Introducing:

User's Guide

Kismet Analytic Corp.'s

kis*Meta* Validator

User's Guide Table of Contents

Kismet Analytic Corp.
PO Box 3218 Arlington VA 22204

www.kismeta.com

For technical support, please e-mail:
support@kismeta.com

703-243-5445

November 5, 1997
February 1, 2000

Kismet Analytic Corp.'s

kis*Meta* Validator User's Guide

1.0 Introduction to Validator

Information is a vital resource, and applying standards consistently is essential to maximizing the value of and safeguarding your data investment. Validator will:

- Test object labels for standards adherence.
- Suggest valid abbreviated labels and generate plain English descriptive names.
- Help make the work of multiple project teams consistent.

The result is more consistent data that is both accessible and shareable.

Validator tests metadata, or data about data. Typically, these are equivalent to the column headings in a spreadsheet or database table. Metadata may, however, include any sort of object label, such as those utilized in records, databases, functions, programs, program code modules/components, and programming variables. Validator helps maintain and enforce metadata standards.

Why Use Validator?

Lack of good metadata compounded by inadequate data standards, delivery and enforcement creates headaches for all project team personnel.

The CIO's Point of View - The "Big Picture":

- Designers can't share data between systems.
- Architects can't determine which data in different systems are comparable.

- Users can't find data, because they have no idea how it is classified.

Central Data / Database Administration Point of View:

- Project teams are too deadline-driven to pay attention to standards, and little time remains to act on the results of last-minute design reviews.
- Data standards are published in a manual, but no one reads it.
- Coordinating the work between teams in adjacent rooms is tough; coordinating work between teams in different countries is nearly impossible.

Project DBA / Team Lead Point of View:

- Data documentation is arduous.
- Standards would be a great asset if they were easier to find and apply.
- Data structures often are understood only by the team that created them.
- Developers on the same team create their own database structures which turn out to be too incompatible to integrate.

As professional data administrators and DBAs in the service of large organizations, the designers of Validator have seen the same sets of problems again and again. We know that this tool can help.

Validator helps disseminate standards and makes them accessible and easy to apply. Standards can be checked and corrections applied quickly and early in the development process. Validator actively helps people who create metadata by facilitating their documentation work and by suggesting alternatives to non-standard labels.

Key Features:

With Validator Standard Edition:

- Set and easily test over 25 types of data standards.
- Import metadata text, dBase, ODBC, Erwin, XML et al.
- Import or create new valid term lists; maintain existing lists.
- Generate abbreviated labels.
- Translate cryptic labels into readable documentation.
- Implement ISO11179 data standards guidelines.

E

With Validator ***Enterprise*** Edition:

- Set standards centrally and distribute validation responsibilities to developers and database designers.
- Automate standards publication/documentation.
- Apply custom rules.
- Centrally post a master set of standards for shared access across the network.

2.0 Installation and Setup

Validator requires a 486/66 processor or higher, 16 MB RAM, and about 10 MB for installation. If you have a minimum configuration, at least an additional 40 MB disk space should be available for dynamic memory allocation and work files.

To setup:

1) Insert **Disk 1** in your floppy disc drive (or the **CD-ROM** in the CD-ROM drive).

2) Select **Run** from **File Manager** or the **Start** Button.

3) Type drive letter and "**Setup**". For example:

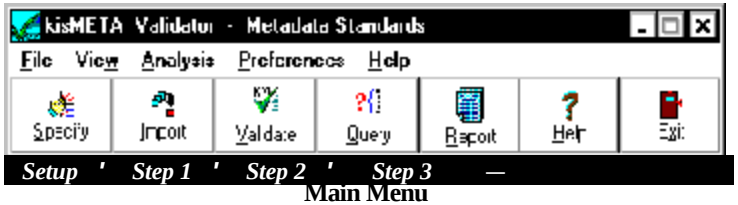
    a:\setup

Follow installation instructions as guided by the setup software.

3.0 Using Validator

3.1 Overview

Although you can run Validator "right out of the box", using it to enforce your organization's data standards requires some up-front customization setup. Setting custom standards can take as little as an hour. If no formal standards exist to implement and must be established by consensus, the time needed depends on your organization's administrative procedures. Validator's validation process is itself a quick three step exercise.



Main Menu

Definition - Specification Set.
Standards specifications, as directed through the Specification form. Specification sets are identified by the shared standards identifier, and by the specific DBMS (or logical set) identifier.

Using the **Specify** form, the user may elect to select a specification set, or to change only the DBMS or logical set to which these standards apply. A set of "generic" standards is included in Validator to give you a head start. Note that while the specification set must be established only once, standards may be added or adjusted at any time. One specification set should be used by all contributors in the relevant project team or organization.

E

The *Enterprise* version of Validator is identical except that the created specification set is used by the distributable validation modules, which rely entirely on these central standards for

validation testing. (However, the validation module allows users to enter standards adjustment "petitions.")

Validation proceeds as follows:

Step 1. Import metadata to be tested.
Metadata that may exist in a repository, CASE tool, database catalog, or other source is fed to Validator. Several data formats can be imported, including .db, .dbf, Access, any ODBC, delimited text, and ERwin. Validator's *Enterprise* edition also has support for several mainframe formats, including (optionally) COBOL File Descriptions.

Step 2. Validate items.
Imported metadata is digested, compared to standards, and analyzed.

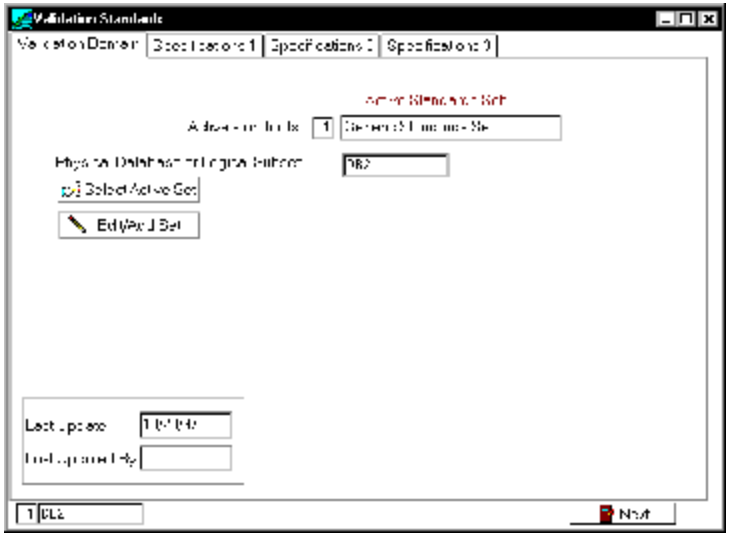
Step 3. Review validation.
Results are available in several report and ad hoc review formats. You may use the validation report results to correct metadata to fit enterprise data standards.

An Aside on Terminology

We recognize that terminology is not uniform in the field; please refer to the Glossary for our usage. The following chart diagrams the correspondence of alternate words:

System	*File*	*Record*	*Element*	*Key*	*Field*
Domain	Relation	Tuple	Attribute	ID	Value
Database	Table	Row	Column	Prime Key	Cell
Function	Database	Record Instance	Field	Identifier	Datum
Subject	Record Type	Transaction		Determinant	
Area	Set			Index	
Application	Entity				

3.2 Setup: Set Specifications



Set Specification Domain

3.2.1 Designating Specification Sets

Standards established through the specification forms are documented internally in a two-part record called the Specification Set. The Specification Set includes a general group of shared standards that apply to all metadata in a given domain (such as a company, or project), and a collection of context-specific standards that apply to a particular physical database such as DB2 or Informix, logical models, or models represented within a CASE tool.

Multiple Specification Sets may coexist to allow for trial runs and comparisons, or may be of use in unusually complex business environments.

Validator ships with "generic" specifications for logical and for common physical databases. These can be altered to fit your needs, or you can add a new standards set.

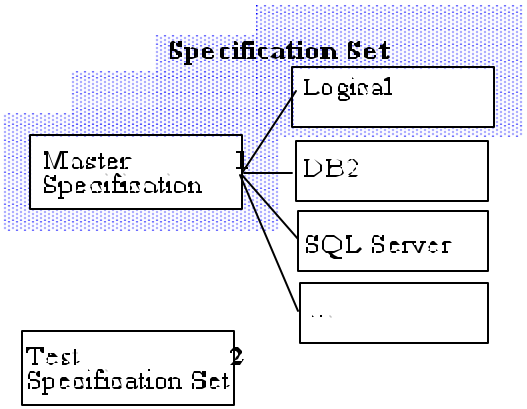


Illustration: Specification Set for Logical Data Standards

The biggest difference between the content of the "master specification" part of the standard and the "DBMS-specific" subset is that each master set has a single valid term list, valid label list, and reserved list, and a few other items such as maximum term count. In other words, the same valid term list can be shared by several subsets. Unique to each subset are the balance of standards such as term length, preferred separators, etc.

To change the Specification Set:

1. Click [Specify] on the Main Menu.
2. From the Validation Standards form, select the Validation Domain page.
3. Click the "Select Active Set" button.

4. To change the entire Specification Set, press the scroll keys to the right of the "Active Standards" edit field.

5. To keep the current active standards, but to change the specific DBMS (or logical set) to which the standards apply, press the scroll up or scroll down arrows to the right of the "Physical Database or Logical Subset" edit field.

To add a new standards set (group of shared standards):

1. From the Validation Domain page, click [Edit/Add Set].
2. Click [Add a new Specification Set].
3. Select the top choice if you want to start a completely blank set, or the bottom choice to inherit specifications from an existing set.
4. If you selected 'inherit', a selection form will appear.
   - Click on the 'SpecName' and 'DBMS' you wish to copy.
   - Click [OK].
   - Change the name of the added set if desired.
   - Click [OK].

   OR, If you selected 'blank', a form will request a new subset name.
   - Enter a name.
   - Click [OK].

   - Change the name of the added set if desired.
   - Click [OK].

To add standards for a particular physical database of logical subset:

1. From the Validation Domain page, click [Edit/Add Set].
2. Click [Include a specific physical DB for this standards set].
3. Select the top choice if you want to start a completely blank set, or the bottom choice to inherit specifications from an existing set.
4. If you selected 'inherit', a selection form will appear.
   - Click on the 'SpecName' and 'DBMS' you wish to copy.
   - Click [OK].
   - Change the name of the added set if desired.
   - Click [OK].

OR, If you selected 'blank', a form will request a new subset name.

- Enter a name.
- Click [OK].
- Change the name of the added set if desired.
- Click [OK].

3.2.2 Setting Data Standards

The results of validation are driven by the data standards Specification Set. Key components and factors include:

- Valid Terms List (Section 3.2.2.1)
- Valid Labels / Special Phrases (Section 3.2.2.2)
- Abbreviation Default (Section 3.2.2.3)
- Terms Characteristics Specifications (Section 3.2.2.4)
- Character Specifications (Section 3.2.2.5)
- Separator Specifications (Section 3.2.2.6)
- Syntax Specifications (Section 3.2.2.7)
- Data Type Class Specifications (Section 3.2.2.8)
- Style Points (Section 3.2.2.9)
- Revised Names (Section 3.2.2.10)
- Roll Your Own Rules (Section 3.2.2.11)

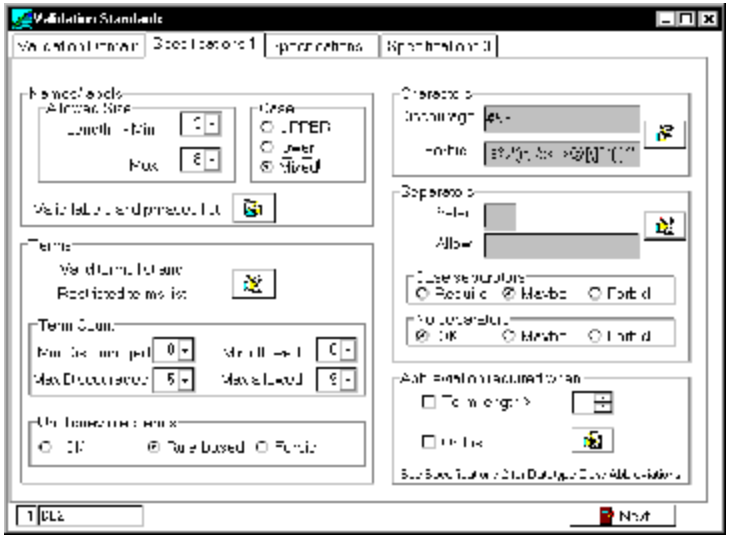
E

Appendix B lists messages that Validator generates as it enforces these specifications. Validator's *Enterprise* edition allows the Specification Module user to edit the text of these messages, and to add custom rules and custom messages.

Several of the specifications allow you to control the quality or urgency of the message that Validator generates. If the issue is one of preferred style, the message may also be worded as a suggestion rather than as an imperative. For example:

Case Should be UPPER.(Violation message)
Upper Case Suggested (Preferred style suggestion)

Terminology Note:
A *label* is the name given to an **element** (or object).



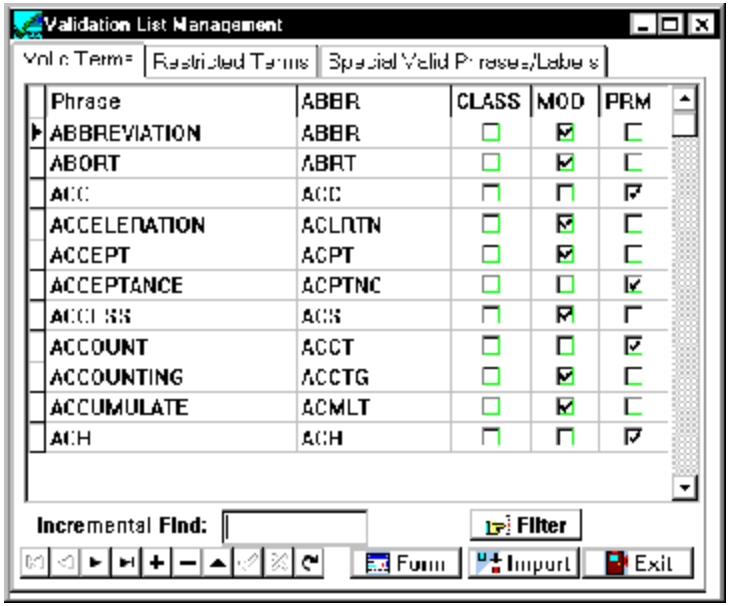
Specification Form, Page 1

3.2.2.1 Valid Terms List

The most influential component of the standards is the Valid Terms List. This list of approved terms and abbreviations, and their "classification," or use in metadata naming, is best created uniquely for your environment.

While Validator will operate without a Valid Terms List, Kismet strongly recommends adopting one. The list defines the preferred or recommended vocabulary for constructing object labels. The actual label should be meaningful, but it is more important to use terms consistently. The enterprise edition allows multiple Valid Term lists, one for each master standards set. Normally, of course, only one Valid Term list would be used in any one organization.

The number of terms approved will vary on circumstances and business environment complexity, but should be kept to a practical minimum; fewer options reduce the chance of redundancy. Valid Term Lists should generally include 300-800 terms, and seldom more than 1000 terms.



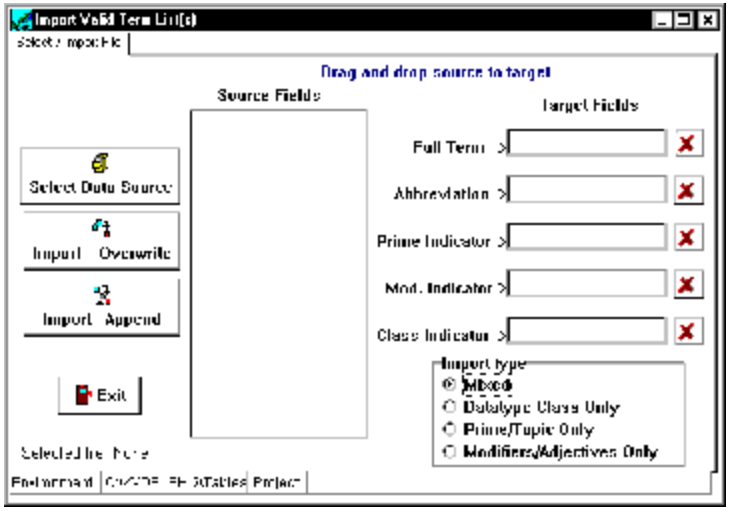
Valid Terms List Edit Screen

Creating a Valid Terms List

1. From the Specifications options, select the Specifications 1 screen page.
2. Click the data entry icon  beside the phrase "Valid Terms List and Restricted Terms List" and then select "Valid Terms."
3. Items may be added one by one, or sets of items also may be imported from an existing source. (Validator's *Enterprise* edition allows direct data import from the "Terms Not Found" list to the Valid Terms List.)

Saving specifications.
The system saves edits as they are being made on the screen, as soon as user activity shifts from the control you are editing. Specifications are saved upon exit. To exit the edit screen without saving changes, click the "x" glyph at the upper right corner of the window (not the Exit button).

Adding New Terms Individually. Items may be entered individually by pressing the "+" key, or by clicking "Form" to enter terms on a less cluttered screen. Here, you will be able to tell Validator whether the word is acceptable as a class word, modifier, or prime.



Valid Terms Import Screen

Building Your Own Valid Terms List. Validator offers several collection of terms that may be used in building your own Valid Terms List, each located in the "/Lex" directory. Subjects include common:

- Acronyms..... (TACCR97.db)
- Abbreviations..... (TABBRR97.db)
- Business Modifier Terms..... (TBUSR97.db)
- Business Topics/Primes..... (TBUST97.db)
- Datatype Class Terms
  - 3 letter Abbr. (NME, DTE, CDE....) (TCLAS397.db)
  - 2 letter Abbr. (NM, DT, CD....) (TCLAS297.db)
  - DOD 2 letter Abbr. (TCLASDOD.db)
- Information System Topics/Primes.. (TISR97.db)

However, your organization may have an existing Valid Terms List, or if not probably will wish to add to or replace many of these

terms. Kismet recommends at a minimum that you establish a standard data type class terms set (see TCLAS397.db).

Installing Data Sets Into the Valid Terms List:

1. From the Valid Terms screen, click on Import, then on Select Data Source.
2. Specify the file type and file name and click on Open.
3. Drag and drop source fields to their matching Validator counterparts; clicking on a red "X" allows you to clear the assignment.
4. Finally, click on Import Append. *(Important: Clicking Import Overwrite will eliminate your original Valid Terms List.)*

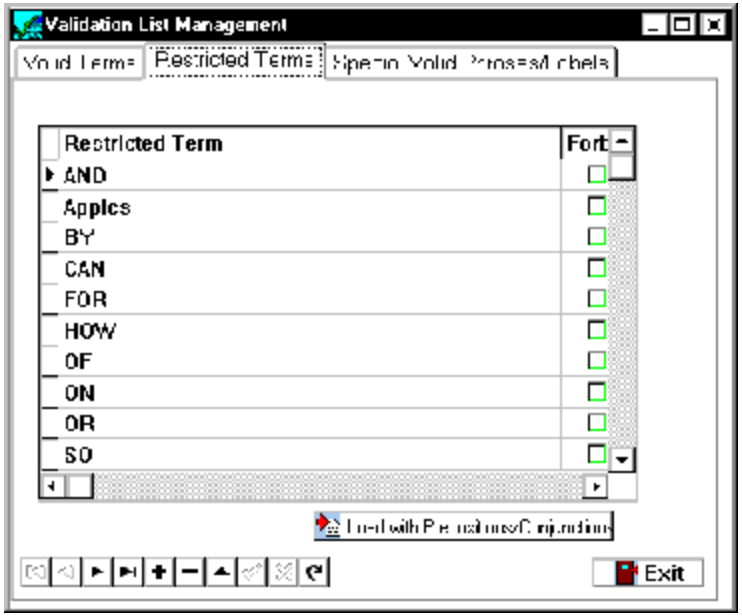
The metadata may be in either of two formats in any order. In the table below, a term may be both a *modifier* and a *datatype class*.

Full term	Abbreviated term	Class indicator "Y"or ""	Prime indicator "Y" or ""	Modifier indicator "Y" or ""
Alpha(33)	Alpha(20)	Alpha(1)	Alpha(1)	Alpha(1)

In the next table, each item must be one or the other (or null).

Full term	Abbreviated term	Class indicator "C, M, or P"
Alpha(33)	Alpha(20)	Alpha(1)

Using the **Filter** key, you may sort the Valid Terms List by Data Type Class words, prime terms, and modifiers.



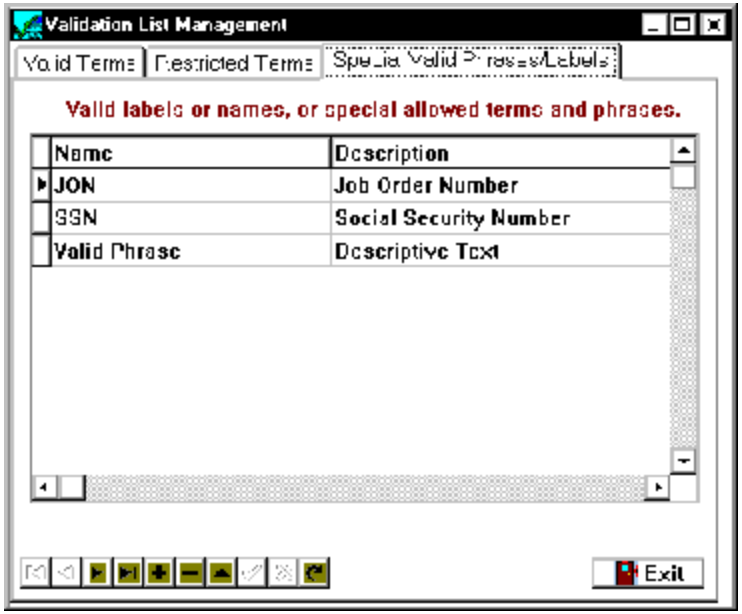
Restricted Terms Edit Screen

Restricted Terms. On the Restricted Terms screen page, the user may enter restricted or reserved terms which no element name may contain. "Of," " a," "data," and "the" are examples of frequently restricted terms. The button "Load with Prepositions and Conjunctions" allows quick population of a set of common prepositions, articles and conjunctions.

Invalid terms and automatic substitution

E

If a term or acronym is used in an organization, but is invalid, Validator generates a "term not found" message if it is not in the lexicon. You can instruct Validator to suggest an appropriate substitution by adding the item and valid substitute to the Validator Lexicon. The result is a "term invalid" message with a substitute.

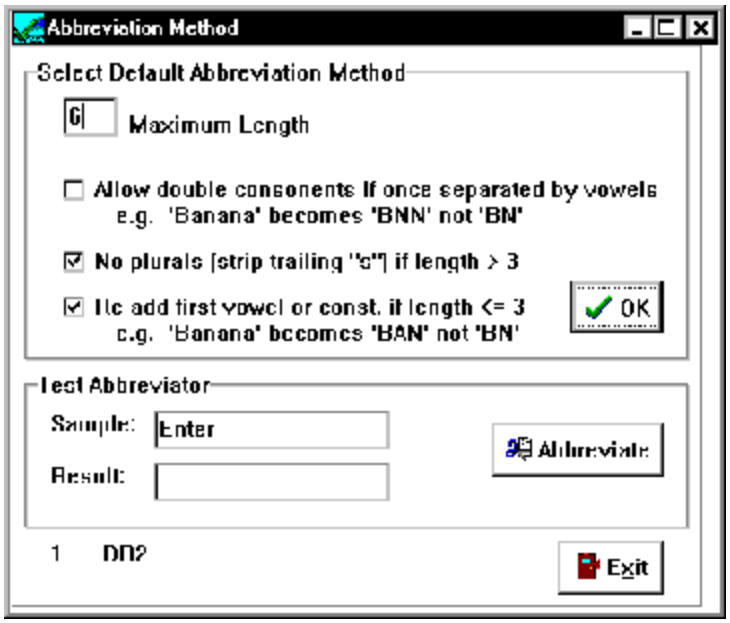


Special Phrases/Valid Labels Edit Screen

3.2.2.2 Valid Labels/Special Phrases

Some organizations specify lists of standard labels as whole element names that are context independent; they stand alone. A common example is "SSN" for Social Security Number. Acronyms and short abbreviations that are widely understood are the best examples. The category may also include legal and other phrases that always appear as a set of terms, such as "Bill of Lading." The previous phrase will be accepted even if "of" is a restricted term. Phrases in this category are not enforced with case sensitivity. This option may be accessed directly from the Specifications 1 Screen by clicking the **Valid Labels and Phrases List** icon, or secondarily from the Valid Terms List and Restricted Terms List icon.

As with Valid Terms, items may be entered individually or imported from an existing source.



Abbreviation Default Edit Screen

3.2.2.3 Abbreviating Terms

In considering the subject of abbreviations, first decide whether or not unabbreviated terms will be forbidden, or whether this abbreviation style should follow rules which the user will choose (see method below). From the Specifications 1 screen, bottom left, one of three options must be checked; you must select "Rule-Based" to set term abbreviation parameters.

Setting Rules. From the Specifications 1 screen, the user may stipulate whether an abbreviation will be required when a given term length is exceeded; when it is on the Required Abbreviations list; and when the term is a data type class word. To mark an abbreviation as mandatory, click the icon to the right of "On list," locate the term, and then check the "Abbr. Mandatory" box.

Abbreviation Default is Validator's algorithm for abbreviating terms. The simplest abbreviation convention is to keep the first letter, strip out all following vowels, and truncate the word to a desired length. Additional steps may include stripping consonant pairs; replacing a vowel for short terms; and stripping plural forms.

Validator allows you to control these typical variables through the Abbreviation Method screen.

To make changes to the Abbreviation default (see caution below), click on the "Define Method" button on the Specifications 2 screen page.

Important!

Once the abbreviation default algorithm is set and used, any change to the algorithm will mean that the old work will no longer be compatible with new work and new standards. Once you set and test a default, do not change it unless absolutely necessary.

However, abbreviations that may differ from the product of this method may be widely used. For example, NUM instead of NBR, or APROP instead of APRPN. All common allowable exceptions should be entered in the Valid Terms List. (Note that Validator has an internal lexicon of common English abbreviations.)

3.2.2.4 Terms Characteristics Specifications

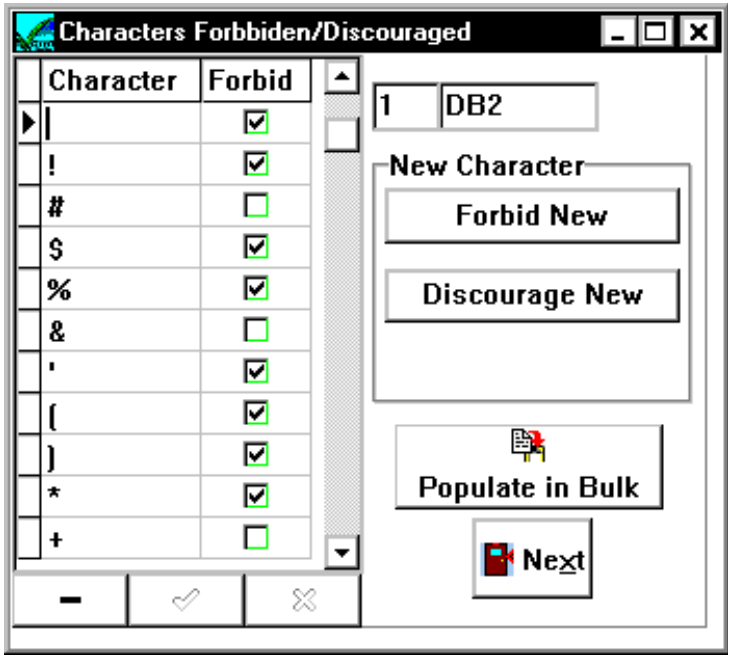
Terms Characteristics Specifications define a group of standards that include:

- *Label Length:* Minimum and maximum number of characters allowed.
- *Case:* Upper or lower case, or 'mixed' allowed.
- *Term Count:* Number of allowed terms in a label.

These are specified on the Specifications 1 screen, left side.

3.2.2.5 Character Specifications

Principally, Character Specifications comprise a list of characters that are restricted or forbidden. For physical databases such as SQL Server, a base set of restricted terms may be specified. For example, the SQL Server manual states that the character "/" is not allowed in an element name.

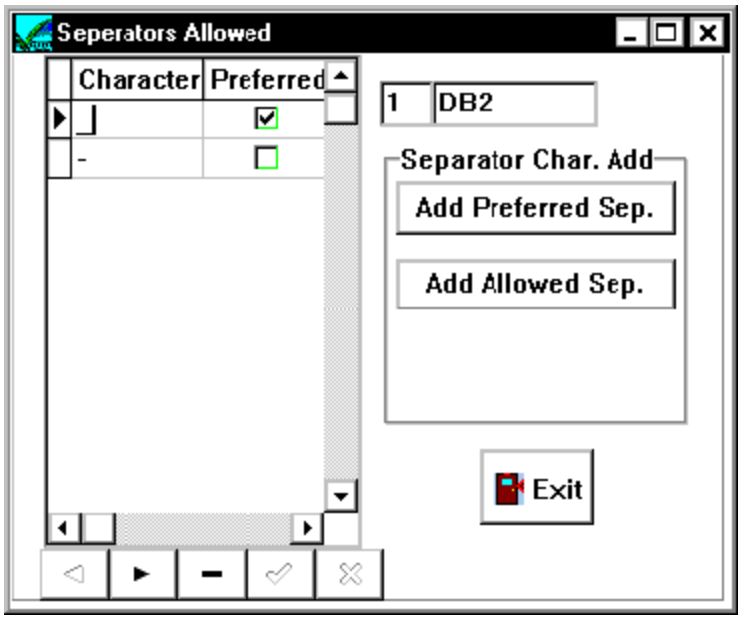


Characters Forbidden/Discouraged Screen Page

To forbid/discourage characters:

1. Click on the data entry icon  to move to the "Characters Forbidden/Discouraged" screen.
2. Click "Forbid New" to add a new forbidden character. An edit box will appear. Enter the character into the edit box and click OK.
3. Click "Discourage New" to add a new character which is allowed, but discouraged. Characters also may be deleted from this list.

When adding a new specification set, adding the set of all non-alphanumeric characters (except '_') and dealing with exceptions is easier. Use the "**Populate in Bulk**" characters button.



Term Separator Screen Page

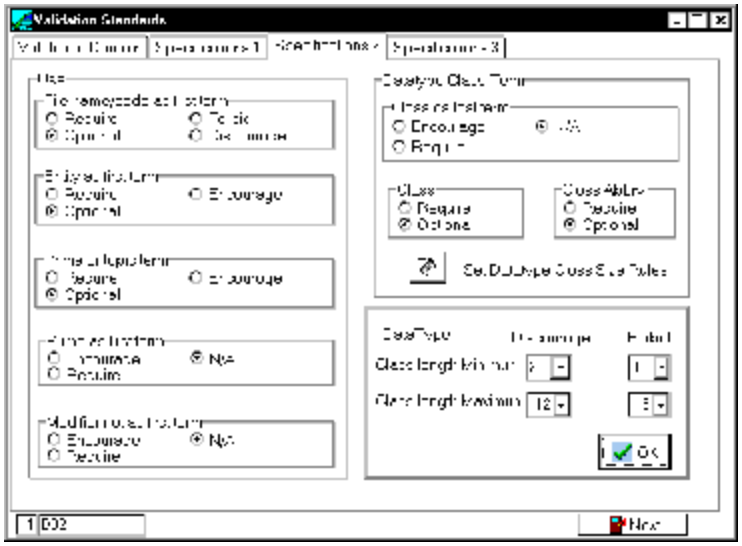
3.2.2.6 Separator Specifications

Separator Specifications identify the preferred term separator within labels, and acceptable alternatives if any. For example, underscores are most commonly used, followed by hyphens and spaces.

A variety of other characters may be used for this purpose, including commas, periods, slashes, etc. In additional, some environments prefer case separators, (e.g. "EmpLastName"), and in rare cases have adopted a no-separator approach (e.g. "EMPLASTNAME"). People (and Validator) often have difficulties interpreting labels created with the no-separator method.

Because Validator first searches for datatype class terms, if no-separator labels are common in your environment, it is imperative that you enter the most commonly used datatype class terms into the Valid Terms List.

To specify preferred and allowed separators, click on the icon to the right of the Separators field on the Specifications 1 screen page.



Specifications 2 Screen Page

Additionally, the Specifications 1 screen page provides fields for designating whether case separators should be required, allowed, or forbidden, and whether labels with no separators are permitted, suggested, or forbidden. If case separators are required, Validator has a built-in **Case Separator Generator** which will separate terms by case on the Revised Name Query Report.

3.2.2.7 Syntax Specifications

Syntax is meaning derived from term order and selection. A data labeling system conventional for COBOL copy code and common

in older mainframe systems includes, as a prefix, an application, database or system name and a file/record name as a part of every data element name (as a prefix). An example is: "A58-PERSREC-SOC-SEC-NUM", where A58 is the application/database ID and PERSREC is the File ID.

It is common practice today - and in our opinion preferable on theoretical grounds - to regard only the element-unique portion of the name as the "label", and to inherit the system and record attribution from context, whenever applicable.

The Specifications 2 screen page gives the user the opportunity to specify whether the file name, entity, or prime term should be the first term in the label, and whether a data type class word should be the last. Note that Datatype Class must be defined as mandatory before "Datatype Class as Last Term" can be enforced, and that Prime/Topic must be defined as mandatory before "Prime as First Term" can be enforced.

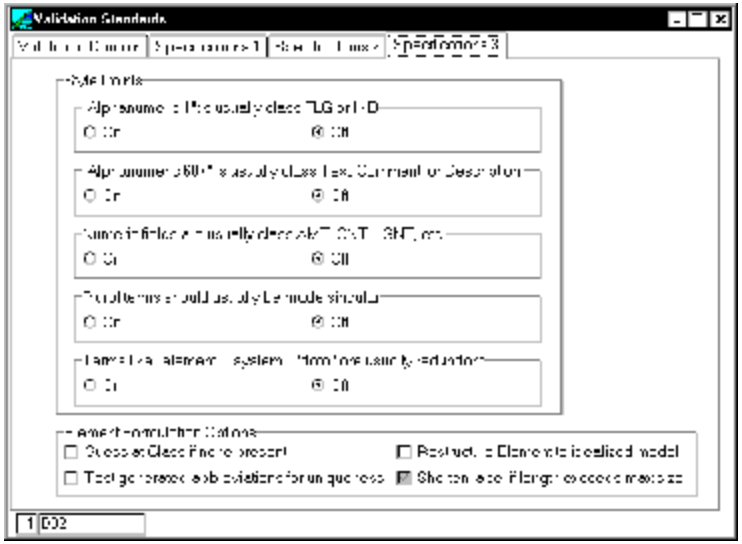
3.2.2.8 Datatype Class Specifications

This specification category augments the syntax specifications: it determines whether datatype class words such as "code," "amount," "name," and "number" are required in every label, and whether they should be the last term in the label (Specification 2 screen page).

In addition, you can tell Validator here whether datatype class words must be abbreviated, and then specify the minimum and maximum length. From the Specifications 2 screen page, click on the icon to the right of "Set Datatype Class Size Rules." If you assign length "3," as an example, this tells Validator that the term *must* be abbreviated to no more than three characters.

Note that Validator's *Enterprise* Edition contains the feature **"Guess at Class if None Present"** (see Specifications 3 screen page). With this option, Validator will make an informed attempt to add a data type class word to the end of element names if one is lacking. This guess will appear on the Revised Names Query page.

3.2.2.9 Style Points



Specifications 3 Screen Page

Stylistically consistent practices can be encouraged, for example providing that:

- data types are compatible with datatype classes;
- plural forms are not used;
- terms such as "data" and "element" are not used redundantly.

From the Specifications 3 screen page, five "on/off" style switches are available:

Alphanumeric 1 is usually (data type) class FLG or IND. As a matter of style, INDICATOR or SWITCH commonly is used for Boolean true/false or for a handful of category options, such as Male/Female/Unknown. In contrast, CODE is used for a medium number of category options (such as State_Code), while ID or NUM is used for unique identification, such as SSN. If this option is selected, a seemingly inappropriate term use will generate a suggestion issue message: "*Alphanumeric I attribute usually IND or FLG.*"

Alphanumeric 60+ is usually (data type) class TXT, COM, DSCR, (e.g. Blob, IMAGE or VARCHAR). As a matter of style, long

descriptive fields often are identified with data types such as TEXT, DESCRIPTION, MEMO, or COMMENT. A seemingly inappropriate term use will generate a suggestion issue message: "*Alphanumeric 60+ attribute usually TXT or DSR .*"

Numeric (or Monetary) fields are usually (data type) class AMT, CNT, LGT, etc. As a matter of style, numeric data commonly is identified with AMOUNT, COUNT, LENGTH, QUANTITY, VOLUME or other numeric measure data types. If checked, a seemingly inappropriate term use will generate a suggestion issue message: "*Numeric attribute usually AMT or CNT.*"

Plural terms should usually be made singular. As a matter of style, plural terms are not typically allowed in element labels. If checked, a seemingly inappropriate term use will generate a suggestion issue message: "*Plural terms are generally not allowed*." Validator will also strip plural forms while generating naming suggestions.

Terms like "element", "information", "data" are usually redundant. As a matter of style, such words usually are not used in labels, since their meaning is usually implied by context. If checked, a seemingly inappropriate term use will result in a suggestion issue message: "*'Data,' 'element,' 'system' etc. often redundant.*"

3.2.2.10 Revised Names

Revised names are alternate or suggested metadata labels. Validator uses available information to offer suggestions toward reformulating metadata labels for a variety of purposes (see 3.4, Reviewing Validation Results). Several options include:

- **Abbreviated Name.** Every term is abbreviated. The result is the shortest conventionally acceptable name using the original terms. From the Specifications 1 screen page, select "Forbid" from the Unabbreviated Terms panel.

- **Long/Business Name.** Validator will produce its best estimate of the long name, in which every terms is spelled out fully. Validator will automatically translate any abbreviation or acronym found in the Valid Terms List or the Valid Phrases/Labels List to its full name, e.g. "COGS" would

generate "Cost of Goods Sold." In addition, Validator can use its internal lexicon to "guess" at short term meanings. **Term Guessing** can be switched on and off through the Preferences option on the main menu; the results should be manually reviewed. Term guessing "on" results in more information but also yield a higher "false positive" rate.

< **Short Name.** A label is shortened through abbreviation just enough to meet maximum size constraints. On the Specifications 3 screen page, select **"Shorten label if length exceeds maximum size."** This feature will shorten middle terms to as little as one character, beginning with the next to the last term.

< **Ideal Model Name (Enterprise Edition).** A label is restructured to meet standards as completely as possible. Restructuring supported by Validator may include:

- < Placing the datatype class name in the final position.
- < Adding a consistent data type class.
- < Placing primes in the first position.
- < Placing system and/or record names in the first position.
- < Abbreviating a label to meet maximum size constraints.

These options (except for the last; see above discussion of Short Name) may be activated on the Specifications 2 screen page.

Uniqueness. If the option **"Test Generated Abbreviations for Uniqueness"** on the Specifications 3 screen page is turned on, Validator will ensure that different terms do not end up with the same abbreviated form in the short name. The term that Validator first finds will be allotted the "standard" abbreviation.

Attribute Formulation Options - Guess At Class if None Present *(Enterprise Edition*). Using a data type class word as the last term of an element label is a common data standards convention. Validator's *Enterprise* Edition can suggest a class term if none is present. The Specifications 3 screen page contains a check box to activate this option: "Guess at Class if none present." The results appear in the "Reformulated" column of the Revised Names Query.

E

Attribute Formulation Options - Restructure Element to Ideal Model *(Enterprise Edition).* Validator's *Enterprise* Edition regards either the sequence "Prime – Prime Modifier – Class Modifier – Data Type Class" or " Prime Modifier – Class Modifier – Data Type Class" as the ideal element label syntax model. Checking the Specifications 3 screen page option "Restructure Element to Ideal Model" instructs Validator to attempt to produce this sequence from provided terms. The results appear in the "Reformulated" column of the Revised Names Query.

3.2.2.11 Roll Your Own Rules *(Enterprise Edition)*

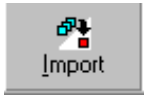
E

The *Enterprise* Edition of Validator allows you to enter your own rules and issue messages. A critical decision is at which level the rule is to be enforced:

1) System
2) Record
3) Element
4) Term

Observe that if the rule is at the term level, potentially every one of hundreds or thousands of terms may generate the issue message.

Validator allows the user to compose a rule using a simple scripting language similar to Basic or Pascal. You select an actor from several options, and then enter the constant or variable.



3.3 The Validation Process

Data is imported to Validator in a two-step process. The first step, collecting metadata, is in your hands. The second step, importing metadata, is assisted with automation.

3.3.1 Collecting Metadata

In many organizations, metadata will be found in an easy-to-access format such as a repository, data dictionary, or database catalog. Data in these tools need only be selected, and exported to a .dbf, .db, comma-delimited, or .csv format.

The metadata input requirements and mapping information is shown on the graphic "Metadata Import Format" (following page). Fields may be in any order. Strictly speaking, only the element short name is required for validation, but the other fields, when available, are used by Validator to improve analysis and add context to reports. Record/File and System IDs help ensure that the validation product's context will be recognizable to a user, especially in the form of a printed report.

The metadata source may also be a CASE tool such as ERwin, Designer 2000, Systems Architect, or IEF. In general, data from these tools can be readily exported in Validator-ready formats. Validator can also read ERwin's .dbf export product directly.

In some organizational environments, metadata may be widely scattered and in many formats, and therefore must be collected manually. Kismet recommends CASE tools with reverse engineering capabilities, such as ERWIN, as metadata collection aids. One area in which many CASE tools are deficient is mainframe/flat file (COBOL) file description (FD) reverse engineering. Kismet offers a Validator-compatible FD reverse engineering tool (*fd*REVERSER) as an additional-cost option.

System/ Database ID	Record/File ID	Element/ Attribute Short Name	Data Type	Data Type Length	Business Name/ Descriptive Name	Key Indicator
Alpha-numeric, max 33 characters	Alpha-numeric, max 40 characters	Alpha-numeric, max 40 characters	Alpha-numeric, max 9 characters	Alpha-numeric, max 3 characters	Alpha-numeric, max 80 characters	Alpha-numeric, max 1 character
Optional	Optional*	Required	Optional*	Optional*	Optional	Optional*

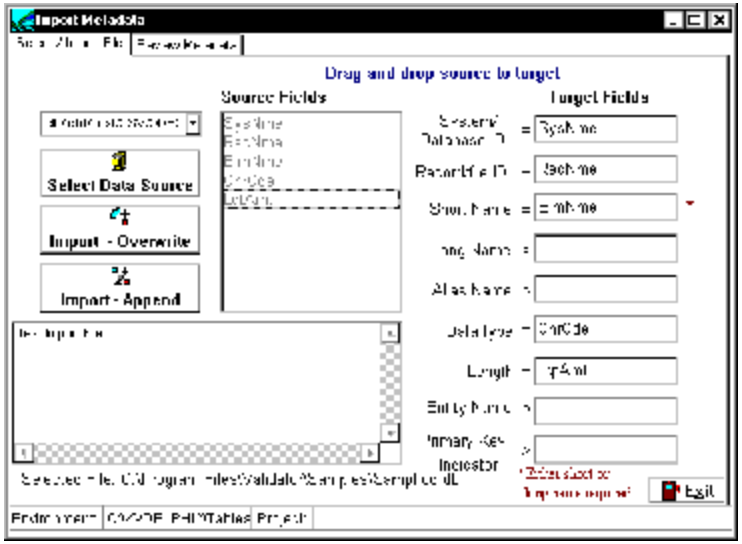
Metadata Import Format

**strongly encouraged*

3.3.2. Importing Metadata into Validator

Once metadata is available in .dbf, .db, comma-delimited, or .csv format, or is exported from ERwin or a SML-compliant CASE tool, importing metadata into Validator is a simple three-step process:

1. Select data source and type.
2. Map source fields to validation fields ("targets").
3. Click on Import/Overwrite (or Import/Append, if you are combining several data sets together for a single



Import Metadata Screen

validation run).

Map data from source fields (the data structure of your databases) to the fields that Validator uses internally - the "target fields" - by clicking on an individual source field and dragging it to the corresponding target field.

Sample source-target mappings are listed in the box on the following page: *Alternate names - Physical and Logical Examples*. Specific steps required to import metadata from ERWin are outlined below.

Physical Style	System/ Database ID	Record/File ID	Element/ Attribute Short Name	Data Type	Data Type Length	Business Name/ Descriptive Name	Key Indicator
Logical Style	Function Owner Domain	Entity	Item Field	Character Category Type	Length Size	Long Name Full Name	Key
Data Examples	A43 Personnel	PERSREC Employee	SocSecNum EMP_NM	Code Amt A(22)	0 22	Social Security Number	K

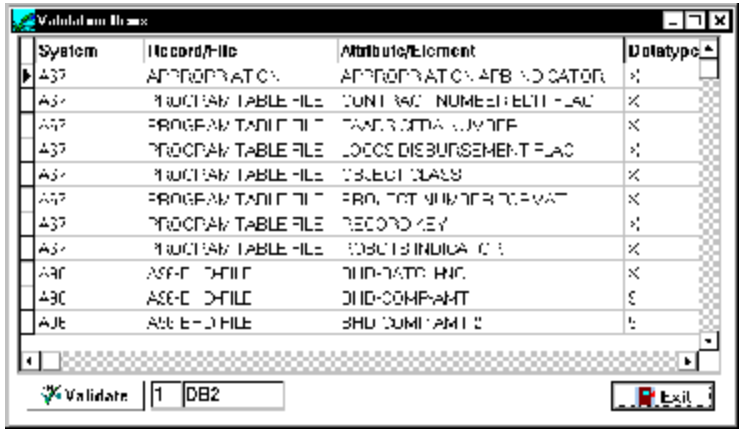
Alternate Names / Physical and Logical Examples

Special Section: Importing Metadata from ERwin

1. Open the ERwin data model that you want to export to Validator.
2. Choose "Dictionary Manager…" on the File menu. ERwin opens the Target Dictionary Storage DBMS dialog.
3. Click the "Clipper" button and click "OK." You do not have to have Clipper installed to save to a Clipper-based dictionary. ERwin opens the Clipper Database Directory dialog.
4. Double-click on an existing directory name or type a new name (e.g., ERWDICT) in the "Directory Name" text box.
5. Click "OK." If you typed a new name, ERwin prompts you to confirm the creation of a new directory. Click "Yes" to create the new directory or click "No" to cancel the creation of a new directory and return to the Clipper Database Directory dialog. ERwin then opens the Dictionary Manager dialog.
6. Click the "Check In" button. ERwin opens the Check In Diagram dialog. Click the "Check In" button. If you have already saved a diagram dictionary to the selected directory, ERwin prompts you to confirm the replacement of the existing files. ERwin saves the model dictionary to the directory specified in Step 4 as a set of DBF files and displays a message when the process is completed.
7. Click "OK." ERwin closes the dialog and returns to the active diagram.

3.3.3 Validating Metadata

Before initiating validation, Kismet recommends that you first review the metadata selected (from Import Metadata screen page, click on **Review Metadata**), and then check the specification set to be used (lower right corner of the validation form), to ensure that the tool will validate as anticipated. When ready, click on the Validate button on the main menu. (Note that if you have both

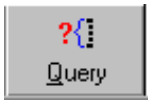


Validation Items Form

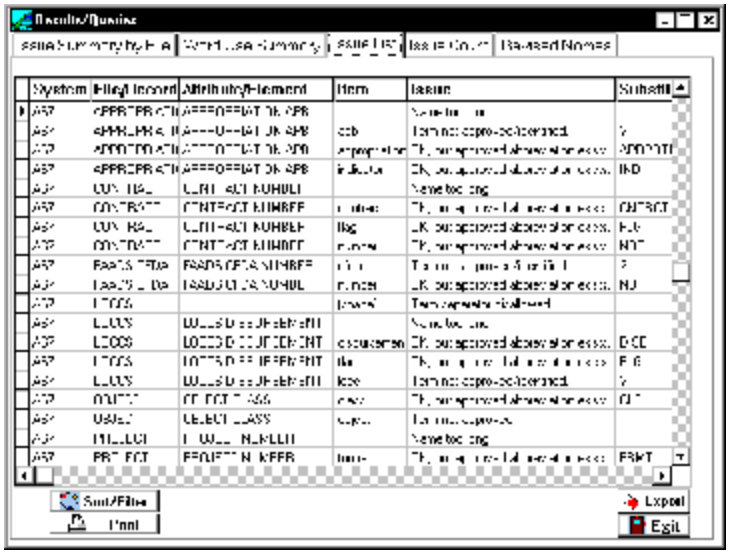
logical and physical metadata, you should run each separately.)

Press  to begin the analysis process.

Validator uses intensive database processing. If you have a 486 computer or have limited memory, keep the number of items to validate in each batch to under 500.



3.4 Reviewing Validation Results



Query Screen - Review Results

Validator provides several means for reviewing and using validation results. Information types include:

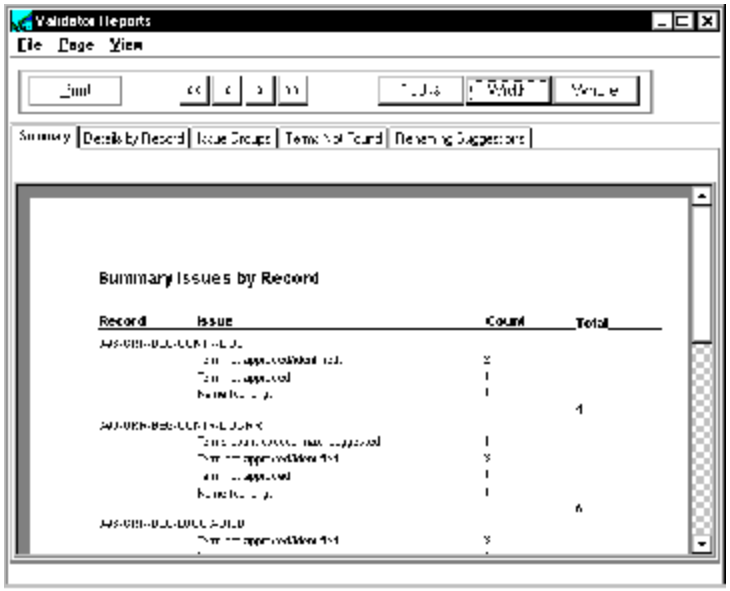
- Standards violation issues (specific errors identified)
- Stylistic or standards issues (suggestions made)
- Element name alternative suggestions
- Plain-English business names suggestions
- Statistics

Issues and revised names may be both printed and exported.

The following table shows information available from the **Query** screen. Besides issues, the revised names is an important validation product. Revisions include short names -abbreviated and to fit length specifications, long names - which

translate/document abbreviations and acronyms to plain English, and ideal name - e.g. Prime-Mod-Class.

Query Page	Purpose
Issue Summary by File	Issue type and count by system and record. Statistical information can reveal areas that require special attention.
Word Use Summary	Terms used in element names, listed with issues, and which record(s) and element(s) contain the terms. Results may be printed.
Issue List	List of all issues. The **Sort/Filter** feature allows you to: - Sort by issue, record, element, or term. - Filter out suggestion and syntax or morphology issues. - Locate a specific attribute name. - Show only items not found, or issues specific to class, character, or syntax. The selected/filtered results may be printed, or exported to a .db/.dbf file.
Issue Count	The Issue Count screen is a high level summary of counts and statistics.
Revised Names	Business names generated by Validator are offered as suggestions or documentation aids. In addition, a suggested abbreviated form is shown. Starting with the original metadata provided, non-standard abbreviations are standardized, and full words are abbreviated. In the *Enterprise* edition, additional normative corrections (such as the results of automatic datatype class generation) are also shown. Results may be printed, or exported to a .db/.dbf file.



Report Screen - Summary Issues

Validator makes several pre-formatted reports available. The **Report** option provides information similar to the Query option, but the output is formatted to generate attractive presentation-quality printed reports. Your company name and a header may be added to the reports.

4.0 Special System Features

Printer setup. From the main menu, select the File drop-down menu.

Background overlay. From the main menu, select the Preferences drop-menu.

Hints on/off. Many Validator controls provide hints and suggestions when the cursor passes over them. Experienced users may find these hints unnecessary. On the Preferences drop-down menu, check to turn hints on/off for most screens.

Saving specifications and other edits. The system saves edits as they are being made on the screen, as soon as user activity shifts from the control you are editing. Specifications and edits are saved when you click the Exit button on the current form.

Cancel. In input forms, the button with the "X" glyph cancels that input assignment.

Term Guessing. In order to guess at the meanings of terms not found in the Valid Terms List, Validator uses its internal lexicon to look up word roots and substrings. 'Guessing' yields more 'false positive' errors, but also often gives more good information. From the main menu, select the Preferences drop-down menu.

ISO11179 Standards Conventions On/Off. Selecting this option causes screens and reports to adhere to ISO11179 conventions.

E

Changing issue messages. The *Enterprise* edition of Validator allows the DA/DBA to edit issue messages. From View, select Edit Issue Messages.

Generate documentation. The *Enterprise* edition of Validator features automatic standards documentation generation that summarizes the rules entered in Validator's specification form. From View, select Standards Documentation.

Export. Validator generates SML which may be imported into many case tools, such as ERWin. From File, select Export. In addition, tabular data may be saved from the query screens.

Index