



# **Tool to Ingest and Elucidate Requirements (TIGER)**

**Pro Version**

© 2006 Joseph Kasser

# TIGER Pro

**Software to produce better requirements**

---

*by Dr Joseph Kasser DSC, CENG, CM*

*TIGER Pro demonstrates the power of the object-oriented approach to defining requirements.*

# TIGER Pro

© 2006 Joseph Kasser

**TIGER Pro - Version 1.13.9 - Registered to SEEC**

File Tools Settings Report Help

Database: C:\Program Files\Borland\Delphi7\projects\tiger pro\source\Database\MCSS

**Requirement**  
Requirement ID: 5.9.9 Date: 4/09/2006 Poor word:   
The MCSS shall not be damaged by a sudden loss of power or prolonged transients of the kind mentioned in paragraphs 5.9.7 and 5.9.8 on the power supply line.

Keywords:   
AutoCreate:   
Assign Create

Select Requirement  
5.9.9  
Post  
Autoincrement ID  
New Badger  
Display control  
Apply Filter

Reports and Graphs View Data Additional Attributes

Ingest/Elucidate Priority Risk Severity Risk Probability Risk Level Cost Estimates

**Total number of requirements with defects**

Defect type	Requirements
1	5
2	115
3	86
4	23
5	0
6	0
7	0

**Elucidation Results**

**Defects**

1- Multiple reqs	4
2- Poss. multi.	115
3- Unverifiable	86
4- Wrong word	23
5- User defined	0
6- Poss. design	0
7- Incomplete	0

**Summary**

All Records	246
Requirements	246
Ignore PW	0
Defects	228
Fig. of merit	8

Clear Help

Update Summary  
Automatic  
Manual

Summary data  
Records: 246  
Requirements: 246  
Comments: 0

Ingest Mode: All (selected) By word  
Format: Text (selected) CSV DOORS  
Ingest Help

Elucidate Report: Text (selected) Graph  
Elucidate Help Filtered



This program is distributed as Shareware and if copied and distributed must be copied in full.

Printed: September 2006 in Adelaide, Australia.

# Table of Contents

Foreword	1
<b>Part I Introduction</b>	<b>3</b>
1 Background .....	3
2 Uses for TIGER Pro .....	4
<b>Part II Overview</b>	<b>8</b>
<b>Part III Major functions</b>	<b>10</b>
1 Ingesting a requirements document .....	10
Format of requirements that can be ingested .....	11
The parts of a requirement .....	11
Poor words.....	12
Types of defects.....	12
2 CSV Control Panel .....	13
3 Elucidating a requirements document .....	14
4 Working with the elucidation results .....	14
Teaching TIGER Pro new Poor Words .....	15
Overriding poor words .....	15
5 Inheriting from a second requirements document .....	15
Mode of operation .....	16
Replace Text .....	16
Duplicate requirement ID Options .....	16
Replace ID prefix .....	17
6 Adding a new requirement .....	17
7 Changing Requirement IDs .....	18
<b>Part IV The Main window areas</b>	<b>20</b>
1 Top Tool Bar .....	20
Adding key words to the database .....	20
2 Requirement Panel .....	20
Keywords Panel .....	21
3 Select Requirement Panel .....	22
4 Display Control Panel .....	22
5 Summary Panel .....	23
6 Reports and Graphs Panel .....	23
Reports and Graphs .....	24
The View Data Panel .....	24
7 Ingest Control Panel .....	25
8 Elucidate Control Panel .....	25
<b>Part V The Menus</b>	<b>28</b>

<b>1 File Menu</b>	<b>28</b>
Database	28
<b>2 Tools Menu</b>	<b>29</b>
The BADGER Template	29
Editor - Files/Poor Words	30
Edit a number of files	31
File Control Panel	31
<b>3 Settings Menu</b>	<b>31</b>
Font	32
Set Background Colour	32
Sounds	33
<b>4 Report Menu</b>	<b>33</b>
<b>5 Export Menu</b>	<b>34</b>
<b>6 Help Menu</b>	<b>34</b>
<b>Part VI Attributes of a requirement</b>	<b>36</b>
1 Acceptance criteria	36
2 Cost estimates	36
3 Key words	37
4 Priority	37
5 Rationale	38
6 Risk	38
7 Traceability	39
<b>Part VII Troubleshooting and FAQs</b>	<b>42</b>
1 Trouble shooting	42
2 FAQ	42
<b>Part VIII Version History</b>	<b>44</b>
<b>Part IX Credits</b>	<b>46</b>
<b>Part X Author</b>	<b>48</b>
<b>Part XI Software Licence</b>	<b>50</b>
1 Registering TIGER PRO	50
2 Shareware	51
<b>Index</b>	<b>52</b>

## Foreword

The systems and software development industry is characterized by a paradigm of project failure. The situation has been described by Cobb's Paradox, which stated "We know why projects fail, we know how to prevent their failure --so why do they still fail?" One of the known contributing causes of these project failures is poor requirements engineering and management, which has been repeatedly and widely discussed and documented for at least 10 years. However, this continual documentation and discussion of the problem of poor requirements engineering and management has not resulted in a practical solution to the problem.

There has been recent recognition that a requirement is more than just the imperative statement; additional properties (e.g. risk, priority, and traceability) have been proposed, yet the current paradigm scarcely provides a place to store those properties. However, in practice, there is difficulty in adding these additional properties to the traditional requirement document or database and then managing them. Moreover, activities that have become increasingly identified as being critical to success (e.g. risk planning and management) in many cases are not performed in the current paradigm. In those cases where they are performed, they tend to be treated as add-on's, and implemented in a complex process that could have been designed and pictured by W. Heath Robinson (UK) or Rube Goldberg (USA).

TIGER Pro presents a preliminary introduction to an Object-Oriented approach to requirements engineering and management that might help to reduce the contribution of poor requirements engineering and management to project failures because the object-oriented paradigm provides for these add-on properties and activities and provides a place to store them, namely within the requirement object. Moreover, as shown by TIGER Pro, the packaging of processes or functionality together with the data inside the requirement object provides for automating some of the manual processes that contribute to the successful realisation of systems but which may be overlooked in the current paradigm.

Joseph Kasser  
Adelaide, Australia

# TIGER Pro

Objective requirements  
engineering

# Part



# 1 Introduction



## Tool to InGest and Elucidate Requirements - Professional (TIGER Pro)

Copyright © Joseph E. Kasser 2006

Software to help you smarten up your requirements.

This software is [Shareware](#) and should be licensed if used for more than 14 days except by students in a course.

TIGER Pro is a front end tool that assists you in writing good requirements and can fix the [two types of defects](#) most commonly created by students writing requirements for the first time.

TIGER Pro is designed to help you smarten up your requirements and demonstrate the difficulty of writing good requirements.

TIGER Pro's main functions are:

1. [Ingest requirements from a text file](#) (filename.txt).
2. [Elucidate requirements](#) to point out possible defective requirements from the perspective of testing or verifying compliance to the requirements.
2. Fix the two most commonly made mistakes by students learning about requirements.
3. [Inherit subsets of requirements](#) from one database into another.
4. Show how the object-oriented approach of combining attributes of requirements from the process and product dimensions can improve the current requirements engineering paradigm.

TIGER Pro scans requirements databases for the presence of [poor words](#). You can also [teach TIGER Pro about new poor words](#) when you read the entries in the requirements database.

## 1.1 Background

TIGER Pro is a front end tool for assisting the user to write good requirements. TIGER Pro is designed to help you smarten up your requirements and understand the difficulty of writing good requirements.

The expensive cost and schedule impacts resulting from poor requirements have been repeatedly documented for at least 10 years. However the continual documentation of the problem and its attributes has not resulted in a practical solution. Even requirements management tools, have, in the main, avoided dealing with the problem of poor requirements. The problem of poor requirements thus seems to be in the category of those problems for which a complete solution cannot be found, and consequently tends to be tolerated.

TIGER Pro provides a partial solution to the problem of poor requirements by synergizing the principles of Total Quality Management, expert systems, and knowledge management. The part of the problem solved, is the common practice of producing documents containing requirement statements that:

- are poorly worded so as to be vague and unverifiable, and
- contain multiple requirements in a single paragraph, which complicates the traceability of tests to requirements as discussed below.

The format and characteristics of good and bad requirements have been published for years, yet people are still producing poorly worded requirements. This is because many of the writers of requirements have little idea of the consequences of poorly written requirements.

TIGER Pro contains an Expert System that performs syntactic processing on text mode requirements from the testing perspective and points out requirements that may be difficult to verify, or that are written in a manner that complicate testing. TIGER Pro will not find all the defects in requirements but it does go a long way to improving written requirements.

TIGER Pro was written in Borland's Delphi (Visual Pascal). The implementation was to develop a table-driven approach, so that each user could extend the vocabulary of "poor words" without any further programming. The table is loaded from a Poor Word File which contains the following information

1. The "poor word".
2. The number of times the word was allowed to appear in the requirement. Except for the word "shall", which has to be present once to signify a requirement, the number of allowable occurrences was 0.
3. The [category of defect](#) associated with the "[poor word](#)".

## History

Dr Kasser originally developed a Requirements Workshop for classes in systems and software engineering to illustrate the writing of good requirements at University of Maryland University College (UMUC) in 1998. After joining UniSA, A/Prof Kasser incorporated it into his classes at UniSA. The workshop entailed students reviewing and commenting on a sample requirements document. Realizing that the parsing process which was taking up much of the student's time in the workshop could be automated to allow more time to discuss the content of the requirement, a prototype software tool was conceptualised and then developed to perform the function. The tool parsed the text of the requirement from the testing perspective looking for the presence of 'poor words' that complicated testing that the system's performance conformed to the requirement. The addition of TIGER into the workshop has made significant improvements. Before using TIGER in a class tutorial on requirements engineering in three postgraduate courses (subjects) the discussions in the tutorials had focussed on the structure and format of requirements. After TIGER was introduced and used to elucidate sample requirements, the focus of the in-class discussions changed to cover the difficulties of writing good requirements.

Put a TIGER in your requirements toolkit.

## 1.2 Uses for TIGER Pro

TIGER Pro has the following uses

- [Assist in writing good requirements](#)
- [Quick review of documents](#)
- [Ensuring the completeness of testing requirements contained in multiple requirement paragraphs](#)
- [Passing on lessons learned](#)

### Quick review of documents

TIGER Pro is useful for people who have to review requirements documents in short-turn around situations such as just before awarding contracts, as well as for those persons who are writing the documents. The Figure of Merit FOM can be determined as part of the review process to give a rapid determination of the quality of the document. A low FOM could indicate that the producers of the document do not understand the need. The process of clarification of the vagueness would also help identify some of the missing and wrong requirements.

## Ensuring the completeness of testing requirements contained in multiple requirement paragraphs

TIGER Pro could be used to increase the effectiveness of the planning and testing of a system with defective requirements. This example is based on the approach used to plan the testing of Build 3 of the software for the Hubble Space Telescope Data Archiving and Delivery Service (DADS) processing centre in 1993. The tests were planned so that each test would test a group of requirements. However, the requirements were defective containing many examples of "poor words" and multiple-requirements in a single requirement paragraph. Thus each requirement allocated to Build 3 had to be evaluated and split into a number of individual requirements to ensure completeness of the testing. Each "poor word" had to be identified and clarified to ensure that all parts of the defective requirements were tested. Some multiple paragraph requirements had to be tested in several different tests. This required unplanned and unbudgeted meetings between the test and design personnel and ways of documenting partial requirements in the Requirements Traceability Matrix. As an example of the work that TIGER Pro could have expedited, consider the following requirement (ST-DADS 1992):

204.1 DADS shall automatically maintain statistics concerning the number of times and the most recent time that each data set has been accessed. These same statistics shall be maintained for each piece of media in the DADS archive.

Two vague phrases had to be clarified. In the absence of any traceability to an operations concept, the following clarifications were made:

- The vague phrase "automatically maintain statistics concerning" was interpreted to mean ONLY the "total number of times" and the "most recent time". Thus it was interpreted as meaning that there was to be no test to determine if DADS kept a log of access information (times and user) as far as this requirement was concerned.
- The term "piece of media" was interpreted to mean the physical disk on which the data was stored.

The requirement was then split into the following four requirements to simplify tracking the completeness of the test plans:

204.1a DADS shall automatically maintain statistics concerning the number of times ~~and the most recent time~~ that each data set has been accessed. ~~These same statistics shall be maintained for each piece of media in the DADS archive.~~

204.1b DADS shall automatically maintain statistics concerning ~~the number of times and the most recent time~~ that each data set has been accessed. ~~These same statistics shall be maintained for each piece of media in the DADS archive.~~

204.1c DADS shall automatically maintain statistics concerning the number of times ~~and the most recent time~~ that each data set has been accessed. ~~These same statistics shall be maintained for each piece of media in the DADS archive~~ [has been accessed].

204.1d DADS shall automatically maintain statistics concerning ~~the number of times and the most recent time~~ that each data set has been accessed. ~~These same statistics shall be maintained for each piece of media in the DADS archive~~ [has been accessed].

Leaving the sections of the requirement that were not being tested in place but stricken through clearly identified which section of the requirement was being tested. An unfortunate side effect was that it also clearly showed the defects in the requirement. Note that the phrase 'has been accessed' has been moved in the last two sub-requirements to clarify the sub-requirement.

## TIGER Pro saves time

Building the Test Plan was a labour-intensive process. Each requirement had to be manually scanned for vague words, the 'and' and 'or' words, as well as further occurrences of the word "shall". Defining these words as defects, and categorising them, allows TIGER Pro to parse the document and identify requirements needing clarification and splitting to ensure a complete test. If TIGER Pro had been used

---

on the DADS program it would have saved at least 200 person-hours of test planning effort; and DADS was not a large project!

### **Passing on lessons learned**

TIGER Pro can be used to prevent time from being spent on clarifying meanings of poorly worded requirements in future projects. If the vague phrases extracted from requirements documents in one project are added to the list of "[poor words](#)", they can be prevented from appearing in future requirements documents. The students at UMUC produced documents that lacked the "poor words" they were shown. However each iteration of a class produced its own crop of new "poor words". These could be added over time thus institutionalising the lessons learned in the effect of "[poor words](#)". Management of the poor word file would be a task performed by the Quality Assurance or Test department to ensure that should someone write that phrase in the future, it would be clarified before being adopted.

# TIGER Pro

Objective requirements  
engineering

# Part



## 2 Overview

TIGER PRO performs the following functions:

- [Ingesting of requirements](#) from text documents and the keyboard into requirement databases.
- Exporting CSV format files.
- [Elucidating requirements](#) based on a set of "[poor words](#)" and pointing out up to seven [types of \(potential\) defects](#) in each of the requirements.
- Modifying existing requirements.
- [Fixing the most common types of defects](#) produced by students in Requirements workshops.
- Allowing for additional "[poor words](#)" to be added as they are identified.
- Allowing for "[poor words](#)" to be used in a requirement when their use is appropriate. For example, the requirement that "the system shall display the combined total of A and B" is a good requirement.
- Providing a [Built-in Agent using Deterministic Grammar for the Engineering of Requirements \(BADGER\)](#) that facilitates the correct format for writing requirements by, prohibiting many "poor words", and minimizing the need for retyping by the use of drop down lists.
- Producing a report documenting each occurrence of a "poor word" in the requirements.
- Producing a Figure of Merit for the goodness of a requirements document.
- Allowing users to view, document/assign and modify the following attributes(for each requirement):
  - [keywords](#) (up to three),
  - [acceptance criterion](#),
  - [rationale](#),
  - [traceability](#) (to source(s) and sideways),
  - [priority](#) (level and basis),
  - [risk](#) (severity, probability, basis and mitigation), and
  - [cost](#) (estimates, basis and totals) for each requirement.
- Producing summary reports on elucidation of requirements, priority allocation, risk assignment and cost estimation in textual and graphical formats.

# TIGER Pro

Objective requirements  
engineering

# Part



## 3 Major functions

TIGER Pro contains the following major functions

- [Ingest](#) of a text file into a requirements database.
- [Elucidating](#) a requirements database.
- [Inheriting](#) requirements from a second database.

### 3.1 Ingesting a requirements document

TIGER Pro ingests a requirements document in text and comma separated value (CSV) formats.

#### Ingesting CSV format Files

(1) FIRST set the Ingest options in the [Ingest Control Panel](#) to CSV then (2) set the attributes to be transferred using the [CSV Control Panel](#) when it appears. Note that you cannot [merge](#) into or [append](#) to a CSV file.

#### Ingesting text format documents

Requirements documents in other formats such as Word (\*.doc) have to be converted to \*.txt format before being ingested into TIGER Pro.

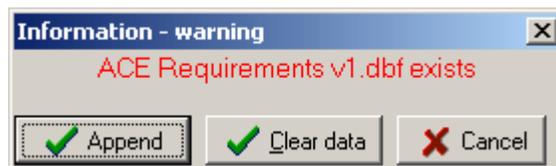
To ingest a text file into a new database (1) FIRST set the Ingest options in the [Ingest Control Panel](#) then (2) select the file to be ingested by either:

- selecting the **File | ingest Document** option from the drop down **File** Menu, or
- clicking the **Ingest Button** in the [Top Tool bar](#), or
- selecting a previous file in the drop down list that appears when clicking on the **File** Menu.
- clicking on the **Ingest Button** in the [Ingest Control Panel](#).

TIGER Pro creates a new database with the same name as the text file. Thus for example, the requirements file 'MCSS.txt' is ingested into the newly created database 'MCSS.dbf' (together with several other files).

To merge a text file into an existing database select the **File | Ingest (Merge)** option from the drop down File Menu. This option would, for example merge a file named 'STARSHIP.txt' into the existing 'MCSS.dbf' database.

Before attempting to ingest a requirements file into a new database, TIGER Pro checks to see if a database with the same name exists. If it does, you are presented with the dialogue shown below (where 'ACE Requirements v1.dbf' is the name of the database).



This dialogue asks you to confirm whether you want to:

- **append**: append new records to the current records in the existing database
- **clear**: clear all records from the existing database before adding the new records
- **cancel**: abort the ingest process.

The ingest process aborts should you try to ingest a requirement with a duplicate ID.

An Ingest Report is generated in the [Reports and Graphs Panel](#) after ingestion is complete

### 3.1.1 Format of requirements that can be ingested

TIGER Pro is designed for use in the classroom. As such it expects to find the requirements in the following formats

#### Text files

Lines beginning with an ID followed by a space (optional). A typical example could be:

```
3.4 The system shall be able to ingest a text file.
3.5 TIGER Pro shall elucidate text files exported from DOORS as ASCII text.

3.6 next requirement.
```

If there is a poor word associated with the requirement it should precede the text of the requirement prefixed by the ignore "poor word" instruction as in the following example.

```
PW: and
3.7 The system shall display the total number of apples and oranges.
```

TIGER Pro will not do very well with requirements formatted in lists such as

```
5.6 The system shall display
5.6.1 apples
6.6.2 pears.
```

#### CSV Files

See CSV Files.

#### DOORS exported files

Format the file as follows

```
Requirement on a single line (with or without numbers).
The Unique Requirements Identifier on a single line.
Empty line(s)
Next requirement on a single line...
```

Inhibitor words can be used to identify statements as non requirements.

### 3.1.2 The parts of a requirement

The parts of a requirement are:

1. The requirement Identification (ID), usually a number such as 3.4.5, but might be an alphanumerical value such as A.345.67.
2. The text of the requirement which contains mandatory and optional sections as follows:

#### Mandatory Parts

The mandatory part of the text of the requirement consists of

- the system name - e.g. "The MCSS"
- the system action - what the system does, e.g. [shall] display
- the system target - the object of the verb - e.g. the status of the control circuit

#### Optional Parts

The optional parts are the following parts of the requirements text.

- The tolerance preposition - e.g. within
- The tolerance direction - words such as 'greater than', 'less than' etc., and not used in this example
- The tolerance level - a value such as 'one'
- The tolerance units - the units of the value - e.g. second.
- The condition - or trigger for the action, e.g. after.
- The Initiator - what triggers the action - eg. the user. .
- The action - e.g. clicks on
- The target - what the action is done to - e.g. the cancel button.

If the example was entered into [BADGER](#), and the Imperative was set to "shall", BADGER would build the following requirement.

"The MCSS shall display control status within one second after the user clicks on the Cancel button."

3. [The additional attributes](#) of the requirement.

### 3.1.2.1 Poor words

#### Poor words

Poor words are certain words that, if present in a requirement would point to the presence of a [defect in the requirement](#), making the requirement unverifiable, or by virtue of having multiple requirements in the paragraph complicate ensuring the completeness of testing.

Except for the word "shall" which is allowable once since it signifies a requirement, all other poor words are not allowable in a requirement. The poor word list grows by experience, expandable by users of the TIGER Pro as they discover new ones. Any time the Test and Evaluation function have to clarify the meaning of a word, it is a candidate for the "poor words" list on future requirement documents.

A built-in poor word is the '/' slash character. It is often used in a way that incorporates two requirements in a sentence ( a short cut for the word "and"). Examples of this type of use are 'and/or', 'read/write', 'identification/validation' and 'input/output'.

However, '/' can also be used in the form of '+/- 10%' when specifying a tolerance. In this case, the use is legitimate and the poor word can be [overridden](#).

TIGER Pro stores the poor words in the Poor Word file.

[Teaching TIGER Pro New Poor words](#).

[Why teach TIGER Pro new words?](#)

[Overriding poor words](#) or teaching TIGER Pro to ignore poor words in a requirement.

### 3.1.2.2 Types of defects

#### Types of Defects

TIGER Pro can recognise up to six potential types of defects namely:

1. Multiple requirements in a line.
2. Possible multiple requirements in a line.
3. Not verifiable or testable.
4. Use of wrong word to define a requirement.

5. User defined defective word.
6. Possible design constraint.
7. Incomplete sentence.

TIGER Pro can only detect certain types of defects. For example, TIGER Pro needs to be told that the terms "statistically monitor" and "degrading media" in the following requirement are bad. Why are they bad? Well, what do they mean?

"DADS shall statistically monitor the integrity of data stored in the archive and safe-store in order to detect degrading media."

Or that the requirement

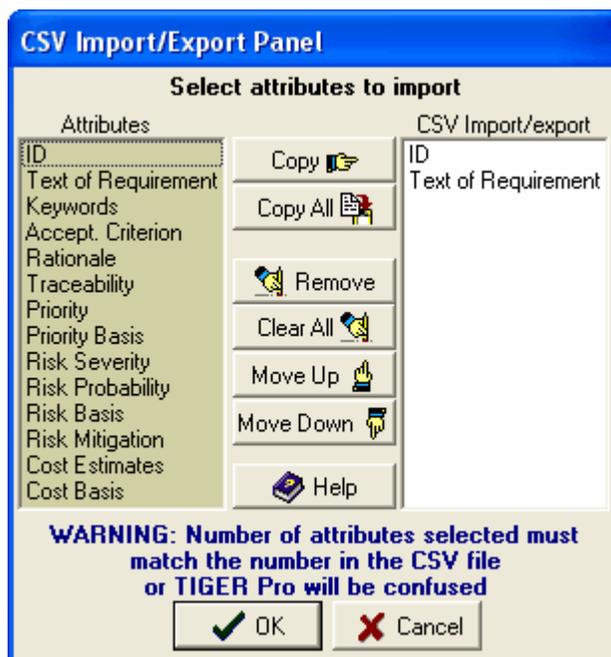
"3.6 The system shall transmit all outgoing data at a rate of 1 KBPS using Reed-Solomon encoding following receipt of a transmit command."

would be better written as the following series of requirements

"3.6 Outgoing data requirements" [comment]  
 "3.6.1 The system shall transmit all outgoing data at a rate of 1 KBPS (+/-50 BPS)"  
 "3.6.2 The system shall transmit all outgoing data using Reed-Solomon encoding."  
 "3.6.3 The system shall start to transmit the outgoing data within 10 Seconds of receipt of the command to transmit data."

## 3.2 CSV Control Panel

The CSV Control Panel allows you to set the number and order of attributes to transfer between (export and import) TIGER Pro and the CSV files. A Typical Panel is shown below.



The **Attributes Box** is read-only. Copy the desired attributes to the **CSV Import/export Box** by clicking on the desired attribute in the **Attributes Box** and then clicking on the **Copy Button**.

To remove an attribute from the **CSV Import/export Box** click on the desired attribute and then click on the **Remove Button**.

To clear all entries in the **CSV Import/export Box** click on the **Clear All Button**. The ID and Text of Requirement are always in the box unless specifically deleted.

Use the **Move Up** and **Move Down Buttons** to change the order of the attributes listed in the **CSV Import/export Box**.  
clicking

The **Help Button** shows this information.

If no attributes are selected in the **CSV Import/export Box**, TIGER Pro will assume that the **ID** and

Text of Requirement are to be transferred.

### 3.3 Elucidating a requirements document

Elucidating a requirements document or database identifies requirements containing words that make testing and verification difficult. TIGER Pro scans each requirement and generates a Figure of Merit for the entire requirements document.

To elucidate a database either:

- Selecting the **File | Database | Elucidate** option from the drop down Main Menu, or
- clicking the **Elucidate Button** in the [Top Tool bar](#), or
- clicking the **Elucidate Button** in the [Elucidate Control Panel](#).

When you choose one of the above options, a standard Windows file selection menu will be shown. Select the file to be elucidated. TIGER Pro will then start to elucidate the file.

TIGER Pro expects requirements to be written in a [simple format](#).

When the elucidation function is complete, TIGER Pro will display a summary in the Report area and copy the summary into a Summary file.

### 3.4 Working with the elucidation results

After TIGER Pro has elucidated a database, you have to examine the report and determine if the requirements are defective.

The Elucidation Report in the [Reports and Graphs Panel](#) text mode display should be at the bottom of the report panel displaying the summary. If not click on the **Reports Graphs Tab** in the [Reports and Graphs Panel](#), and then on the **Ingest/Elucidate Tab**. If the display is a graph, click on **Text** in the **Report Area** of the [Elucidate Control Panel](#).

Scroll up through the report to display the errors associated with the individual requirements. If you have difficulty locating requirements with errors, click on **All** in the **Display Area** of the [Elucidate Control Panel](#) and perform the [elucidation function](#). This configuration will generate a report limited to the requirements containing defects.

A typical error display could be as shown below

```
9.9.28 The MCSS shall be able to operate using under-floor plenum cooling air
between 65 and 80 degrees Fahrenheit; relative humidity will be between 35 and 70
percent non condensing.
and defect type 22 times
will defect type 4
```

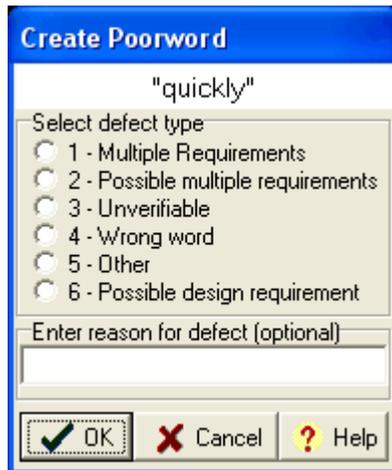
You will have to change wording of the requirement to remove the defects. To select the requirement move the cursor to the line containing the requirement ID and double click on the mouse. Alternatively use the controls on the [Select Requirement Panel](#). Edit the text of the requirement and post it either by using the Post Button on the [Select Requirement Panel](#) or advancing to another requirement.

If you identify a poor word that TIGER Pro didn't recognise you may [teach TIGER Pro the new word](#).

TIGER Pro can [fix some defects](#) in the requirements.

### 3.4.1 Teaching TIGER Pro new Poor Words

To teach TIGER Pro a new [poor word](#) found in a requirement, select the requirement, then position the mouse cursor over the poor word in the text and double click on it. TIGER Pro will then show the following display panel **if the word is not** in its poor word file. In this example, the poor word to be taught is "quickly".



The [poor word](#) shows up in the top panel to confirm that TIGER Pro has selected the correct word.

Select the [defect type](#) by clicking on the appropriate radio button.

Enter the reason why the word is poor. This is optional.

Click on the **OK Button** to teach the poor word to TIGER Pro.

Click on the **Cancel Button** to cancel the operation.

Click on the **Help Button** to show this information

If you fail to select the defect type, and click on the **OK Button**, TIGER Pro will not add the word to the poor word file.

If the poor word you wanted to add, is in TIGER Pro's poor word file, TIGER Pro will politely thank you for suggesting it.

### 3.4.2 Overriding poor words

From time to time, TIGER Pro will find a [poor word](#) whose use is correct. As an example consider this requirement.

3.4.5 The system shall display the number of inputs and outputs.

TIGER Pro will pick up the word 'and' as a potential defect. Does the word mean two individual displays or a combined total? If the intent is to display the combined total, then the user should rewrite the requirement to remove the ambiguity as

3.4.5 The system shall display the combined number of inputs and outputs.

To override the poor word, simply write the [Poor word](#) in the **Poor Word: Box** at the top of the [Requirement Panel](#). The box location is at the end of the panel as shown below.



TIGER Pro can only override one [poor word](#) in each requirement.

## 3.5 Inheriting from a second requirements document

One of the advantages of the object-oriented approach is the ability to inherit from one object to another. TIGER Pro provides the ability to inherit requirements from a second database.

[Before inheriting requirements](#), you should select how they will be inherited using the controls on the

**Inherit Panel** which contains the following areas

- **The name of the open database** containing the requirements to be inherited.
- **Data view** - containing the following sub areas:
  - a read-only display of the requirements in the same style as the [View Data Panel](#).
  - [Select Requirement Panel](#) without any editing or addition functions.
  - [Display Control Panel](#).
- **Attributes to be inherited.** A set of Check Boxes to allow you to select which attributes are to be inherited. The Requirement Text and distinction between requirement and non requirement attributes are always inherited, so are not selectable. There is also provision to tick and untick all boxes.
- **Action Buttons** - the following are provided
  - **Inherit** - initiates the inherit function.
  - **Help** - displays this panel
  - **Close** - closes the panel.
- **Inherit controls.** This is a context sensitive panel containing control settings for what TIGER Pro is to do while inheriting requirements. The main choices are:
  - [Mode of operation.](#)
  - [Duplicate requirement ID Options.](#)
  - [Replace Text.](#)
  - [Replace ID prefix.](#)

### 3.5.1 Mode of operation

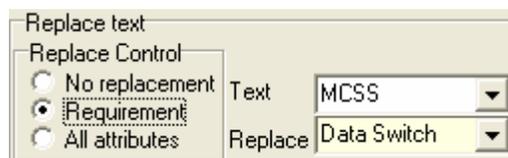
This defines which requirements are to be inherited. The choices are:

- **All requirements** - inherits all requirements in the database to the main database.
- **Selected requirement** - inherits the selected requirement.
- **By keywords** - inherits requirements containing the selected (combinations of ) [keywords](#).
- **By ID** - inherits requirements with ID values starting the with the selected ID. For example, if 3.12 is selected, then all requirements that begin with 3.12 (3.12, 3.12.1, 3.12.2 etc.) will be inherited.

### 3.5.2 Replace Text

**Replace Text:** This option allows you to replace some text in the requirement being inherited. You can use this to change the name of the system.

For example, if requirements are being inherited from a database for the MCSS to a database for a Data Switch, then select this option by selecting the **Requirement Radio Button** in the **Replace Control Panel in the Replace text Panel** as shown below, the word "MCSS" will be replaced by the words "Data Switch" when the requirement is inherited. Similarly, if the **All Attributes Radio Button** is selected, then the word replacement will be performed on the word in all the [attributes](#) of the requirement.



### 3.5.3 Duplicate requirement ID Options

**Duplicate requirement ID Options** specify **how** data to be ingested should there be any requirement IDs that already exist in the working database. The options are:

- **manual:** ingest the duplicate requirement but renames its ID by another ID that you specify. When

this option is selected, every time TIGER Pro detects a duplicate ID, it will prompt you for the new ID. This option is designed for inheriting a small number of requirements, since it can be time consuming if you have to manually renumber every requirement in the database.

- **overwrite:** overwrites the existing requirement text and other properties with the requirement being inherited.
- **ignore:** ignores the requirement containing the duplicate ID.

### 3.5.4 Replace ID prefix

**Replace ID prefix** allows you to automatically replace ID prefixes. In the example shown below the prefix 3.2 will be replaced by 3.6. This is an indirect way of [renumbering or changing the prefix of the requirement ID](#).

## 3.6 Adding a new requirement

### Adding a new requirement

You may add new requirements by:

- Using the **BADGER** Wizard, by clicking on the **Badger Button** in the Panel shown. The panel is located directly below the [Select Requirement Panel](#).



- Entering them manually by clicking on the **New Button** and entering the appropriate information into the **ID** and requirement fields. Don't forget to untick the **Requirement Check box** if you are entering a comment or other non-requirement.

After clicking on the New Button, an entry box for the ID will appear over the (read-only) ID field in the [Requirement Panel](#) as shown below. Enter the ID into this box.

The Panel with the enterable ID looks like this.

The regular display is shown below. Note the bevel in the ID: panel in the example shown above and the different [background colours](#) in the ID: and Date: fields in each example.

If the **Autoincrement ID Check Box** located below the [Select Requirement Panel](#) is checked, then TIGER Pro will automatically create a new requirement ID based on the currently selected requirement ID.

Once TIGER Pro has configured the [Requirement Panel](#), enter the text of the requirement and any other pertinent attributes. After entering the information, **post the requirement into the database by either** clicking on the **Post Button** in the [Select Requirement Panel](#) or by [selecting another requirement](#).

### 3.7 Changing Requirement IDs

You can replace Requirement ID prefixes as follows.

1. Close the working database using the **Close Button** in the [Top Tool Bar](#) or the **File | Database | Close Options** in the [Main Menu](#).
2. Create a new blank database using the **File | Database | New Options** in the [Main Menu](#).
3. Open the [Inherit Panel](#) using the **Inherit Button** in the [Top Tool Bar](#) or the **File | Inherit from a second database Options** in the [Main Menu](#).
4. On the **Inherit Panel**, set the [Mode of Operation](#) to **All Requirements**.
5. On the **Inherit Panel**, set the [Duplicate ID](#) to **Ignore**.
6. On the **Inherit Panel**, tick the **Replace Check Box** in the [Replace ID Prefix Panel](#).
7. On the **Inherit Panel**, in the [Replace ID Prefix Panel](#) enter the existing ID prefix into the **ID prefix** box.
8. On the **Inherit Panel**, in the [Replace ID Prefix Panel](#) enter the new ID prefix into the **Replace** box.
9. On the **Inherit Panel**, click on the **Inherit Button**.

This will provide a renumbered database with a different name.

# TIGER Pro

Objective requirements  
engineering

**Part**



**IV**

## 4 The Main window areas

The Main window contains the following areas.

- [Top Toolbar](#)
- [Requirement Panel](#)
- [Select Requirements Panel](#)
- [Display Control Panel](#)
- [Reports and Graphs](#)
- [View Data](#)
- Additional Attributes
- [Ingest Control Panel](#)
- [Elucidate Control Panel](#)

### 4.1 Top Tool Bar

The Top Tool Bar shown below is located directly under the [Menu line](#) on the Main display screen. Each of the buttons provide quick implementation of TIGER Pro functionality as described below.



- Open a database.
- Close a database.
- [Ingest a requirements file.](#)
- [Elucidate database.](#)
- [Editor - Files/Poor words.](#)
- [Inherit requirements from a second database.](#)
- **Saves** (exports) a summary of the requirements database followed by the contents of the requirements database to a text file (\*.txt). This is the equivalent to selecting the **Report | Save text | [All records](#)** option in the **Main Menu**
- **Prints** a summary of the requirements database followed by the contents of the requirements database. This is the equivalent to selecting the **Report | Print | [All records](#)** option in the **Main Menu**
- **Enables and disables the [sounds](#)** (instructor's comments).
- **Displays the "About"** information.

The full directory path and name of the current database file is displayed in the remainder of the Top Tool Bar.

#### 4.1.1 Adding key words to the database

To add keywords to a requirement use the [Keywords Panel](#).

### 4.2 Requirement Panel

A typical Requirement Panel is shown below.

The top area contains the:

- **Requirement Check box** - check indicates that TIGER Pro considers the paragraph to be a requirement. You may check or uncheck the box. The actual requirement displayed in the Requirement Panel is selected via the controls on the [Select Requirement Panel](#).
- The ID number of the requirement. This is a read only value.
- The date the requirement was written or was last changed. This is an automatic function within TIGER Pro.
- The [Poor Word](#) (if any) associated with the requirement. You may enter one, edit it and delete it as appropriate.

The remainder of the Panel contains the text of the requirement. When working with the text,

- You may [teach TIGER Pro a new Poor Word](#) by double clicking on the word in the text of the requirement.
- You may activate [BADGER](#) by right clicking the mouse when the cursor is positioned on the requirement text.

Note the colour of the data areas are different for requirements and non requirements (comments).

#### 4.2.1 Keywords Panel

The Keywords Panel shown below displays the keyword attribute of the requirement and allows you to enter/edit up to three Keywords for a requirement.

The Top section allows you to enter/display up to three keywords for the requirement.

The Auto create section allows you to create keywords for a number of requirements in the following manner.

- To assign a keyword to **all requirements containing a specific word**, enter the word in the Auto create area, enter the keyword in the top entry box of the top section and click on the **Assign Button**.
- To assign a keyword to **all requirements containing that word**, enter the word in the Auto create area, and click on the **Create Button**.

Keywords in the requirements database may be selected from the drop down lists associated with each Keyword entry box.

I suggest that you select the [View Data Panel](#) before entering keywords to quickly demonstrate the

creation of keywords.

### 4.3 Select Requirement Panel

The Select Requirement Panel shown below is located next to the [Requirement Panel](#).



You may select the requirement to be displayed in any one of the following ways.

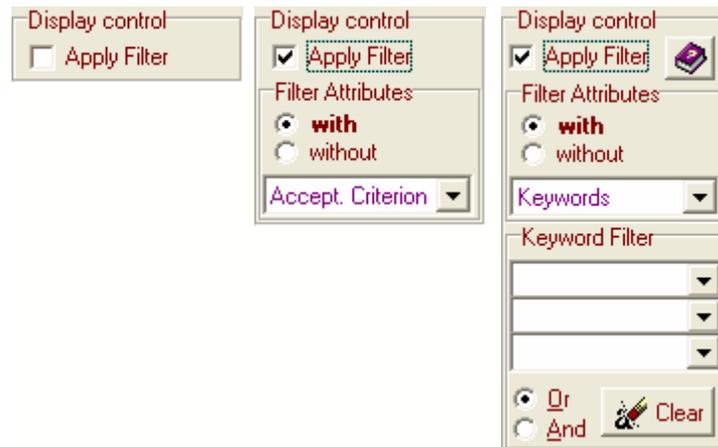
- Entering the identification (ID) number of other value of the requirement in the top box and pushing the "Enter" key.
- Selecting the ID via the drop down list box (showing 5.10.3) in the example of the panel shown above.
- Clicking on the navigation buttons (First requirement, previous, next, and last).
- Scrolling to, and then double clicking on, the line of the requirement text containing the ID number in the Report displayed in the [Reports and Graphs Panel](#).

The remaining buttons in the Select Requirements Panel are used to:

- Delete a requirement.
- Cancel an edit.
- Post an edited or new requirement into the database.

### 4.4 Display Control Panel

The Display Control Panel is context sensitive and takes the forms shown below.



When the **Apply Filter Check Box** is unchecked, the short form display is displayed.

When the **Apply Filter Check Box** is checked, the selection menu appears. You may chose to display requirements with or without various Attributes. If you select **with**, then only requirements with that attribute will be displayed. If you select **without**, then only requirements without the attribute will be displayed.

If you select the [Keywords](#) Filter Attribute, then the display expands to allow you to select up to three [Keywords](#). You may then choose to display requirements containing combinations of [Keywords](#)

(or/and) by entering them into the **Keyword Filter selection boxes**. The **Clear Button** allows you to clear the contents of the **Keyword Filter selection boxes**.

## 4.5 Summary Panel

The Summary Panel consists of two parts as follows:

- A control area to configure how the summaries are updated. If the **Automatic Check Box** is checked, the summaries are updated automatically each time a requirement is changed, if it is not checked, the summaries are only updated after an elucidation function or the **Manual Button** in the Panel is clicked. When working with large databases, the updating function may take time (depending on the speeds of the system processor and disk drive) and should be turned off.
- A context sensitive display area. The attribute summary is only displayed when the [Apply Filter Check Box](#) in the [Display Control Panel](#) is checked.

## 4.6 Reports and Graphs Panel

A typical reports and graphs panel containing a summary report of the elucidated requirements is shown below.

The screenshot displays a software interface with a main report area and a sidebar. The main area shows a summary of requirements and defects. The sidebar, titled 'Elucidation Results', provides a detailed breakdown of defect types and a summary of the overall results.

Ingest/Elucidate	Priority	Risk Severity	Risk Probability	Risk Level	Cost Estimates
/EX					
-----					
Summary					
All Records:	246				
Requirements:	246				
Ignore PW:	0				
Number of Defects:	228				
Defect type 1 - Multiple requirements in a line:	4				
Defect type 2 - Possible multiple requirements:	115				
Defect type 3 - Not verifiable word:	86				
Defect type 4 - Use of wrong word:	23				
Defect type 5 - User defined poor word:	0				
Defect type 6 - Possible design requirement:	0				
Defect type 7 - Incomplete sentence:	0				
Figure of Merit:	8				
*** Elucidation completed at 05-Sep-2006 15:13					

Elucidation Results	
<b>Defects</b>	
1- Multiple reqs	4
2- Poss. multi.	115
3- Unverifiable	86
4- Wrong word	23
5- User defined	0
6- Poss. design	0
7- Incomplete	0
<b>Summary</b>	
All Records	246
Requirements	246
Ignore PW	0
Defects	228
<b>Fig. of merit</b>	<b>8</b>

Buttons: Clear, Help

The Panel contains the following areas.

- **Content Control Tabs** which allow you to select [Reports and Graphs](#), [View Data](#) and enter content into the Additional Attributes
- **Report Selection Tabs** which allow you to select to display reports concerning the [Ingest/Elucidate](#) functions, and the distribution of [priority](#), [risk severity](#), [risk probability](#), and [risk level](#) properties, or a summary of the [cost estimates](#) property associated with the requirements.
- **A report about the last ingest or elucidation function.** In the example shown above, the report summarises the last elucidation function.

- **Summary Displays** of the number and [type of defects](#) in the requirements document. If you click on the **defect names** (eg. 5 - User defined) in the **Defects Panel**, you will get further information about each type of defect.

The **Clear Button** clears the report and resets the counter displays to 0.  
The **Help Button** brings up this screen.

[Working with the elucidation results.](#)

#### 4.6.1 Reports and Graphs

The Reports and Graphs area shows text and graphic summaries of the number of defects in a requirements database and graphical reports of the other attributes of the requirements as selected by the Report Selection Tabs.

The [Ingest Control Panel](#) and the [Elucidate Control Panel](#) are only visible when the **Ingest/Elucidate Tab** is selected.

The following graphical reports are available :

- The distribution of defect types, namely the total number of each type of defect in the requirements document (after an [elucidation function](#)). This display is enabled by the **Graph Button** in the [Elucidation Panel](#).
- The total number of requirements with each [priority](#) level. This is useful in ensuring that the priorities are evenly distributed.
- The total number of requirements with each [risk severity](#) level.
- The total number of requirements with each [risk probability](#) level.
- The total number of requirements with each [risk](#) level.

A report of the total cost estimates is available in the form of a spread sheet.

#### 4.6.2 The View Data Panel

The view data panel displays all the [attributes](#) of the requirement object in **read-only** form. For [attributes](#) containing a lot of text (e.g. requirement, [acceptance criteria](#), etc) only (up to) the first 50 characters are displayed. Part of a typical display is shown below.

ID	Requirement	Req Date	Poorword	Keyword1	
5.1	MCSS SYSTEM DESCRIPTI	17/08/2006			
5.10	TECHNICAL SUPPORT	17/08/2006			
5.10.1	A software users guide.	17/08/2006			
5.10.10	Test plans and test procedure	17/08/2006			
▶ 5.10.11	The MCSS hardware and sof	17/08/2006			
5.10.12	All identical parts shall be at t	17/08/2006			
5.10.13	At least 10 percent spares (m	17/08/2006			

The foreground and background colour of the text indicates one of three states:

- A non-requirement or comment - shown in the same background colour as in the [Requirement Panel](#).
- A good requirement - shown in the same background colour as in the [Requirement Panel](#).
- A defective requirement - shown here with red foreground text and the same background colour as a requirement.

## 4.7 Ingest Control Panel

The Ingest Control Panel shown below allows the selection of the Ingest options



The Ingest Control Panel provides you with the following options for ingesting a document:

### Mode: Applies to Text and CSV files

- Choose **All** to instruct TIGER Pro to treat every sentence in the requirements text file as a requirement.
- Choose **By Word** to instruct TIGER Pro to treat only sentences containing the words 'shall', 'will', 'should' and 'must' as requirements.

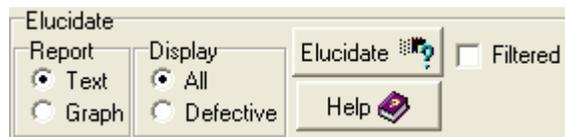
### Format:

- Choose **Text** to select a text file
- Choose **DOORS** to select a text file exported from DOORS.
- Choose **CSV** to select a comma separated value (\*.csv) file.

The **Ingest Button** initiates an [ingest](#) function. The **Help Button** brings up help information about [ingesting a requirements document](#).

## 4.8 Elucidate Control Panel

The Elucidate Control Panel shown below allows the selection of the Elucidation options.



The Elucidate Control Panel provides you with the following options for elucidating a document:

### Report:

- Choose **Text** to display the results of the elucidation in text format in the [Reports and Graphs Area](#).
- Choose **Graph** to display the results of the elucidation in bar chart form in the [Reports and Graphs Area](#).

### Display:

Choose **All** to copy all requirements to the [Reports and Graphs Area](#) as they are elucidated. Choose **Defective** to only copy requirements with defects into the [Reports and Graphs Area](#) as they are identified during elucidation.

The **Elucidate Button** initiates an [elucidation](#) function. The **Help Button** brings up help information about [elucidating a requirements document](#).

The **Filtered Check box** controls which requirements in the database are elucidated as follows:

- If checked, then only the requirements containing [attributes](#) selected by the **Apply Filter Check Box** in the Requirements [Display Control Panel](#) are elucidated.
- If not checked then all the requirements are elucidated irrespective of the state of the **Apply Filter Check Box** in the Requirements [Display Control Panel](#).

# TIGER Pro

Objective requirements  
engineering

# Part



## 5 The Menu

This area contains a drop down menu for all available functionality grouped as follows:

- [Files](#)
- [Tools](#)
- [Settings](#)
- [Report](#)
- Export
- [Help](#)

### 5.1 File Menu

File menu contains:

- Ingest Document - begins the [ingest sequence](#) creating a new requirements database in which to store the document.
- Ingest (Merge) - [ingests a text document into the existing requirements database.](#)
- [Database](#) - provides a number of options
- Inherit from second database - begins the [inheritance sequence.](#)
- Exit - terminates the program.

TIGER Pro also remembers the last file opened in three file history lists in files. This feature allows you to select a file by clicking on the list entry, hence saving you navigating to the file location. The file history lists show up in the Menu as options.

To clear the file history lists and delete the files containing the history lists, use the **Tools | Clear Files name list** option.

#### 5.1.1 Database

This option displays a sub-menu which provides the following functions:

1. **Elucidate:** initiates the [elucidation function.](#)
2. **Open:** allows you to select a requirement database to open. If the database exists, TIGER Pro will close any open database before opening the selected database.
3. **New:** allows you to create a new and blank database.
4. **Rename:** allows you to rename a database.
5. **Save As:** allows you to save the open database under a different name. This use useful when working on a database so as to be able to save intermediate work.
6. **Clear:** allows you to clear the contents of the selected database.
7. **Delete:** deletes the selected database after prompting for a confirmation message.
8. **Close:** closes the selected database.

## 5.2 Tools Menu

The Tools menu contains the following options:

1. **Create Keyword File** - instructs TIGER Pro to gather all keywords in the current database and save it in a keyword file. TIGER Pro normally performs this function automatically when closing a database.
2. **Show BADGER Template.**
3. **Fix defective requirements** - allows you to fix the following type of defects:
  - **Replace 'will, should & must'**. Changes all instances of ' will ', ' should ' and ' must ' in every requirement in the database to ' shall '. These are **Type 4 defects**. The function only works on lower case words in the middle of the requirement sentence.
  - **Requirements starting with the word 'Shall'**. (Some Type 7 defects) TIGER Pro will prompt you for the name of the system (the words) to insert before the "shall". TIGER Pro will then insert those words in every requirement beginning with the word "shall" (or 'Shall').
  - **Insert system name + imperative**. This is for requirements that lack [all of the] the words 'shall', 'will', 'should' and must (in lower case). TIGER Pro will prompt for the system name (e.g. The Starship) and insert the subject followed by the imperative word 'shall' at the front of the requirement.
  - **Add a leading zero to numeric Requirement ID with numbers less than 10**. This is to facilitate the sorting of requirements in the classroom. TIGER Pro will generate an error message of the requirements ID is not a number.
1. **Pack database** - instructs TIGER Pro to pack the open database. Use this function to reduce the size of a database after deleting a large number of records.
2. **Test database integrity** - instructs TIGER Pro to sequentially open every record in the opened database. TIGER Pro will notify you if it has a problem reading any records.
3. **Test database structure** - allows you to select a database and test its structure. This function is designed to be used to identify databases created previous versions of TIGER and other PETS.
4. **Update Database Structure** allows you to update the structure of databases created with previous versions of TIGER and other PETS to contain all current data fields. You will be sequentially prompted to enter the name of the database you want to update, and the name you want to assign to the updated database.
5. **Editor - Files/poor words.**
6. **Clear File names list** - allows you to select which file history list to clear, and which file containing the list to delete.

### 5.2.1 The BADGER Template

BADGER (**B**uilt-in **A**gent for **D**eveloping **G**ood and **E**ffective **R**equirements) is a requirement builder template which takes you through a step by step the process for building a requirement. BADGER can also be activated by right clicking on the requirement text in the [Requirement Panel](#) or by clicking on the [Badger Button](#) below the [Select Requirement Panel](#).

To build a requirement, you need to:

- Verify or ensure the **Requirement Check Box** is checked for a requirement or unchecked for a comment.
- Set the **Imperative** to "shall" or "shall not" as appropriate.
- Fill in the **mandatory parts of the requirement** and any **optional parts** that are appropriate. In this example BADGER would create the requirement "The MCSS shall eat the banana within no less than three seconds after the user clicks on the cancel button." Each entry box contains a drop down list of previously used terms to save you from retyping them.

**It is the drop down feature in the entry boxes that speeds up the entry of requirements, because much of the typing is eliminated when entering a series of requirements containing repetitive text.**

### Using the Buttons

- Click on the **Create Button** to create the requirement. When you have done, so the requirement is displayed in the **Requirement Panel**, and BADGER configures itself for writing the next requirement.
- Click on the **Cancel/Clear button** to cancel or clear an incomplete requirement.
- Click on the **Close Button** to close down BADGER.
- Click on the **Help Button** to display this information.

## 5.2.2 Editor - Files/Poor Words

The Editing Panel contains the following areas:

- **Menu selection.** This allows you to:
  - **Close** or hide the Editor Files/Poor Words Panel
  - **Edit a number of files.**
  - **Load** the poor word or inhibitor file.
  - **Sort** the poor word file.
  - **Tools** - provides the following option:
    - **Show file names in Edit Menu** - when checked, the file names are added to the File Designators in the **Edit Menu**, when not checked, the file designators are only shown.
- **Poor words (read only listing)** - this is a **read only** listing of the **poor words** known to TIGER Pro, the number of times each poor word is allowed to occur in a requirement, and the defect type. In

general, all poor words are not allowed to occur in a requirement. The sole exception is the word "shall" which must occur "once in a "requirement". If the word "shall" occurs twice, then there are two requirements embedded in the requirement. [If you click the mouse on a row in the listing, and the Auto edit Check Box is ticked, TIGER Pro will open up the editor to being editing the Poor word File at the line containing the poor word.](#)

- [File Control Panel](#).
- **Auto edit Check Box** - enables automatically opening the editor at the poor word when clicking on a row in the poor words panel. The box will untick itself each time the automatic edit function is performed. This is to avoid accidentally editing the file and destroying its contents.
- **Help Button** - displays this information.
- **Close Button** - hides the panel.

### 5.2.2.1 Edit a number of files

This option allows you to select the following files and edit them.

- Inhibitor File.
- **Last file edited.** TIGER Pro remembers the name of the file you last edited and prompts you with it. This saves you having to repeatedly navigate to it when working with text files.
- Poor word File.
- Summary File created last.
- Any file (\*.\*)
- [BADGER](#) Files (\*.wil).
- CSV Files (\*.csv).
- Key word files (\*.kwd).
- Poor word files (\*.frd).
- Requirements files (\*.txt).
- [Saved requirements files](#) (\*.prn).
- Summary Files (\*.frd).
- TIGER Pro.ini - the configuration file. **This option is designed to let you look at, but not edit the file, since TIGER Pro will save its configuration, overwriting any changes you make with the editor.**

The state of the [Show file names in Edit Menu](#) menu option in the **Tools** option determines if the actual names of the Inhibitor, Last file edited, Poor word file and Summary file are shown in the Menu.

### 5.2.2.2 File Control Panel

This panel contains context sensitive controls that let you:

- Select, load, sort or edit a Poor word File.
- Select a Summary File.
- Select or load a DOORS ingest Inhibitor File.
- View the Inhibitor words in a read-only panel
- View and change the default Object Identifier used when [ingesting a requirements file](#) exported from DOORS.
- View and change the Ignore "poor word" instruction for ingesting text and csv files.

The context for the display is determined by state of the [Format Radio Buttons](#) in the [Ingest Control Panel](#).

## 5.3 Settings Menu

The Settings menu contains the following options:

- [Font](#)
- [Set Background Colour](#)
- Panel position - allows you to configure TIGER Pro to display the [Select Requirement](#), [Display Control](#) and [Summary](#) Panels to the left or to the right of the [Requirement Panel](#) in the [Main Window](#) and [Inherit Panels](#), and the position of the [File Control Panel](#) in the [Edit Tools Panel](#).
- [Sounds](#)
- Printer Setup
- Auto Reopen Last File
- **Show file directory path** - controls how the filenames are displayed in the selection/display panels in the [Sounds Panel](#), the [Edit File Control Panel](#) and in the requirements database name displays at the top of the [Main Window](#) and [Inherit Panel](#). If checked, then the full directory path is shown. If unticked, then only the name of the file is shown. Clicking on the menu option checks and unchecks the option. The full paths are always shown in the file history lists to allow you to be sure of exactly which version of which file you are reselecting.

### 5.3.1 Font

This option allows you to set or change the font of the:

- [Requirement statement](#)
- [Report \[Ingest/Elucidate\]](#)
- **Requirement/Report** - changes the font of the [Requirement statement](#) and the [Report](#). This option is designed for use in the classroom.
- [Acceptance Criteria](#)
- [Cost Estimates](#)
- [Priority](#)
- [Rationale](#)
- [Risk](#)
- [Traceability](#)

When one of the above options is chosen, TIGER Pro will display a **Windows Font dialog** that allows you to customize font type, size, style, colour and other standard effects. **Note:** Your setting automatically updates TIGER Pro's configuration as soon as you click the **OK button**.

### 5.3.2 Set Background Colour

This option allows you to set background colour of the:

- [Requirement panel](#)
- [Acceptance Criteria](#)
- [Rationale](#)
- [Traceability](#)
- [Priority](#)
- [Risk](#)
- [Cost estimates](#)
- [Report and Graphs area](#) (text part).
- **Read only panels.** These are data panels that are read only. The colour allows you to visually identify them.
- **Prompt information.** These are data panels that require you to select from the options provided.

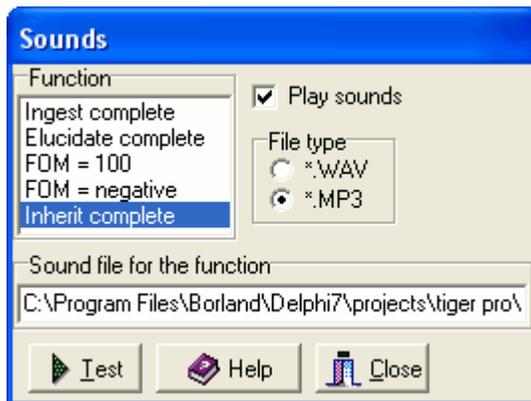
When one of the above options is chosen, TIGER Pro will display a **Windows Colour dialog** that allows you to customize the selected background colour. **Note:** Your setting automatically updates TIGER Pro's configuration as soon as you click the OK button.

### 5.3.3 Sounds

TIGER Pro generates sounds for the Following functions.

- [Ingest function](#) completed
- [Elucidation function](#) completed with a
  - Figure of Merit of between 0 and 100.
  - Figure of Merit of exactly 100.
  - Figure of Merit of less than 0.
- [Inherit function](#) completed.

The Sounds Panel shown below allows you to customise the sounds as follows:.



- Check the **Play Sounds Check box** to turn the sounds on or off.
- Click on the line in the **Function** area to see the directory path and name of the sound file TIGER Pro will play when the function completes.
- Click on the **Test Button** to hear the sound selected for the function.
- Click on the **Close Button** to hide the Sounds Panel.
- Click on the **Help Button** to show this information.
- Click on the **Sound file for the function text box** to display a **Windows File Selection Menu** to allow you to set the sound file associated with the function. Control the display of WAV or MP3 files in the **Windows File Selection Menu** via by clicking on the appropriate file type in the **File type** area.

## 5.4 Report Menu

This option allows you to

- **Update Reports** - update all summary and graphical report displays.
- **Save text** - allows you to save (export) the contents of the requirements database to a text file (\*.prn). There are sub options to this menu choice, namely:
  - **Ingest/Elucidate** - saves the contents of the report area to text file.
  - **All records** - saves all records in the database.
  - **Filtered records** - saves records with the property selected in the [Display Control Panel](#). Each saved field is prefixed by an identifying word to facilitate importing into other programs.
- **By Keyword** - summarises and saves records containing the selected [keywords](#).
- **Save graph** - allows you to select and save the graphs of the numbers of each type of defect discovered in the elucidation function, the priority level profile, and the risk severity, probability and level summaries to picture files.
- **Print** - allows you to print a report containing the same summaries and contents of the database as selectable by the Save text option.
- **Clear** - this option performs the same function as the [Clear Button](#) in the [Reports and Graphs Panel](#).

Before printing or saving text, TIGER Pro will prompt you to select which properties you want in the saved report or file. The prompt screen, an example of which is shown below, provides you with a list

of properties, the ability to select (by tick the box) individual properties or selecting/deselecting all properties.

**Save Control**

Please tick the properties that you want to save

<input checked="" type="checkbox"/> Requirement ID	<input type="checkbox"/> Priority
<input checked="" type="checkbox"/> Requirement/Comment	<input type="checkbox"/> Priority Basis
<input checked="" type="checkbox"/> Requirement Text	<input type="checkbox"/> Cost Basis
<input checked="" type="checkbox"/> Requirement Date	<input type="checkbox"/> Cost Estimate
<input type="checkbox"/> Poor word	<input type="checkbox"/> Cost (low)
<input type="checkbox"/> Keywords	<input type="checkbox"/> Cost (Mean)
<input type="checkbox"/> Acceptance Criteria	<input type="checkbox"/> Cost (high)
<input type="checkbox"/> Acceptance Criteria Date	<input type="checkbox"/> Risk Severity
<input type="checkbox"/> Rationale	<input type="checkbox"/> Risk Probability
<input type="checkbox"/> Traceability	<input type="checkbox"/> Risk Level
	<input type="checkbox"/> Risk Basis
	<input type="checkbox"/> Risk Mitigation

Tick all properties  
 Untick all properties

Click on the **OK Button** to print/save the properties, or on the **Close Button** to cancel the function.

## 5.5 Export Menu

The Export menu provides the following option

- **Export CSV** - export the ID and text of the requirement into a CSV file. The name of the file will be the same as the name of the requirement database with the suffix changed to 'csv'. For example, if the ID and text are exported from the MCSS database, the exported file will be named MCSS.csv. Should the CSV file exist, TIGER Pro will prompt you to confirm that you want to overwrite it.

## 5.6 Help Menu

The Help Menu provides the following options:

- **Table of Contents** - displays the contents of the Help file.
- **Full Text Search** - allows you to find words in the text of the Help file.
- **Help on Help** - tells you how to use Help.
- **Search index** - allows you to search the index of the Help file.
- **Register** - allows you to [register your copy of TIGER Pro](#). This option only shows up if you are using the Shareware version. To register the licence, copy the text of the Registration Code received in response to purchasing a license into the dialogue panel that opens up after selecting this option.
- **About TIGER Pro** - displays information about TIGER Pro as well as and contact information.

# TIGER Pro

Objective requirements  
engineering

**Part**



## 6 Attributes of a requirement

In current research on the properties of object-oriented requirements, the candidate attributes or properties are:

- [Acceptance Criteria.](#)
- [Cost Estimates.](#)
- [Key words.](#)
- [Priority.](#)
- [Rationale for the requirement.](#)
- [Risk.](#)
- [Traceability.](#)

### 6.1 Acceptance criteria

Acceptance criteria are provided in response to the question "How will we know that the requirement has been met by the system?"

These could be as simple as "demonstration" for simple requirements or involve various degrees of analysis for more complicated or expensive requirements. For example, what are the acceptance criteria for electrical fuses? Pass the specified electrical current through it and demonstrate that the fuse blows within the required time. OK. that was a good one, but it is now useless.

#### Entering Acceptance Criteria into TIGER Pro.

Select the **Acceptance Criterion Panel** by clicking on the **Additional Attributes Content Control Tab** in the [Reports and Graphs Panel](#) and then clicking on the **Acceptance Criterion Tab**. TIGER Pro will then display the **Acceptance Criterion Panel**. Enter the information by typing it into the panel.

TIGER Pro will automatically assign a date to the Acceptance Criteria.

### 6.2 Cost estimates

These feed into the management plan and are refined as the project passes through the development life cycle. The level of confidence in the cost estimates should improve as time passes.

The following information may be entered into the Cost property

- **The Basis for the Cost estimate**, entered as text in the Basis for the Cost Estimate Panel.
- **High and Low** cost estimates.

#### Entering Cost Estimate information into TIGER Pro.

Select the Cost Estimates Entry Panel by clicking on the **Additional Attributes Content Control Tab** in the [Reports and Graphs Panel](#) and then clicking on the **Cost Estimates Tab**. TIGER Pro will then display the Cost Estimates Entry Panel shown below.

Acceptance Criterion	Rationale	Traceability	Priority	Risk	Cost Estimates
Cost Estimates					Total Estimated Cost
High: \$	45	Low: \$	6	Cal. Mean	High: \$ 125.75
Mean: \$	25.5	Variance(%)		Cal. Hi, Lo	Low: \$ 29
					Mean: \$ 52
Basis for Cost Estimate					
Joes' guess on 23/08/2006					

You may enter Cost information for each individual requirement in the data entry box.

- **High** and **Low** values, TIGER Pro will calculate the **Mean** value when you click on the **Cal. Mean Button**.
- **Mean** and **Variance**, TIGER Pro will calculate the **High** and **Low** Values you click on the **Cal. Hi, Lo Button**.

You may verify the entry quickly via the [View Data Panel](#).

When ticked, the **Show Totals Check box** enables the display of the **Total Estimated Cost Panel**.

The **Help Button** on the Panel displays this information.

Entry data boxes change color depending on whether the data are for a requirement or a non-requirement.

## 6.3 Key words

Key words allow for searches through the database when assessing the impact of changes.

## 6.4 Priority

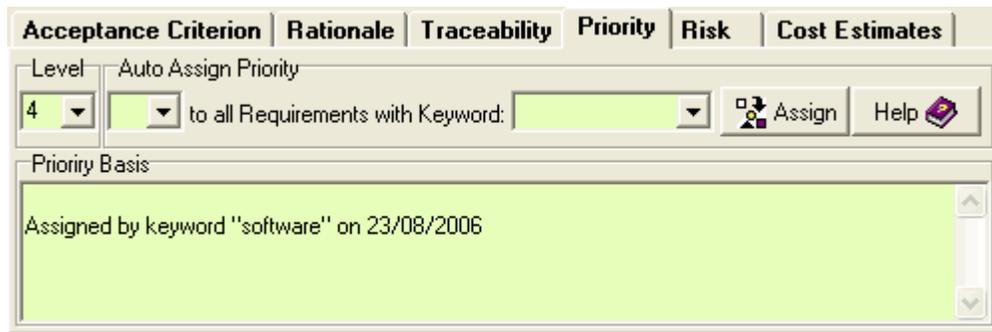
Knowing the priority allows the high priority items to be assigned to early Builds, and simplifies the analysis of the effect of budget cuts.

Priority properties in TIGER Pro are:

- **Level** (1-10) where 10 is the highest probability.
- **Priority Basis** - the documented reason for the assignment of the priority value.

### Entering priority information into TIGER Pro.

Select the Priority Entry Panel by clicking on the **Additional Attributes Content Control Tab** in the [Reports and Graphs Panel](#) and then clicking on the **Priority Tab**. TIGER Pro will then display the Priority Entry Panel shown below.



You may enter **Level** information for each individual requirement in the data entry box.

You may automatically enter **Level** information for sets of requirement selected by keywords. To do so, enter or select a keyword from the drop down panel, then click on the **Assign Button**. TIGER Pro will document this global assignment in the Priority Basis Panel. A typical entry would be:

"Assigned by keyword "software" on "23/08/2006"

You may verify the entry quickly via the [View Data Panel!](#)

The **Help Button** on the Panel displays this information.

Entry data boxes change color depending on whether the data are for a requirement or a non-requirement.

## 6.5 Rationale

The rationale communicates the reason why the requirement was included in the first place. This information is important for considering change requests during the operations and maintenance phase of the system's life cycle. This information is sometimes included as comments in the current paradigm, but is not mandatory.

### Entering Rationale information into TIGER Pro.

Select the **Rationale Panel** by clicking on the **Additional Attributes Content Control Tab** in the [Reports and Graphs Panel](#) and then clicking on the **Rationale Tab**. TIGER Pro will then display the **Rationale Panel**. Enter the information by typing it into the panel.

## 6.6 Risk

### Risk

Any (implementation and programmatic) risk factors associated with the requirement.

Risk properties in TIGER Pro are:

- **Severity** (1-5) where 5 is the most severe.
- **Probability of occurrence** (1-10) where 10 is the highest probability.
- **Risk Basis** - the documented reason for the risk.
- **Risk Mitigation** - the documented mitigation strategy.

The **Risk Level** is a value between 1 and 50 calculated by multiplying the [Severity](#) by the

### Probability of Occurrence.

## Entering risk information into TIGER Pro.

Select the Risk Entry Panel by clicking on the **Additional Attributes Content Control Tab** in the [Reports and Graphs Panel](#) and then clicking on the **Risk Tab**. TIGER Pro will then display the Risk Entry Panel shown below.

You may enter Severity and Probability information for each individual requirement. **TIGER Pro will calculate and display the Risk Level when both Severity and Probability have been entered.**

You may automatically enter Severity and Probability information for sets of requirement selected by keywords. To do so, enter or select a keyword from the drop down panel, then click on the **Assign Button**. TIGER Pro will document this global assignment in the Risk Basis Panel. A typical entry would be:

"Assigned by keyword "hardware" on "23/08/2006"

You may verify the entry quickly via the [View Data Panel](#).

The **Help Button** on the Panel displays this information.

Enter **Risk Basis** and **Risk Mitigation Strategy** information in the respective panels.

Entry data boxes change color depending on whether the data are for a requirement or a non-requirement.

## 6.7 Traceability

There are several kinds of traceability:

- **Traceability to source(s)** – i.e. the concept of operations, regulations, people, etc.
- **Traceability sideways** to other documents (or databases) at the same level of decomposition of the system. This provides information for use by the configuration control board in considering the impact of requested changes.
- **Traceability to implementation** – identifies the Build in which the requirement is scheduled to be implemented.

## Entering Traceability information into TIGER Pro.

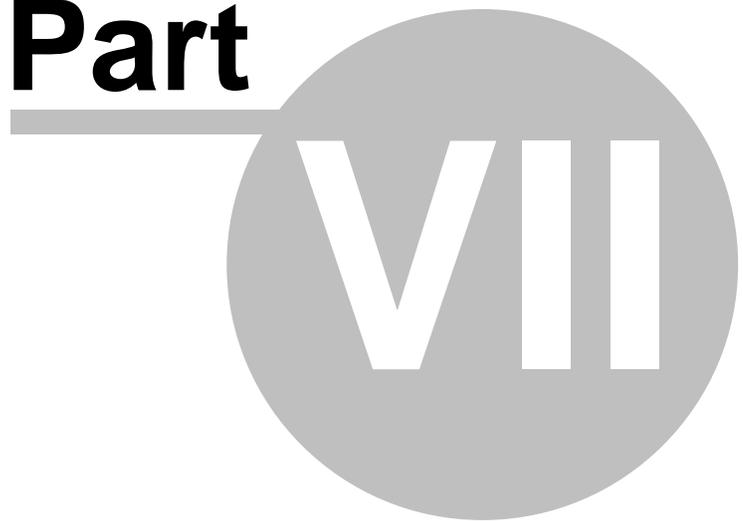
Select the **Traceability Panel** by clicking on the **Additional Attributes Content Control Tab** in the [Reports and Graphs Panel](#) and then clicking on the **Traceability Tab**. TIGER Pro will then display the

**Traceability Panel.** Enter the information by typing it into the panel.

# TIGER Pro

Objective requirements  
engineering

# Part



## 7 Troubleshooting and FAQs

### 7.1 Trouble shooting

#### Errors

The following errors may show up under some circumstances:

- **Key violation Error.** This is an error which occurs when TIGER Pro detects an attempt to write a duplicate requirement ID. To overcome this error, click on **Cancel Edit Button** in the [Select Requirement Panel](#). To terminate the program, use the **File | Exit** options in the **Main Menu**.

#### Known defects

The **Abort Button** is disabled.

The Print functions are disabled.

### 7.2 FAQ

#### Do I have to place my requirements document file into .txt format before ingesting?

"No", You can use use a CSV file.

#### What is the difference between an 'ingest' report and an 'Elucidate' report?

- An 'ingest' report only shows how TIGER Pro treats each line of the requirement specification document, whether it is a requirement or a comment.
- An 'Elucidate' report shows the defects found for each requirement line and gives a summary with a Figure of Merit (FOM).

Comments are not scanned for defects.

#### How do I determine the number of records in a database?

Click on the **Manual Button** in the Update [Summary Panel](#).

#### Can I set my fonts and background colours back to the default?

"Yes", go to the directory where TIGER Pro is stored, locate the file "TIGERPRO.ini" and delete it. Take care as this operation will restore all default values.

#### Can I clear the history list of previously opened files?

"Yes", use the [Tools](#) | **Clear Files name list** option.

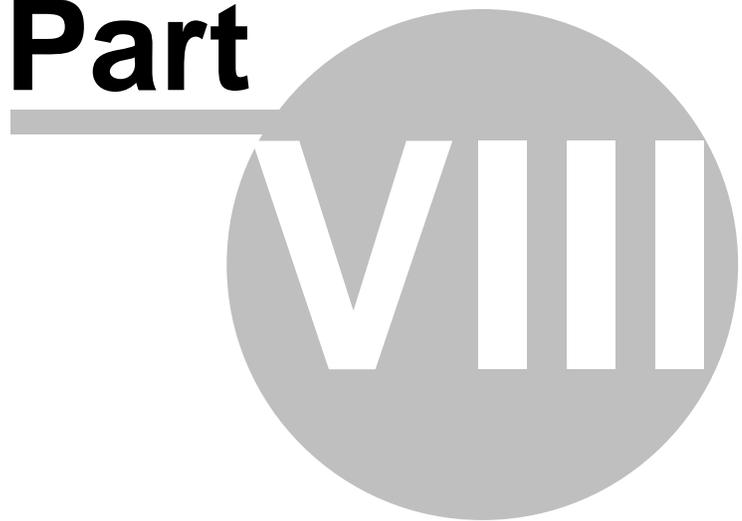
#### Who do I contact about the TIGER Pro tool?

You may contact Dr Joseph Kasser by emailing him at [jkasser@iee.org](mailto:jkasser@iee.org)

# TIGER Pro

Objective requirements  
engineering

# Part



## 8 Version History

This tool is used in classes on systems and software engineering. Each time it is used in a classroom, the workshop gets better and the students or the exercise suggest additional functionality. This had led to an incremental release approach in which the Software is upgraded after each workshop.

### Versions

**TIGER Pro Version 2.0 Release as Shareware September 2006**

**TIGER Pro Version 1.13 Release May 2006**

**TIGER Pro Version 1.1 Release January 2005**

Updated from TIGER 1.21 by combining functionality from the following other no longer supported PETS tools:

- ACE - acceptance criteria assignment management tool.
- ET - rationales, traceability and keywords documentation management tool.
- CARP - priority assignment management tool.
- RAT - risk documentation management tool.
- COW - cost estimation management tool.
- MAX - requirements maximiser tool

# TIGER Pro

Objective requirements  
engineering

# Part

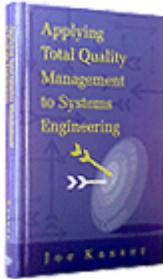


IX

## 9 Credits

### Credits

TIGER Pro was originally developed by **Associate Professor Joseph Kasser** at the Systems Engineering and Evaluation Centre (SEEC) at the University of South Australia. Much of the programming was performed by **Xuan-Linh Tran** who also created the icon. The BADGER concept was suggested by **William Scott** as part of his PhD research in SEEC.



TIGER Pro is based on material published by Dr Kasser in Systems Engineering Symposia and Conference papers between 1994 and 2006 as well as in "*Applying Total Quality Management to Systems Engineering*" published by Artech House.

### Delphi components used.

Some of the functionality in TIGER Pro would not be possible without the use of third-party Delphi components. Acknowledgements are hereby made for the use of the following Freeware Delphi components.

CSV File import and export is made possible by the **PgCSV component** Version 1.31 written by Khashayar Sadjadi ([khashi@pragena.8m.com](mailto:khashi@pragena.8m.com)). Pragena's Delphi Home is (<http://pragena.8m.com>).

The **DBViewCombo** component Version 1.3 by [delphi@korzh.com](mailto:delphi@korzh.com) (<http://www.korzh.com/delphi/>) is used for navigation in dataset. It displays one field value for the current record and contains list of all values of that field in dataset. Selection of a value from the drop down list causes the cursor to move to the corresponding record in dataset. To use it just drop it on the form and set the DataField and DataSource properties.

# TIGER Pro

Objective requirements  
engineering

# Part



---

## 10 Author

Joseph Kasser has been a practising systems engineer for 35 years.

He is the author of "Applying Total Quality Management to Systems Engineering" and many INCOSE symposia papers.

He holds a Doctor of Science in Engineering Management from The George Washington University, and is a Certified Manager.

He is an Associate Research Professor at the Systems Engineering and Evaluation Centre (SEEC) at the University of South Australia.

He performs research into the nature of systems engineering and the properties of object-oriented requirements.

He teaches courses in systems and software engineering in the classroom and via distance mode.

He is a recipient of NASA's Manned Space Flight Awareness Award (Silver Snoopy) for quality and technical excellence for performing and directing systems engineering.

He is a Fellow of the [INCOSE](#). He also serves as Region VI (Asia-Pacific) representative to the INCOSE Member Board and as President of INCOSE-Australia.

# TIGER Pro

Objective requirements  
engineering

**Part**



**XI**

## 11 Software Licence

TIGER Pro is licensed, not sold and provided as-is for the purpose of demonstrating advanced concepts in requirements tools.

For use in educational environments, the instructor's copy should be [registered and the licence fee paid](#). Students may use the [Shareware](#) evaluation version as 'students' for the duration of a course but only in the educational context.

Use in a commercial environment for more than 14 days requires [registration of the software and payment of the licence fee](#). A licensed copy may be installed on any number of computers providing only one copy is in use at any one time.

TIGER Pro is protected by copyright law and international treaty provisions. You acknowledge that:

- No title to the intellectual property in TIGER Pro is transferred to you.
- That the title and full ownership rights to TIGER Pro will remain the exclusive property of Joseph Kasser.
- You will not acquire any rights to TIGER Pro except as expressly set forth in this license.

Joseph Kasser expressly disclaims any warranty for TIGER Pro. TIGER Pro and any related documentation is provided "as is" without warranty of any kind, either express or implied, including, without limitation, the implied warranties or merchantability, fitness for a particular purpose, or non-infringement. The entire risk arising out of use or performance of the software remains with you.

### 11.1 Registering TIGER PRO

There are three versions of TIGER Pro as follows

- **The Shareware version** which displays the splash screen when the program is loaded.
- **The Student version** for use by students in a classroom for a single semester. The instructor should be using a commercially licensed version.
- **[The Commercially licensed version](#)**.

All versions are fully functional.

#### **The Commercially licensed version.**

The individual license fee is \$50.

If you would like to purchase a license for TIGER Pro, you can order on-line at:

<https://secure.element5.com/shareit/checkout.html?productid=300099862&language=English>

Or you can go to <http://www.shareit.com> and enter the program ID number 300099862 in the search window in the screen.

After registering your copy, you will receive a registration code via email to copy and paste into the program via the **Help | Register option** in the **Main Menu**.

For volume discounts and site licences, please email Joseph Kasser at [jkasser@iee.org](mailto:jkasser@iee.org).

## 11.2 Shareware

You are encouraged to copy the program and share it freely with others.

You have the luxury of trying out the product at your own pace and in the comfort of your own home or workplace.

After you have used the material for a [reasonable evaluation period](#), you should either discontinue use of the material or register your copy. Your support is important and greatly appreciated. With it, Shareware authors are encouraged to design and distribute new products. Without it, a great deal of high quality, low cost software will cease to be available. After [registering your copy](#), you will receive a registration code via email to copy and paste into the program via the **Help | Register option** in the **Main Menu**.

Why pay at all?

- You receive support from the author.
- You receive notification of the CURRENT version of the program.
- You receive early notification of upgrades.
- Your input and ideas help shape future products.
- You have a sense of pride and ownership in having honestly participated in the Shareware revolution.
- You help to keep software prices down by supporting a distribution method which doesn't depend on expensive advertising campaigns.

The Splash screen at program initialization is not seen in the registered licensed version of the software.

Be aware of the following restrictions, designed to protect the community of Shareware users and to prevent greedy people from taking unfair advantage of the trust, hard work and good will of Shareware authors.

- No price or consideration may be charged for the material. However, a distribution cost may be charged for the cost of the diskettes, shipping and handling, not to exceed US\$6.
- The files and programs on the disks may not be modified or deleted.
- The material cannot be sold as part of some other more inclusive package.
- The material cannot be "rented" or "leased" to others.
- The end user must be told clearly in writing on the outside of the package and in all advertising that the diskette(s) are "Shareware."
- The package must contain a written explanation that the disk is for evaluation purposes, and that an additional "registration fee" is expected by the author, if the material is used beyond an initial evaluation period.
- In the case of distribution via any telecommunications link, the following must be done:
  - a. An error checking protocol must be used.
  - b. The individual files must be combined into, and transferred in a library or archive format.

# Index

## - A -

about TIGER Pro 34  
acceptance criteria 36  
action 11  
add new poorword 30  
adding a requirement 17  
apply filter 22  
attributes 36  
attributes - keywords 22

## - B -

background color 32  
background colour 32  
Badger 29  
bugs - known 42

## - C -

clarifying poorly written requirements 12  
clearing the file history lists 28  
comment 20  
condition 11  
control display 22  
cost estimates 24, 36  
CSV Control Panel 13

## - D -

database menu options 28  
defects - types of 12  
defects in the software - known 42  
delete - history files 28  
Delphi components used. 46  
display control panel 22

## - E -

edit inhibitor file 30  
edit poor word file 30  
elucidate 25

elucidate control panel 25  
elucidate control panel - visible 24  
Elucidate database 14  
Elucidating files 14  
error - key violation 42

## - F -

FAQs 42  
file 28  
File Control Panel 31  
File Editor 30  
file list 28  
File menu 28  
files - sounds 33  
fix predetermined type 4 defects 29  
fix predetermined type 7 defects 29  
font 32  
font setting 32

## - H -

help 34  
help menu 34  
History 44  
history list 28

## - I -

ignore poor word instruction - view or change 31  
ingest 10, 25  
ingest control panel 25  
ingest control panel - visible 24  
ingest document 10  
ingest merge 10  
ingest requirements 10  
inherit from another database 15  
inherit method 15  
inhibitor file 30  
inhibitors - view 31  
Initiator 11

## - K -

key violation error 42  
keywords 37  
Keywords - assigning 21

keywords - controlling display with 22  
 keywords - entering 21

## - L -

Licence 50  
 License 50  
 load poorword 30

## - M -

Main menu 28  
 Main window 20  
 major functions 10  
 Menu 28  
 Menu - settings 31  
 MP3 files 33

## - N -

new requirement 17  
 non-requirement 20

## - O -

object identifier - view or change 31  
 overview 3

## - P -

parameters 22  
 part of a requirement 11  
 PETS 29  
 poor word file 30  
 poor words 12, 15  
 poor words - overriding TIGER Pro 15  
 poorword list 30  
 print report 33  
 priority 37  
 properties 36

## - R -

rationale 38  
 register 50  
 registering 50

Registering the software 50  
 replace "must" 29  
 replace "should" 29  
 replace "will" 29  
 report 23, 24  
 report menu 33  
 requirement 20  
 requirement - acceptance criteria 36  
 requirement - adding 17  
 requirement - attributes 36  
 requirement - cost estimates 36  
 requirement - format in DOORS exported file 11  
 requirement - format in text file 11  
 requirement - key words 37  
 requirement - new 17  
 requirement - parts of 11  
 requirement - priority 37  
 requirement - properties 36  
 requirement - rationale 38  
 requirement - risks 38  
 requirement - traceability 39  
 requirements - changing ID prefixes 18  
 risks 38

## - S -

save report 33  
 saving time 12  
 select requirement 22  
 set background color 32  
 set background colour 32  
 settings menu 31  
 Shareware 51  
 Short cuts 20  
 sounds 33  
 Sounds - turning on and off 33  
 spreadsheet 24

## - T -

target 11  
 teaching TIGER Pro new Poor words 15  
 test plan 12  
 Test plans - building 4  
 total estimated cost 36  
 tolerance direction 11  
 tolerance level 11

---

tolerance preposition 11  
tolerance units 11  
Tool Bar 20  
tools menu 29  
Top Tool Bar 20  
traceability 39  
trouble shooting 42

## - U -

uses for TIGER Pro 4

## - V -

Version 44  
view data panel 24

## - W -

WAV files 33  
Welcome 3  
William Scott 46

## - X -

Xuan-Linh Tran 46

Endnotes 2... (after index)

Back Cover