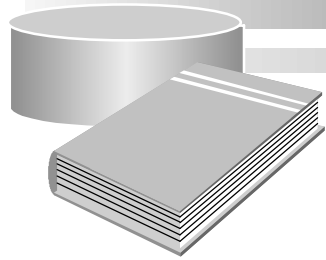


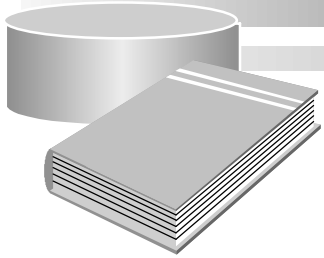
Reusing Information: Database Publishing and Other Methods

Yvonne DeGraw
DeGraw Technical Services
805-683-5784
yvonne@silcom.com



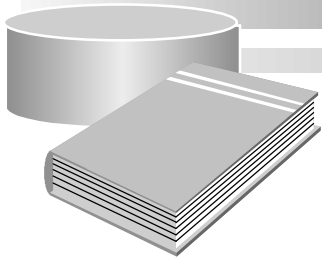
Outline

- Why Reuse Information?
 - Terminology and Background
 - What Information to Reuse
 - Methods for Reusing Information
 - Case Studies
-
-



What is “Reuse”?

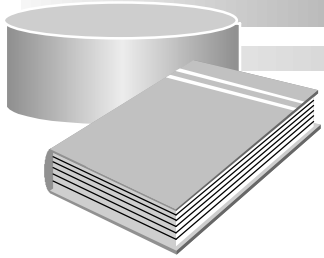
- Opposite of “re-creating” or duplicating information
- Goal is to create documents so that when one piece of information changes, you make only one change to the document set.
- Reuse = Intelligent Planning + Creative Laziness



Why Reuse Information?

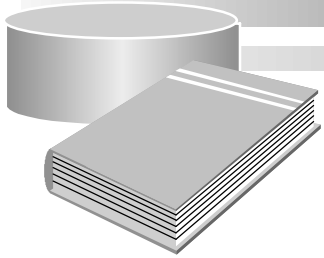
■ Problems you can solve...

- Multiple hardware platforms
- Multiple output formats
- Multiple configurations
- Changes happen late in product cycle
- Users need multiple views of same information
- ???



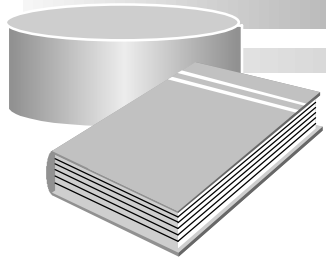
Why Reuse Information?

- Improvements you can make
 - Better accuracy and self-consistency
 - Easier maintenance of information
 - Easier to change formatting
 - Better document structure
 - Be proactive
 - Save time (sometimes)



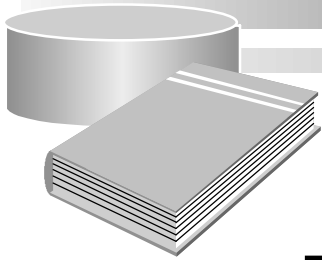
Publishing Tools

- FrameMaker
 - Interleaf
 - SGML
 - Microsoft Word
 - Corel Ventura Publisher
 - ...
-
-



Terminology

- Information
 - Information Function
 - Formatting
-
-



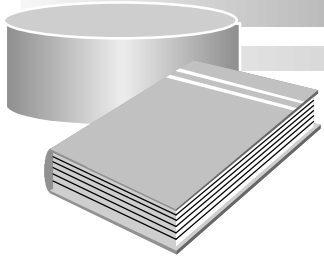
Isolating Information

■ The SGML Model

Information & Function	Formatting
<pre><head>CREATE Command </head> <index>CREATE command </index> <P>The CREATE command ... <\P></pre>	<pre><head> = Bold, 18 pt. <index> = Helv, 10 pt. <P> = Times, 12 pt.</pre>

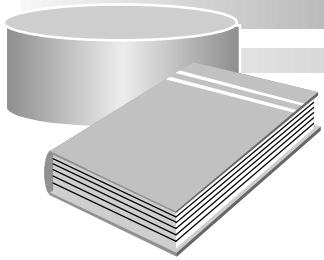
■ Model for Information Reuse

Information	Function	Formatting
<pre>Commands: CREATE ...</pre>	<pre>Use CREATE as: Heading Index Entry Text</pre>	<pre>heading = Bold, 18 pt. index = Helv, 10 pt. text = Times, 12 pt.</pre>



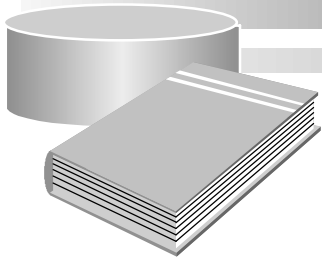
What Information to Reuse

- Any structured information ...



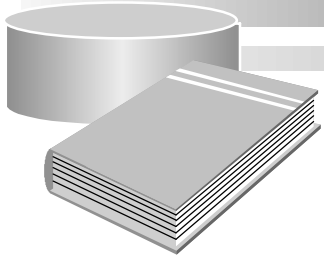
What Information to Reuse

- catalogs
- directories
- rate sheets
- schedules
- part lists
- financial statements
- address lists



More Information to Reuse

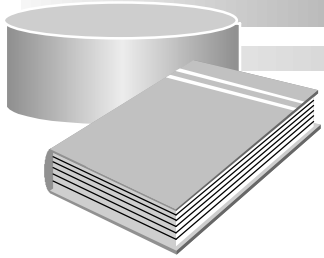
- product names
 - commands and other syntax
 - function or routine libraries
 - object-oriented software
 - procedures
 - error messages
 - lists (datatypes, system variables...)
 - topic --> topic relationships (hyperlinks)
 - . . . any information that is currently duplicated
-
-



Where to Reuse Information

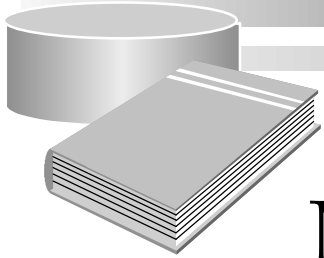
- Any place it is currently duplicated...

- Any place you would have liked to use it but didn't want to duplicate it...



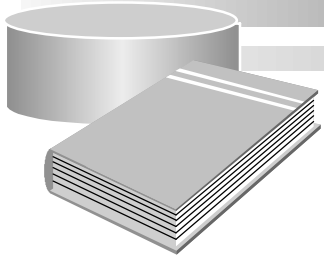
Where to Reuse Information (Output formats)

- Detailed reference documents
- Quick reference documents
- Online documents and online help
- Tables, lists, and matrixes
- WWW pages or gopher site
- Inside the software
- Database for user access



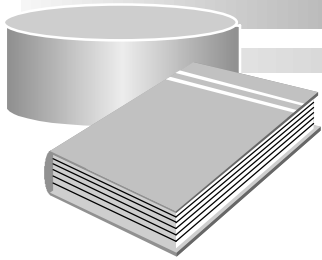
Methods for Reusing Information

- ① Using variables and cross-references
 - ② Storing information in topic files
 - ③ Storing graphics in separate files
 - ④ Using conditional processing
 - ⑤ Using automatic conversions
 - ⑥ Using a database
-
-



Method 1: Variables and Cross-References

- What terms are used many times and may change?
 - product name
 - product-specific terms
 - version number
 - book titles
 - chapter and section names and numbers
 - Define once and reuse.
 - If name changes, update in one place.
 - Be consistent.
 - If section names are names of items, reuse cross-refs
-
-



Variables and Cross-References

■ Example:

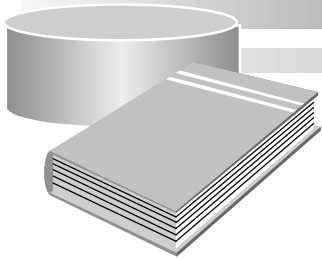
– With `<Product_name> <version#>`, you can do just about anything.
In fact, `<Product_name>`'s kitchen sink feature, lets you...

- » pre-release “SmartStar Windows”
- » V1.0, V1.1 “SmartStar Vision”
- » V1.2 “Sapiens Vision”
- » V1.3 “Sapiens Ideo”
- » V1.4 ???

■ Watch out for “a” vs. “an”.

■ Watch out for plural vs. singular -- verb agreement.

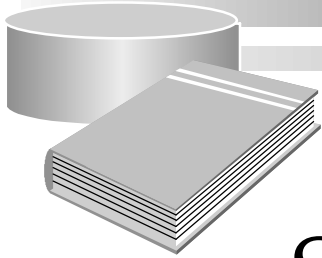
■ Watch out for formatting.



Variables and Cross-References

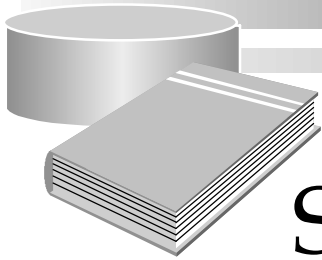
■ How to ...

- FrameMaker: variables (keep in template file in case changes), cross-references
- Interleaf: define in catalog (shared across documents)
- SGML: reference, symbol, or similar tags
- Microsoft Word: AutoText field and other fields
- Ventura Publisher: variable text fields



Method 2: Storing Information in Topic Files

- Keep document sections in small files; use “by reference”
- How to reuse:
 - Reuse subsets of topic files
 - Reuse files in different orders
 - Reuse a file in several places
- What is in a topic file?
 - Files may be long or short
 - Files may use other techniques (such as conditions and variables)
 - Files may contain formatting information
- Name files consistently (may want to combine with database techniques)

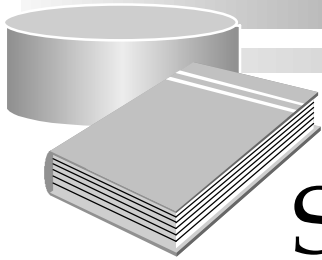


Storing Information in Topic Files

■ Examples:

- SAPIENS, SmartStar, SmartGL, and SMARTSTAR are registered trademarks, and SAPIENS Ideo, NoGL, ObjectPool, Object Modeler, ReportPainter, and SmartObject are trademarks of <reference>(COMPANY_NAME).

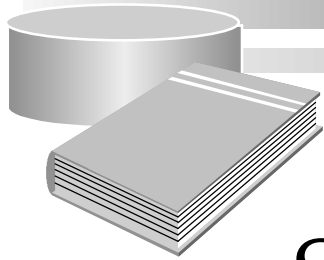
 - <x>(STRICT_TRANS switch<xs><reference>(STATEMENT))
The STRICT_TRANS switch determines whether all records are rolled back if any record cannot be <reference>(ACTIONPAST) or you want to be able to <reference>(ACTION) more than 99 records (without data protection).
 - » Statement: DELETE, INSERT, UPDATE, transaction
 - » Action: delete, insert, update, modify
 - » ActionPast: deleted, inserted, updated, modified
-
-



Storing Information in Topic Files

■ How to ...

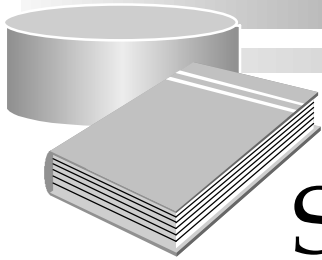
- FrameMaker: Publish & Subscribe on Mac
separate MIF files (or MML)
cross-references to longer text
API coding
- Interleaf: catalog (shared components)
- SGML: easy in most DTDs
- Microsoft Word: Includetext field, document links, master documents, DDE, OLE
- Ventura Publisher: text always imported by reference



Method 3: Storing Graphics in Separate Files

- Keep graphics in separate files
- Use graphics “by reference”
- Update graphics in one place only

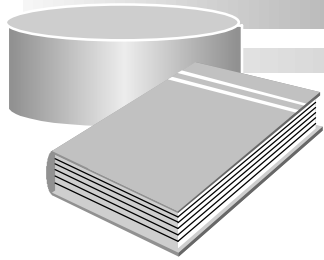
- Name files consistently (may want to combine with database techniques)



Storing Graphics in Separate Files

■ How to ...

- FrameMaker: include by reference
- Interleaf: shared frames
- SGML: specify filename or item in lookup table in tag
- Microsoft Word: insert picture with “link to file”
- Ventura Publisher: graphics always imported by reference

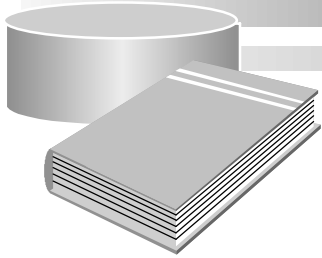


Method 4: Using Conditional Processing

- Include text based on condition
- Create multiple documents from one document
- Example:
 - When you click this button, the window changes to a small “icon” and moves to the edge of your screen. You can also minimize a window by pressing

```
<condition>(MOTIF)  
< Alt + F9 >.  
<endcondition>
```

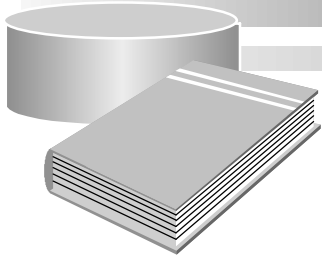
```
<condition>(PC)  
< Alt + Spacebar > and typing “N” to select “Minimize” .  
<endcondition>
```



Using Conditional Processing

- May want to create or set multiple conditions
- Some tools let you see effects interactively
- Some tools allow setting conditions for part of document

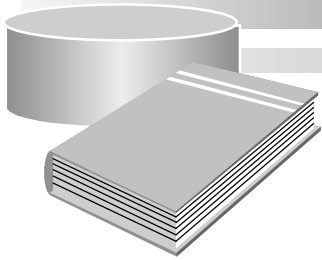
- What types of conditions might you need to set?



Using Conditional Processing

■ Uses for conditions:

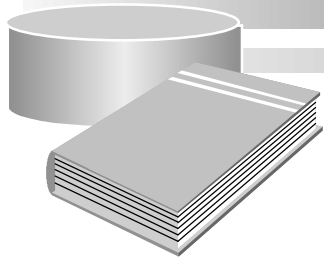
- Version control
- Print/online differences
- Platform differences
- Internal comments
- Different product configurations
- Reusing short sections that need to be slightly different
- ...



Using Conditional Processing

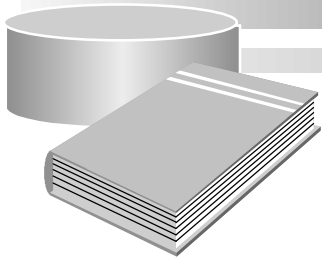
■ How to ...

- FrameMaker: conditions
- Interleaf: effectivity
- SGML: recommended method is for tags to understand conditions
- Microsoft Word: “if” field
- Ventura Publisher: set color for a text type to white or FontSize to 0



Method 5: Using Automatic Conversion

- Not true reuse; actually duplication
- Most conversions do 80-90% of work; better than 0%
- Use to get to output formats
 - PostScript, RTF, HTML, ASCII text . . .
- Use the Internet to find solutions for specific needs

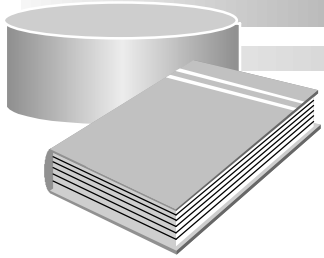


Using Automatic Conversion

■ How to ...

- FrameMaker: `comp.text.frame`
`framers@uunet.uu.net`
- Interleaf: `comp.text.interleaf`
- SGML: `comp.text.sgml`
- Microsoft Word: `comp.os.ms-windows.apps.word-proc`
- Ventura Publisher: `???`

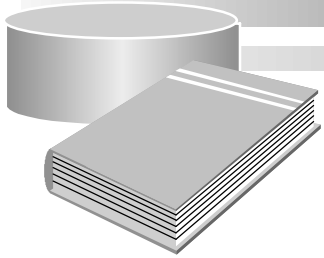
■ First, check the FAQ if there is one.



Method 6: Using a Database

■ Where to find information

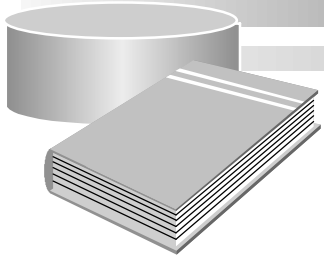
- relational database
- data files
- spreadsheet
- CASE tool
- specifications
- documents
- ...



Using a Database

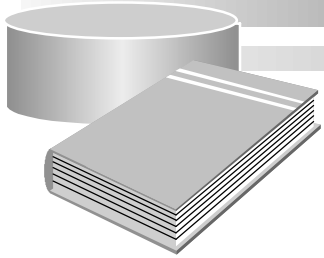
■ What can act as a “database”?

- Relational database
- Object-oriented database
- Spreadsheet
- CASE tool
- formatted data file



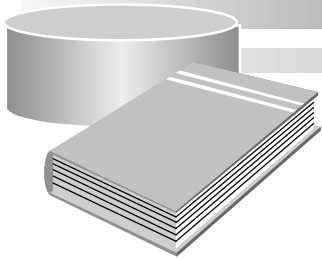
Creating a Database

- ❶ Analyze information
- ❷ Create mock-up of formats you want
- ❸ List tables
 - functions, arguments, status codes, relationships
- ❹ List fields for each table
 - arguments: name, datatype, length, optional?, read/write, version, ...
- ❺ Create tables in database (helps to have SQL experience)



Creating a Database (cont.)

- ⑥ Add test data to database
- ⑦ Create documents using test data
- ⑧ Modify database and document generation as needed
- ⑨ Add all data to database
- ⑩ Create process documents
 - How data is structured
 - How documents are generated
 - How to maintain data
 - How to create documents



Creating a Database

■ Example:

Class

c_name
c_version c_public parent default_attr short_descr

Attribute

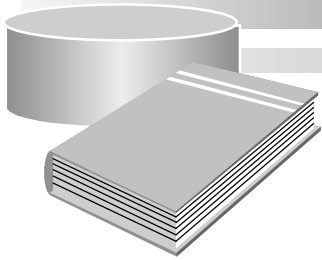
a_name
internal_name a_type a_version a_public short_descr

Class_Attr

c_name a_name
ca_version ca_public create_yn put_yn get_yn if_null in_dialog extra_descr

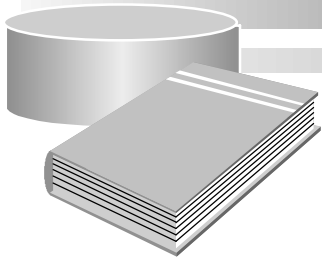
Subtopic

p_topic c_topic
category relation



Using a Database

- Use the database's report generator to create documents
 - generate some underlying tag language
 - generate formatable text (requires formatting re-work)
- Use a third-party tool that simplifies report generation
 - interactive interface to select and format data
 - questions to ask:
 - » support for if ... then ... else processing
 - » support for substituting text for coded data
 - » support for including fixed text (not from database)
 - » support for various types of sorting
- Some tools support 3GL programming

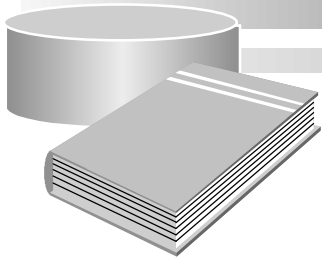


Using a Database

■ How to ...

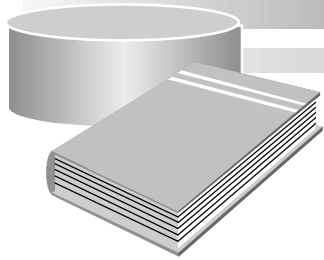
- FrameMaker: 3rd-Party: BrioPublish, UniMerge, MergeMaker
 Other: API coding, generate MIF with database
- Interleaf: 3rd-Party: SmartLeaf
 Other: LISP, generate Interleaf ASCII
- SGML: Other: Generate SGML
- Microsoft Word: Other: Generate files for mail merge or RTF
 MS Word Basic?
- Corel Ventura: Add-on: Database Editor (Database Publisher)
 Other: Generate tagged ASCII format.

■ Lots of consulting firms will set this up for you



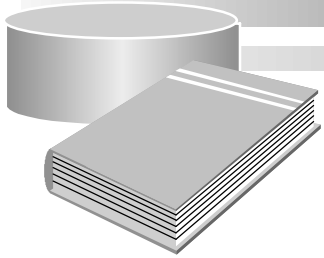
Tips and Techniques

- Document the processes you use
 - Know your publishing tools -- become your own guru
 - Get buy-in from management and/or development
 - Get buy-in from entire documentation department
 - Deal with data security issues if necessary
 - Deal with change tracking issues if necessary
 - Keep long sections of text outside databases
 - Sometimes you don't need to reuse; know when it saves time
-
-



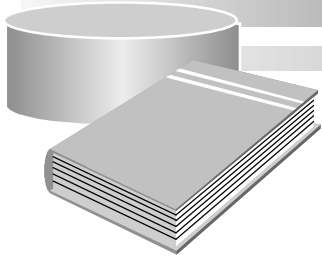
Case Study 1

- Make a list of the standard parts of a document to reuse in the multiple documents in a document set.
- Hint: start by thinking about the title page and preface.



Case Study 2

- List examples of situations in which people in your group could use conditional processing to avoid duplication of effort.



Case Study 3

- Propose a database structure for a function reference.
- Each function returns a value, but the datatype of the value varies. Each function has a category (like “text handling”, “file processing”, or “system access”). Each function has a set of standard arguments (that also have different datatypes. The arguments have meaningful names. In addition to a standard detailed reference, you want to be able to create a list of the functions that use a particular argument.