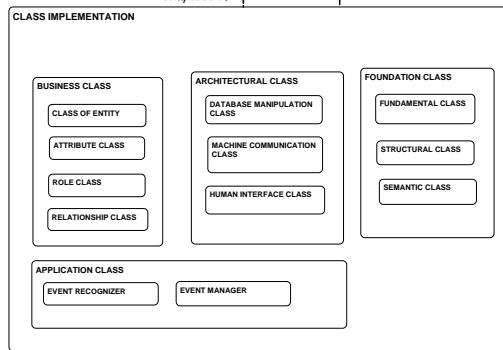
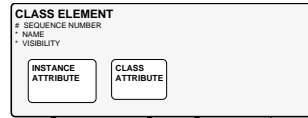
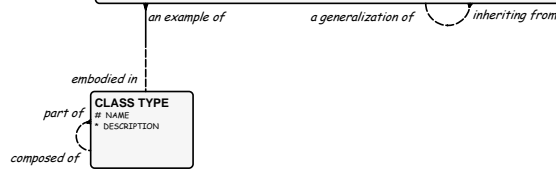
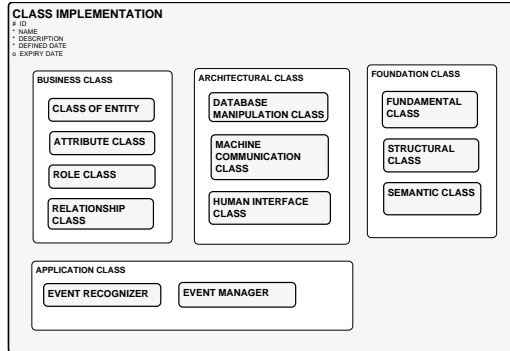
VIEW
SQL EXPRESSION
EFFECTIVE DATE
UNTIL DATE

TABLE





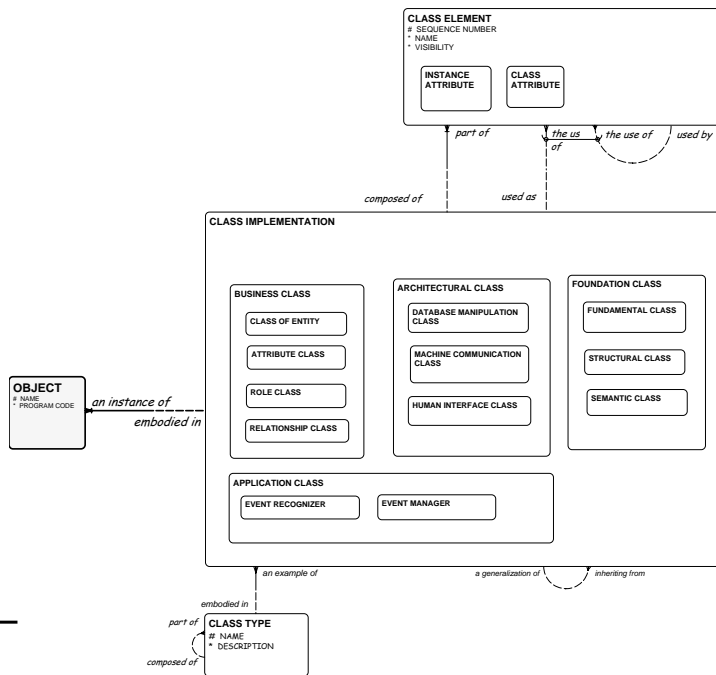
The Object-oriented Version . . .

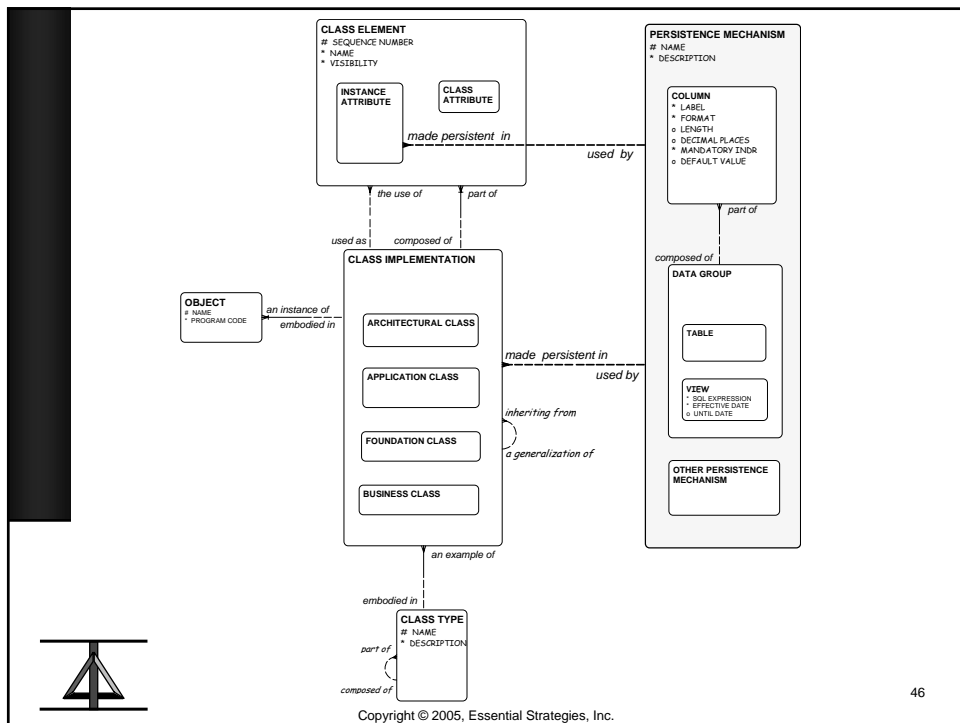
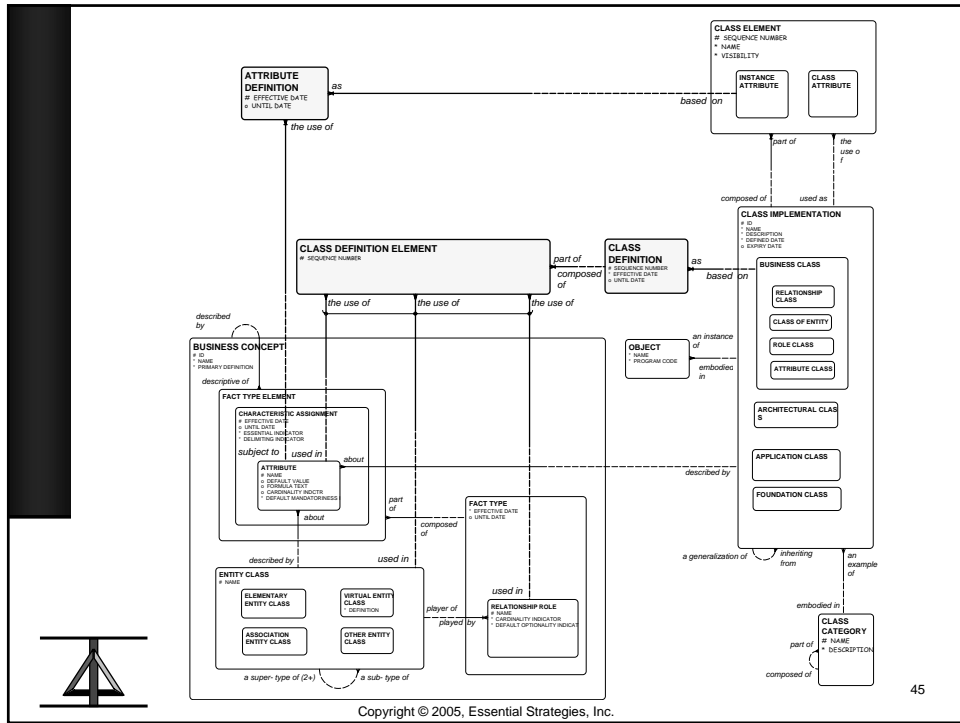


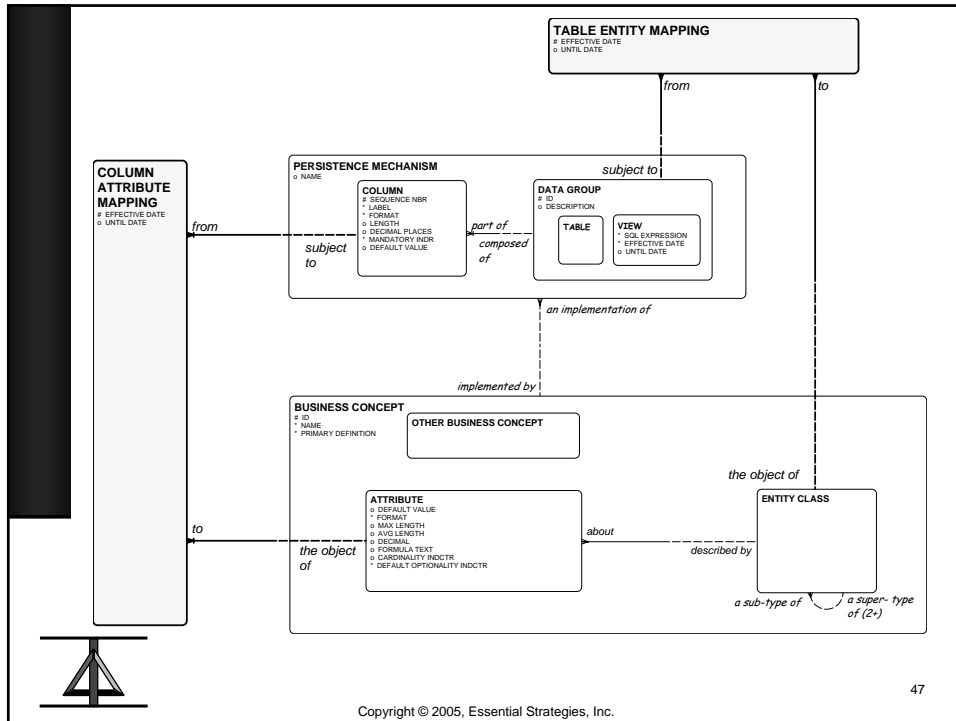
An object-oriented program . . .

```

Hominoid
New: Hominoid
// creates and returns a new instance of Hominoid
    
```

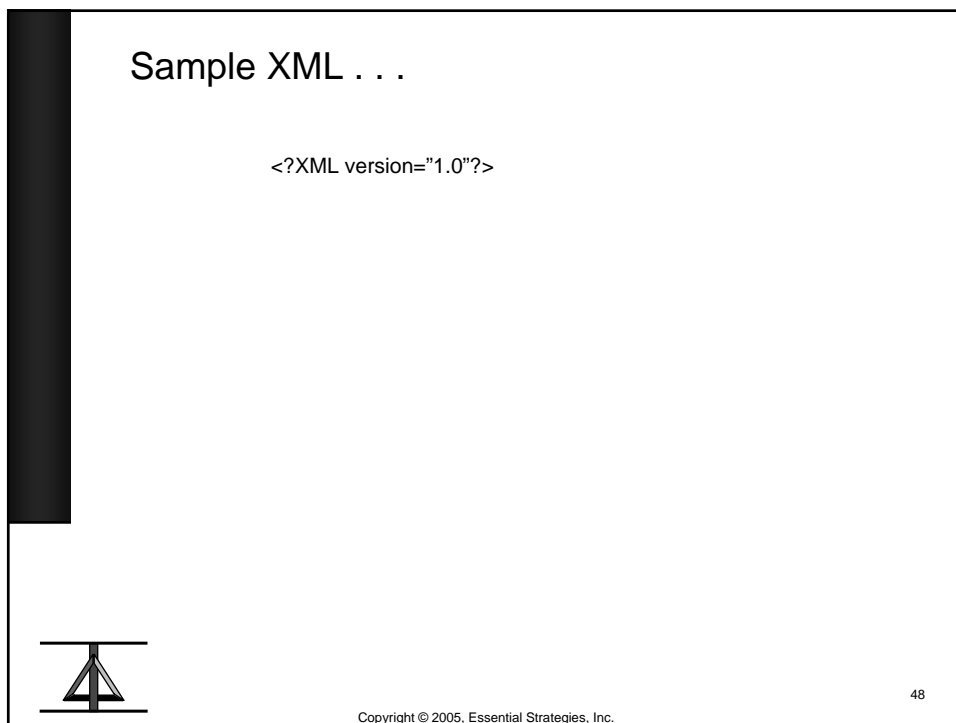






Sample XML . . .

```
<?XML version="1.0"?>
```



Sample XML . . .

```
<?XML version="1.0"?>
<!--      **** Basket ****  -->
<PRODUCT>
</PRODUCT>
```



Sample XML . . .

```
<?XML version="1.0"?>
<!--      **** Basket ****  -->
<PRODUCT>
  <product_id>98756</product_id>
  <product_name>basket</product_name>
  <unit_of_measure>each</unit_of_measure>
</PRODUCT>
```

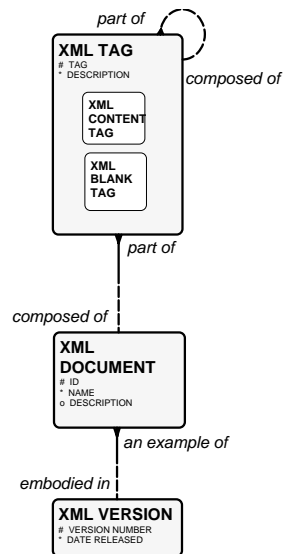
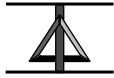


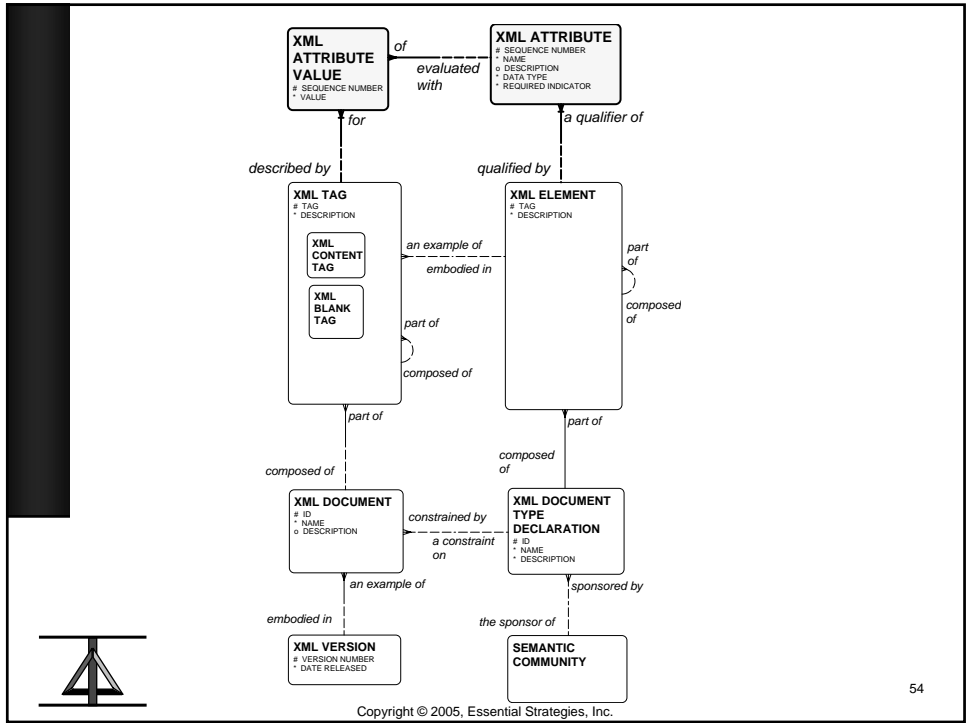
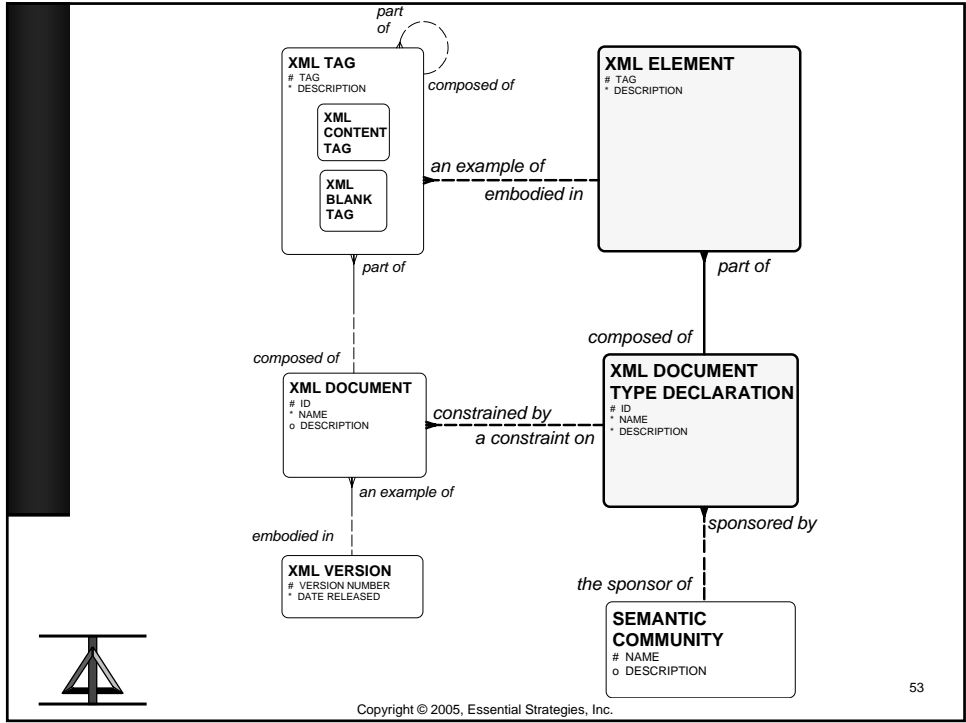
Sample XML . . .

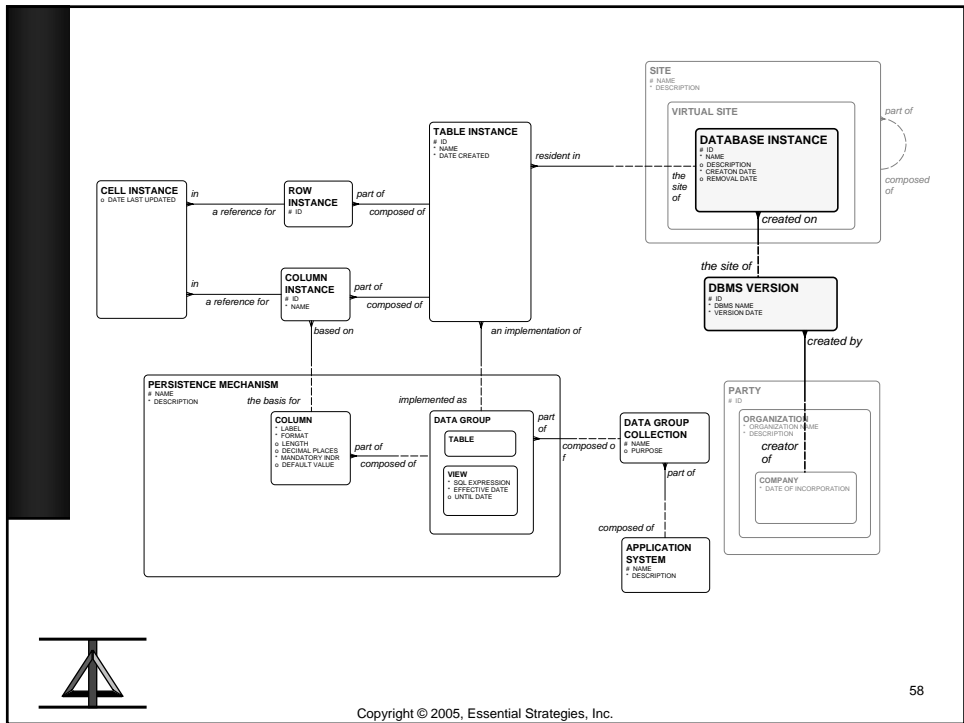
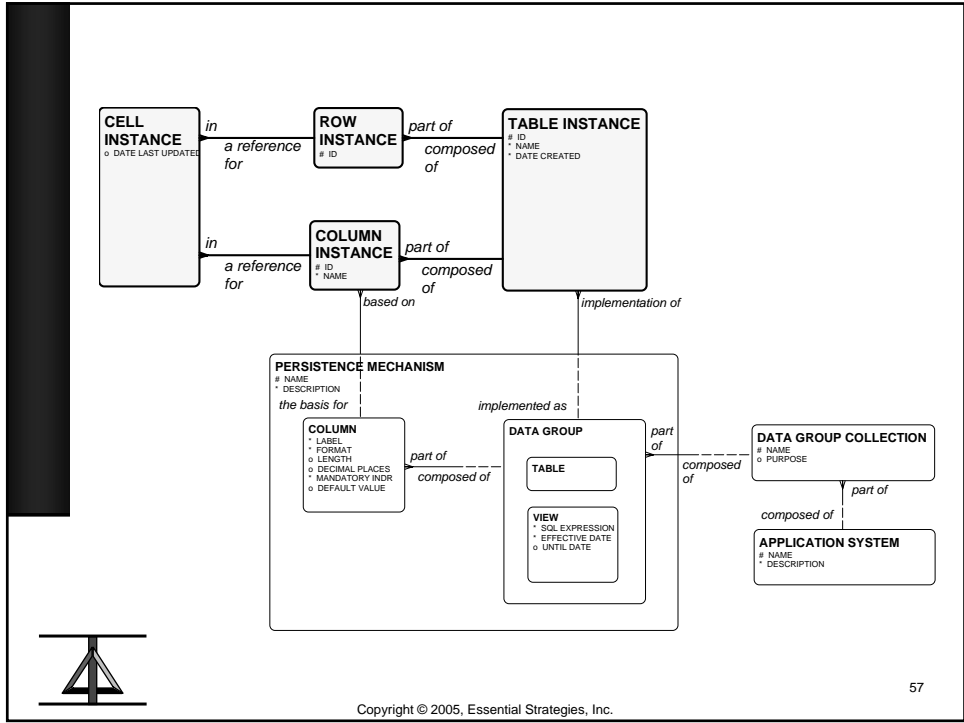
```

<?XML version="1.0"?>
<!-- **** Basket **** -->
<PRODUCT>
  <product_id>98756</product_id>
  <product_name>basket</product_name>
  <unit_of_measure>each</unit_of_measure>
  <specification>
    <variable>color</variable>
    <value>blue</value>
  </specification>
  <specification>
    <variable>size</variable>
    <value>large</value>
  </specification>
  <specification></specification>
  <specification/>
</PRODUCT>

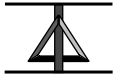
```







... to be continued ...



Copyright © 2005, Essential Strategies, Inc.

59