

REQUIREMENTS AMBIGUITY CHECKER TOOL - INSTALLATION AND USAGE INSTRUCTIONS

**VERSION 1.0**

A Tool for Requirements Analysis

This template was created to enable departments to more easily develop their project plans. The Department of Technology, Consulting and Planning Division, created this template based on its experiences. The template relies on industry best practices combined with decades of experience on California state information technology projects. The way it was structured is to enable a department to complete the information related to its project without having to write background information related to the discipline. A department may use as much or as little of the template as it wishes.

**Template Instructions:**

* ***Instructions for completing*** this template – written for the author of the project plan - are encased in **[ ]** and the text is ***italicized*** *and* ***bolded.***
* *Examples* are provided as a guideline to the type of sample information presented in each section and the text is *italicized*.
* Boilerplatestandard language for each section is written in the document font and may be used or modified, as necessary.
* A department’s project specific information goes within the brackets ***<< >>***.
* *Informational text is italicized* within square brackets [ ] for informational purposes to the person who has to create the plan and includes background information, explanation, rationale, etc.

DOCUMENT HISTORY

|  |  |  |  |
| --- | --- | --- | --- |
| DOCUMENT REVISION HISTORY | | | |
| Date | Document Version | Revision Description | Author |
| 4/21/2014 | 1.0 | Initial Version | R. Peterson |
|  |  |  |  |
|  |  |  |  |

TABLE OF CONTENTS

[1. What is the Ambiguity Checker Tool? 1](#_Toc403029426)

[2. Installing the Ambiguity Checker Tool 1](#_Toc403029427)

[3. Using the Ambiguity Checker Tool 8](#_Toc403029428)

[4. Interpreting the Results 11](#_Toc403029429)

[APPENDIX A. Standard Dictionary (ambig\_list1.txt) A-1](#_Toc403029430)

What is the Ambiguity Checker Tool?

The Ambiguity Checker Tool is a Microsoft Word utility that uses the following Macros to check a document in .doc or .txt format:

1. Find\_ambiguous\_words – this macro scans the loaded word document looking for words and phrases based on a text file called ambig\_list1.txt. (Going forward, we’ll refer to the ambig\_list1.txt file as the ‘standard dictionary’.)
2. Add\_word\_to\_dictionary – this macro allows you to quickly add new words to ambig\_cust.txt file. (Going forward, we’ll refer to the ambig\_cust.txt file as the ‘custom dictionary’.)
3. Show\_Key – this macro displays the classifications of the highlighted color.

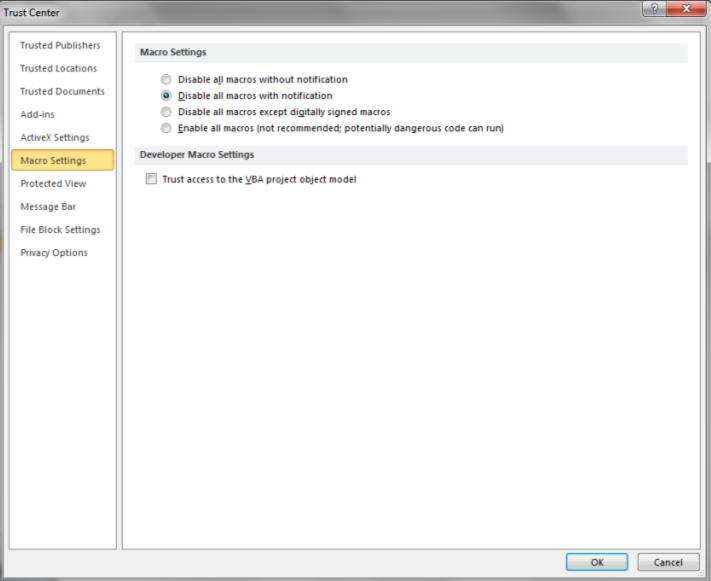
Installing the Ambiguity Checker Tool

The application consists of three (3) files that you need to unzip from “Ambiguity Checker Tool File Ambig.zip” file, which is a separate file that must be downloaded and copied to a local folder, as described in the directions below:

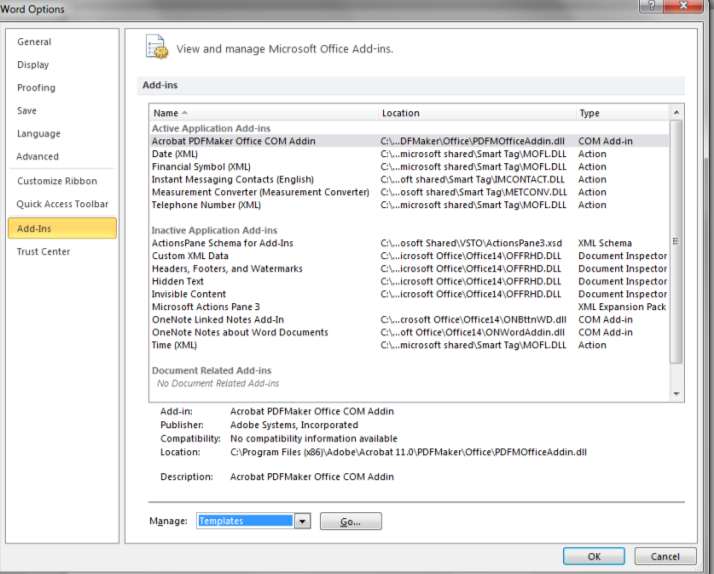
* ambig\_list1.txt
* ambig\_cust.txt
* ambig.dot

**Directions**

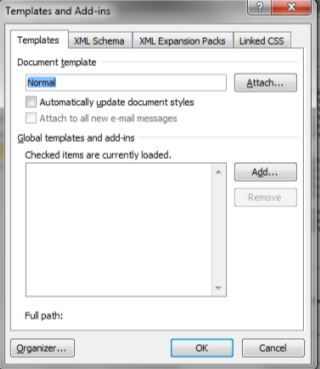
1. Create a folder named “Ambiguity Checker” on your C: drive. (The folder must be named exactly as indicated, “C:\Ambiguity Checker”, else the Tool will not function properly.)
2. Copy the 3 files noted above to the “Ambiguity Checker” folder on your C: drive.
3. Open Microsoft Word.
4. From the File – Options – Trust Center – Trust Center Settings screen, select the Macro Settings option on the left-hand side of the screen, as shown below.



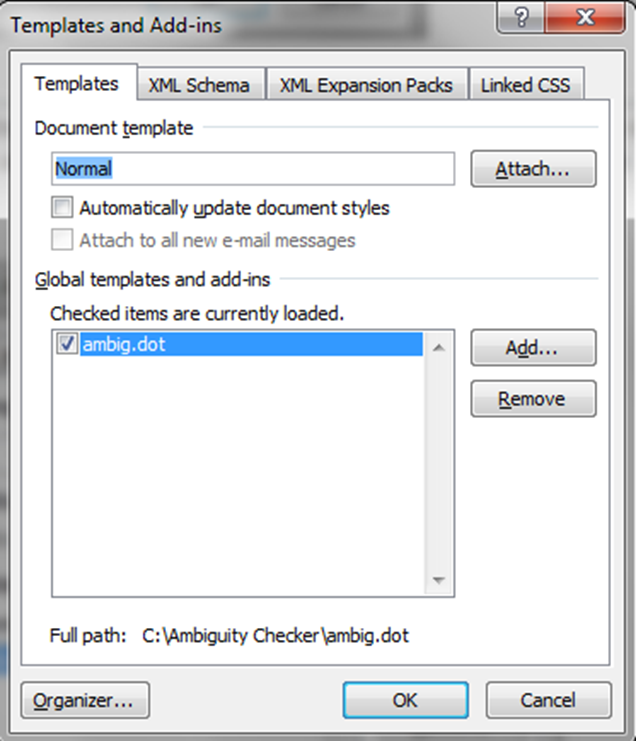
1. Select the “Disable all macros with notification” radio button and then click the OK button. (Since this tool uses macros, Word must be configured to all user selected macros to run.)
2. On the File – Options screen, select the Add-Ins item on the left panel. Then select “Templates” in the Manage dropdown box, as shown below, and then select Go…



1. The Templates and Add-ins screen appears, as show below. From this screen, select Add… and navigate to the Ambiguity Checker folder created above and select the ambig.dot file.

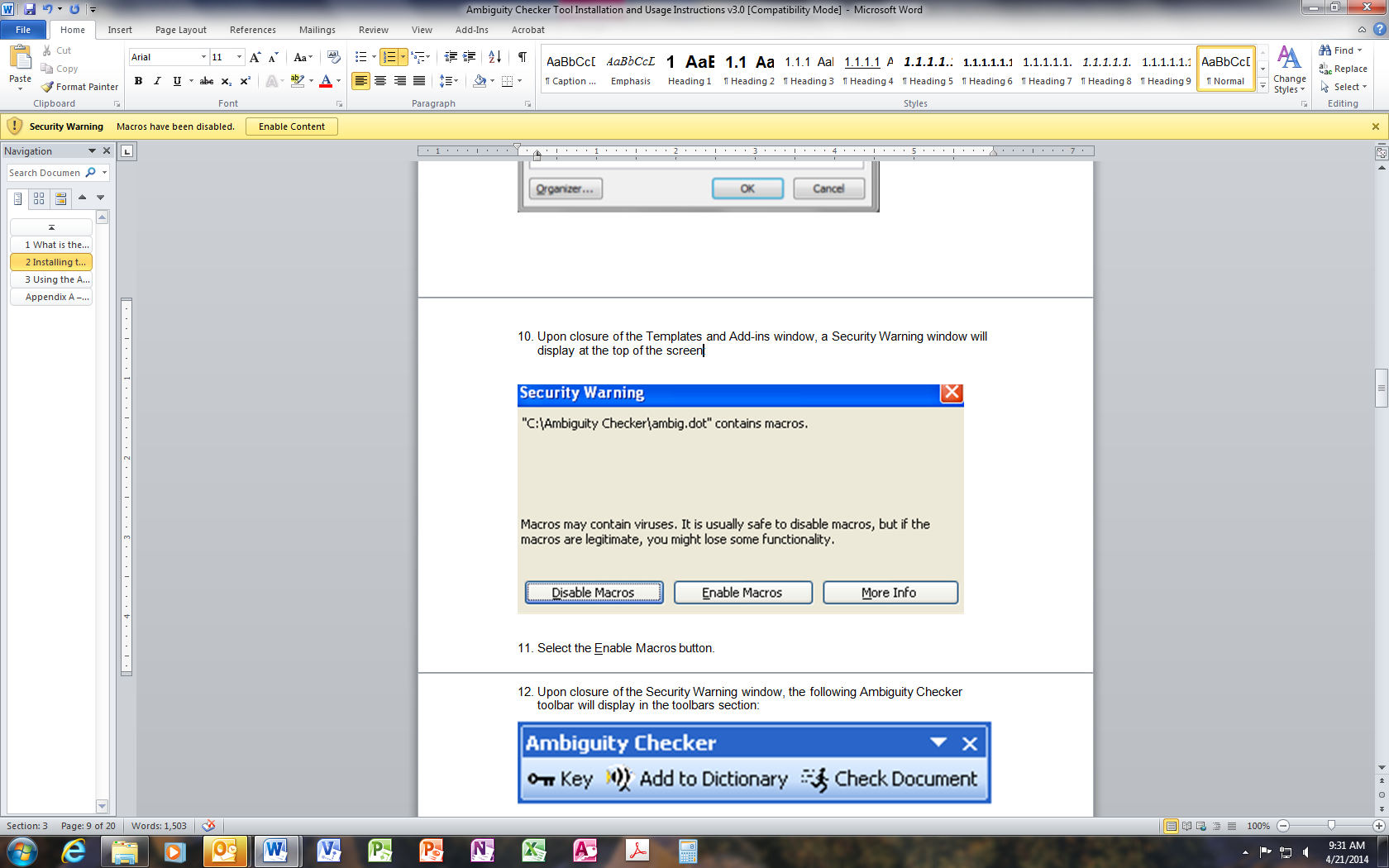


1. Once selected, the template will be added to the list of templates as shown on the Templates and Add-ins screen below. Click OK to add this template.

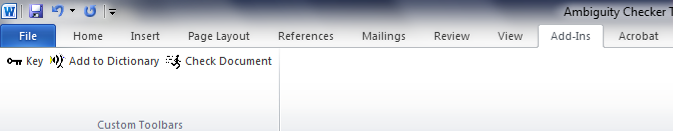


Note: Ensure that the “Document template” located at the top of the page is a valid template that exists on your system. While not part of this tool, by including the ambig.dot add-in, MS Word will also validate that the “Document template” file exists and is valid. If in doubt, click the “Attach…” button and select the MS Word default template called “Normal”.

1. Upon selecting OK on the Templates and Add-ins window, a Security Warning window may be displayed at the top of the screen. Select Enable Contents. Depending on your systems security settings, a Security Warning box may open up to ask “Do you want to make this file a Trusted Document?” If so, select “Yes”.



1. Upon closure of the Security Warning, select the Add-Ins tab at the top, and the following Ambiguity Checker Custom Toolbar will be displayed.



The function of each of these buttons is describe below; do not click these buttons yet as the explanation on how they work is provided later.

The Image of key button returns a window with the classification of each highlight color.

The Image of Add to Dictionary Button button returns a window where the user may enter a new word or phrase, to be added to the custom dictionary (file name “ambig\_cust.txt”).

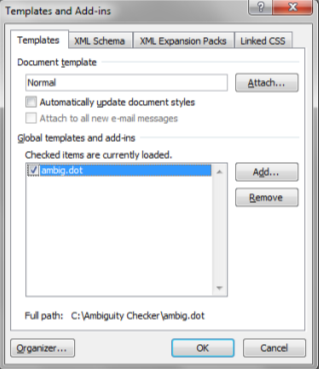
The Image of Check Document Button button scans the document for words or phrases maintained in the standard dictionary and custom dictionary.

1. Close Word, which should not require you to save the document due to the changes made to install this tool; if asked to save the document, it is due to changes you may have made to the document that was open and it is your discretion as to save your document or not.

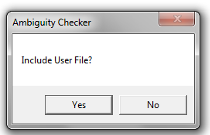
Using the Ambiguity Checker Tool

To use the Ambiguity Checker Tool:

* 1. Open the file you want to scan.
  2. The macros must then be enabled; for security reasons, and by default, the macros are not always turned on. To turn the macros on, go to the File – Options – Add-Ins screen, select “Templates” and then Go…
  3. On the Templates and Add-ins screen, check the ambig.dot checkbox, ensure the Document template file is valid, (e.g., Normal), and select OK. The Security Warning will open at the top of the screen; select Enable Contents. You may also be prompted to make the document a Trusted Document, select OK.



* 1. Select the Add-Ins tab at the top of the page and then select the Image of Check Document Button button in the Ambiguity Checker toolbar.
  2. The following Ambiguity Checker window will open:



* Select Yes to include words or phrases in the custom dictionary, ambig\_cust.txt, in addition to the words or phrases in the standard dictionary, ambig\_list1.txt.
* Select No to only use words or phrases in the standard dictionary.

The Ambiguity Checker Tool will then scan your document and highlight the words and phrases by classification, as follows:

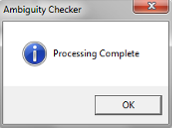
* Pink = IMPERATIVES  
  Consistent use of imperatives is desirable in a requirements document. An imperative is a word or phrase that definitively specifies a “must” in a requirements document. Examples include the use of “shall” to indicate an objective that a business must be able to achieve with an application / system, a task that a user must be able to perform with the application / system, and a behavior or attribute that the application/system must demonstrate.
* Bright Green = DIRECTIVES  
  Consistent use of directives is desirable in a requirements document. A directive is a word, phrase or punctuation mark that points to further related information that elaborates on or further defines the substance of a requirement. Examples include references to Figures or Tables, phrases like “that is”, and a colon as a lead-in to a list.
* Yellow = WEAKPHRASES  
  Weak words and phrases are undesirable in a requirements document and point to the need for further analysis and clarification to resolve any ambiguities.
* Red = INCOMPLETES  
  Incompletes are undesirable in a requirements document and should be resolved before the requirements can be considered ready for approval and acceptable as a basis for design, development, and testing. Examples include TBD or phrases like “to be determined”, “at a minimum”, and “not limited to”.
* Grey = User Defined Weak Phrases

These are words or phrases that are added by the user to the custom dictionary (in file ambig\_cust.txt) via the Ambiguity Checker Tool. At the onset the user can elect for the Ambiguity Checker Tool to include or exclude the words maintained in the custom dictionary, when scanning the document.

The Ambiguity Checker Tool will also generate a summary report in a separate Word document that provides the following information:

* The full path and filename of the scanned document
* The date and time of the scan
* The total number of instances in each of the following categories:
  + Imperatives
  + Directives
  + Weak Phrases
  + Incompletes
  + User Defined Weak Phrases
* The total number of words counted

When the Ambiguity Checker Tool has completed the scan and generated the report, the following dialogue window will display.



1. Click on OK button to close the dialogue window.
2. Save the scanned document under a new filename.
3. Save the report with a filename that associates it with the scanned document.

Special Note:

If you are scanning a large document, you may see the following message or something similar.



Click on Yes and the scan will complete successfully.

Interpreting the Results

By reviewing the Summary Report, an assessment can be made regarding the “tightness” or lack of ambiguity documented within the requirements scanned. In addition, the scanned document can be reviewed to identify the specific instances of words that have been identified and reported by the tool.

For the best results and benefits of using this tool, the scanned document should be reviewed with special attention on the highlighted words and phrases. As defined above, Imperatives (highlighted in pink) are the most desirable and the strongest. Directives (highlighted in green) are also desirable but the requirements that use Directive type words should be reviewed carefully because the requirement will often be incomplete, especially if a list is used in conjunction with the requirement and the list is incomplete. All others, Weak Phrases (highlighted in yellow), Incompletes (highlighted in red), and User Defined Weak Phrases (highlighted in grey) should all be reviewed and modified to make the requirement stronger, which is typically done by eliminating the weak phrase or incomplete word and replacing it with a stronger phrase or word.

The following is an example of an actual Summary Report from the requirements section of an RFP that was released and posted on a procurement website.

-----------------------------------------------------------------------------------------------------------

Ambiguity Report

Document: XXXXXXXXX (Blocked out so as to not identify the Organization/Project)

Review Date: 4/30/2014

Time: 7:25:49 AM

IMPERATIVE: 6086

DIRECTIVE: 12

WEAKPHRASES: 1479

INCOMPLETES: 0

User Defined Weak Phrases: 2

Total Number of Words: 72140

-----------------------------------------------------------------------------------------------------------

In performing a closer look at the individual requirements, the following are samples extracted for requirements identified as using an Imperative and using a Weak Phrase.

Imperative:

The XXXXX solution shall provide system and user error messages that do not require the user to look up error codes or numbers.

For this requirement, the use of “shall” and the use of “and” are both considered imperatives, which makes the requirement a generally stronger requirement.

Weak Phrase:

The XXXXX solution should conform to the W3C Web Content Accessibility Guidelines [http://www.w3.org/TR/WCAG10/].

For this requirement, ‘should’ is defined as a weak phrase, as it is not clear that conformance to the W3C guideline a requirement or not. The requirement does not clearly state that conformance is a requirement and therefore an attribute that the system must have to be accepted by the State. While this may be intentional, it is considered a poor practice and the requirement should be re-worded or re-written to strengthen the requirement.

Often, you will get a single requirement that uses multiple types of words and phrases. The following is an example from the same RFP where both an Imperative and a Weak Phrase is used within the requirement.

The XXXXX solution shall provide the capability to specify time-out values for online applications.

While “shall” is a strong word, an Imperative, the phrase “capability to” is a Weak Phrase. When combined, the overall requirement is considered a weak requirement. For this example, all that is required is that when the system is implemented that the online application time-out values are specified somewhere within the solution and the values can be changed, somehow. Therefore, it would be sufficient and this requirement would be met if the time-out values were set as static values within custom source code. It is hoped that this is not really what the organization wanted.

The bottom line is that this tool only aides the Requirements Team by focusing them on clearly identifiable areas within the requirements that should be reviewed. If the wording is intentional, then no further action is required. However, if not, then an effort must be made to make the requirement stronger. The tool will not find missing requirements or duplicative requirements. It is only used to check the wording of the requirements that have been written.

1. Standard Dictionary (ambig\_list1.txt)

The Ambiguity Checker Tool and the initial set of words in the standard dictionary were developed by various sources, such as International Organization for Standardization, International Electrotechnical Commission, Institute of Electrical and Electronics Engineers, etc. Provided in this section are the initial set of words in the standard dictionary.

Since the Standard Dictionary is a simple text file, words can be added to it or deleted from it as required by the Project. When adding or deleting words, ensure they are placed within the correct category (Imperative, Directive, Weak Phrases, Incompletes) and only use an ASCII Text Editor, such as NotePad; do not leave blank lines or unnecessary white space unless that is part of what you want to search for and report.

When deleting words, care should be used and words should not be deleted just to obtain a better result.

Standard Dictionary Contents

| IMPERATIVE | DIRECTIVE | WEAK PHRASES | INCOMPLETES |
| --- | --- | --- | --- |
| shall | that is | (eg | TBD |
| must | figure | (etc | TBS |
| is required to | table | (ie | TBE |
| are applicable | note | about | TBC |
| are to |  | accordingly | TBR |
| responsible for |  | adequate | not defined |
| will |  | adjust | not determined |
| below |  | after | but not limited to |
| as follows |  | all | as a minimum |
| following |  | almost | provide for |
| listed |  | also | ###### |
| in particular |  | alter |  |
| support |  | although |  |
| and |  | amend |  |
| : |  | any |  |
|  |  | appropriate |  |
|  |  | approximately |  |
|  |  | as a minimum |  |
|  |  | as appropriate |  |
|  |  | as required |  |
|  |  | as the appropriate time |  |
|  |  | as well |  |
|  |  | at a given time |  |
|  |  | at the appropriate time |  |
|  |  | be able to |  |
|  |  | be capable of |  |
|  |  | besides |  |
|  |  | but |  |
|  |  | but not limited to |  |
|  |  | by and large |  |
|  |  | calculate |  |
|  |  | can be |  |
|  |  | capability of |  |
|  |  | capability to |  |
|  |  | certain |  |
|  |  | change |  |
|  |  | commonly |  |
|  |  | compare |  |
|  |  | compatible |  |
|  |  | complete |  |
|  |  | compute |  |
|  |  | conventional |  |
|  |  | could |  |
|  |  | could be |  |
|  |  | custom |  |
|  |  | customarily |  |
|  |  | customize |  |
|  |  | daily |  |
|  |  | data |  |
|  |  | derive |  |
|  |  | determine |  |
|  |  | e.g. |  |
|  |  | easy |  |
|  |  | easy to |  |
|  |  | effective |  |
|  |  | efficient |  |
|  |  | efficiently |  |
|  |  | eliminate |  |
|  |  | enable |  |
|  |  | etc. |  |
|  |  | even though |  |
|  |  | every |  |
|  |  | fast |  |
|  |  | few |  |
|  |  | for example |  |
|  |  | frequent |  |
|  |  | frequently |  |
|  |  | furthermore |  |
|  |  | generally |  |
|  |  | handle |  |
|  |  | hardly ever |  |
|  |  | i.e. |  |
|  |  | improve |  |
|  |  | improved |  |
|  |  | in a while |  |
|  |  | in addition to |  |
|  |  | indicate |  |
|  |  | inexpensive |  |
|  |  | infrequent |  |
|  |  | infrequently |  |
|  |  | intuitive |  |
|  |  | is one of |  |
|  |  | it |  |
|  |  | just about |  |
|  |  | later |  |
|  |  | likewise |  |
|  |  | limited |  |
|  |  | major |  |
|  |  | manage |  |
|  |  | manipulate |  |
|  |  | many |  |
|  |  | match |  |
|  |  | maximize |  |
|  |  | may |  |
|  |  | may be |  |
|  |  | might |  |
|  |  | minimal |  |
|  |  | minimize |  |
|  |  | modify |  |
|  |  | monthly |  |
|  |  | more often than not |  |
|  |  | more or less |  |
|  |  | moreover |  |
|  |  | most |  |
|  |  | mostly |  |
|  |  | must be |  |
|  |  | nearly |  |
|  |  | normal |  |
|  |  | normally |  |
|  |  | not defined |  |
|  |  | not determined |  |
|  |  | not quite |  |
|  |  | notwithstanding |  |
|  |  | occasionally |  |
|  |  | often |  |
|  |  | on the odd occasion |  |
|  |  | on the other hand |  |
|  |  | optimize |  |
|  |  | optionally |  |
|  |  | ordinarily |  |
|  |  | ordinary |  |
|  |  | perform |  |
|  |  | process |  |
|  |  | produce |  |
|  |  | properly |  |
|  |  | provide for |  |
|  |  | quickly |  |
|  |  | rapid |  |
|  |  | rare |  |
|  |  | rarely |  |
|  |  | regularly |  |
|  |  | reject |  |
|  |  | reliable |  |
|  |  | responsive |  |
|  |  | robust |  |
|  |  | roughly |  |
|  |  | rule |  |
|  |  | same |  |
|  |  | scalable |  |
|  |  | seamless |  |
|  |  | seamlessly |  |
|  |  | seldom |  |
|  |  | series |  |
|  |  | set |  |
|  |  | several |  |
|  |  | should |  |
|  |  | should be |  |
|  |  | similar |  |
|  |  | simple |  |
|  |  | slowly |  |
|  |  | small |  |
|  |  | some |  |
|  |  | sometime |  |
|  |  | somewhat |  |
|  |  | soon |  |
|  |  | specific |  |
|  |  | specified |  |
|  |  | standard |  |
|  |  | state-of-the-art |  |
|  |  | status |  |
|  |  | still |  |
|  |  | substantial |  |
|  |  | such |  |
|  |  | support |  |
|  |  | the above |  |
|  |  | the complete |  |
|  |  | the entire |  |
|  |  | the previous |  |
|  |  | these |  |
|  |  | this |  |
|  |  | though |  |
|  |  | timely |  |
|  |  | too |  |
|  |  | typical |  |
|  |  | typically |  |
|  |  | up to |  |
|  |  | usual |  |
|  |  | value |  |
|  |  | verify |  |
|  |  | virtually |  |
|  |  | weekly |  |
|  |  | whereas |  |
|  |  | will be |  |
|  |  | yearly |  |
|  |  | yet |  |