Document analysis system and method

Field of the invention

This invention relates to a system for analysing, processing and navigating through documents, and in particular documents of a structured nature for example legal and semi-legal documents such as contracts, warranties, policies, legislation etc.

Summary of the invention

In a first aspect, the invention resides in a document analysis system including a set of element classes, each element class having an associated set of document elements and an associated set of format rules, and identifying means for identifying one or more document elements within an original document, wherein the system creates a representation of the original document formatted in accordance with the format rules for the element class of the document elements identified in the original document.

In a second aspect, the invention resides in a document analysis method including the steps of:

- identifying one or more document elements within an original document,

- classifying identified document elements into one or more of a plurality of element classes and

- creating a representation of the original document formatted in accordance with a set of format rules associated with the element classes to which the identified document elements belong.

In a third aspect, the invention resides in a document analysis system including a set of element classes, each element class having an associated set of document elements and an associated set of document dividing rules, identifying means for identifying one or more document elements within an original document, said identified document elements dividing the document into a plurality of sections according to the dividing rules for the element classes of the identified document elements, said document sections being linked by the document elements and storing means for storing the document sections in a database.

In a fourth aspect, the invention resides in a document analysis method including the steps of :

- identifying one or more document elements within an original document,
- classifying identified elements into one or more of a plurality of element classes,
- dividing the document into a plurality of document sections linked by the document elements in accordance with a set of rules associated with the element classes to which the identified document elements belong, and
- storing the document sections in a database.

In a fifth aspect, the invention resides in a document mapping system including a set of element classes, each element class having an associated set of document elements and an associated set of format and mapping rules, identifying means for identifying one or more document elements within an original document, dividing means for dividing the document into a plurality of document sections in accordance with the set of rules for the element class of the identified document elements and mapping means for creating a map of document sections linked by labels representing the respective documents elements pertaining to those document sections.

In a sixth aspect, the invention resides in a document analysis method including the steps of:

- identifying one or more document elements within an original document,
- classifying identified elements into one or more element classes,
- dividing the document into a plurality of document sections in accordance with a set of rules associated with the element class of the identified document elements, and
- representing the original document as a map of document sections linked by labels representing the respective document elements pertaining to those document sections.

Preferably the document elements are link element linking sections of document subject matter. More preferably, the document elements are grammatical construct elements, that is words or phrases, linking sections of textual subject matter of the original document. The element classes may further include punctuation that assists in identifying the grammatical construct elements for that class.

Preferably, the format and mapping rules for one or more element classes include graphic elements for displaying the document sections pertaining to the respective document element.

In one embodiment, the system creates an amended version of the original document containing highlights in accordance with the format rules for the identified document elements.

Preferably, the system and method of the invention are embodied in computer software.

Brief description of the drawings

The invention will now be described, by way of non-limiting example only, with reference to preferred embodiments and to the accompanying drawings in which :

Figure 1 is a schematic of a database used in the present invention;

Figure 2 shows a schematic sample of a document being converted to a document map;

Figure 3 is a sample screen shot of a mapping software application interface;

Figure 4 is a schematic example of a record structure;

Figure 5 shows a document map after an initial step in a validation process;

Figure 6 shows the document map of Figure 5 after full compliance;

Figure 7 shows the affect of a non-complying requirement in the document map of Figure 5;

Figure 8 shows a document map including evidence fields;

Figure 9 shows how a document map can be used to plan compliance;

Figure 10 shows a preferred method for mapping embedded requirements; and

Figure 11 shows a document map produced for a Chinese document.

Detailed description of the embodiments

Many structured or formatted documents contain lengthy and complex clauses that make navigating through the document difficult. The system of the present invention recognises that while such documents are unique and contain sections drafted for the express purpose of the particular document, these sections are typically linked using specific and generic grammatical constructs that are common to many documents.

The system of the present invention as depicted in Figure 1 includes a database 10 containing a set of element classes 12, 13, 14, 15, 16, 17. Each element class has a class heading and a plurality of document elements. Each element class also has a linked set of document formatting and mapping rules 12a, 13a, 14a, 15a, 16a, 17a.

The element class heading represents a grammatical function type found in a document. The document elements for an element class include the standard ways in which that function type may be grammatically expressed within a document. That is, the class represents a logical operator and the document elements are the text strings used to express the logical operator. Typically the document elements are grammatical elements used to link "requirements" eg of a clause in a contract. For example a clause may specify :

"Statement A AND Statement B",

where "Statement A" represents a first requirement, "Statement B" represents a second requirement. The "AND" element is the grammatical element used to link the first and second requirements. The "AND" element also identifies the rule for satisfying the entire clause, ie to satisfy an "AND" clause, all requirements (Statement A, Statement B in the present example) must be satisfied in order for the clause to be deemed complied with.

In the present example eleven element classes or operator classes are shown, with the headings "Multiple Requirements", "Choice of Requirements", "Leading Conditional Requirements", "Trailing Conditional Requirements", "Comparative Preferential Requirements", "Requirements with Leading Exceptions", "Requirements with Trailing Exceptions", "Progressional Preferential Requirements", "Mandatory Requirements", "Optional Requirements" and "Sequential Requirements".

By way of example, the "Multiple Requirements" class includes as document elements within the class, the words and phrases or text strings "AND" "&", "with", "together with", "as well as", "also", "adding", " in addition to", "includes", and "covering all things in". The term "AND" is the lead document element and in the Figure is depicted in capitals.

An exemplary list of document elements for each of the element classes is shown in Tables 1a to 1f below. The tables referred to also show exemplary format, mapping and compliance rules, including colour schemes, for the element classes.

	Multiple requirements	Choice of requirements

Compliance rules	Each of the unique requirements must be satisfied for compliance.	The choice must be satisfied for compliance.
Example requirement	a and b	a or b
Colour coding	Green	Red
Summary word	and	or
Typical words and phrases	• & • along with • also • and • as well as • covering all things in • furthermore • in addition to • includes • including • moreover • plus • together with • with	• alternative • any two of the following • choice between • either • one of the following • or • or else • one or more of the following • otherwise • two or more of the following
Mark up symbols	Semi-circle A and b. A and b. A and b. A and b. A and b.	Circle A or b. A or b. A or b. A or b. A or b.

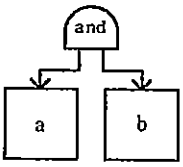
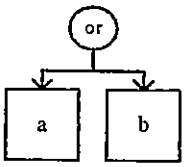
Document mapping layouts		

Table 1a – Multiple and Choice element classes

	Trailing conditional requirements	**Leading conditional** requirements
Compliance rules	A is to be made as to whether the condition is complied with, and whether the consequence needs to be complied with. If the consequence is to be complied with, then it must b.	
Example requirement	a **if** b	**if** b, **then** a
Colour coding	**Blue**	
Summary word	**if**	**If then**
Typical words and phrases	**Trailing condition** • after • as soon as • before • but • for • granting that • if • in the case of • in the event of • on • on condition that • once • provided • proviso • subject to • unless	**Leading Condition**

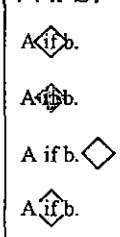
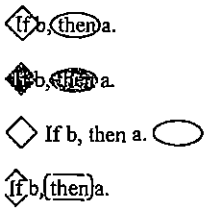
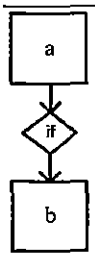
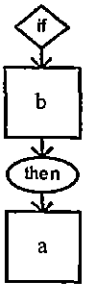
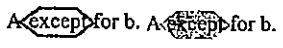
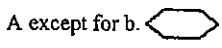
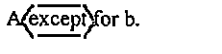
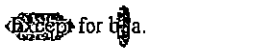
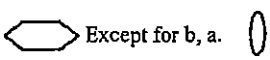
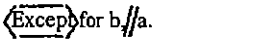
	• upon • when • where required by • without	
Dividers		Consequence • then • , (comma)
Mark up symbols	Diamond A **if** b. 	Diamond & oval **If** b **then** a. 
Document Mapping layouts		

Table 1b – Conditional element classes

	Requirements **with trailing exceptions**	Requirements **with leading exceptions**
Compliance rules	Requirement must be complied with, except when certain requirements are complied with.	
Example requirement	a **except** for b	**Except** for b, a
Colour coding	Orange	
Summary word	except	
Typical	**Trailing exception**	**Leading exception**

words and phrases	• apart • aside from • bar • barring • besides • but, but not • despite • except for • excepting • exclude • excluding • exclusive of • exempt • from • not including • omitting • other than • unless • with the exception of	
Dividers		Usual requirement , (comma)
Mark up symbols	Hexagon A except for b.   	Hexagon & oval Except for b, a.    

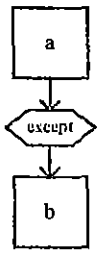
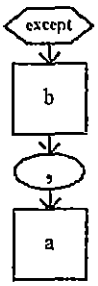
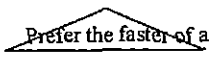
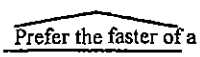
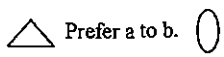
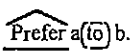
Document mapping layouts		

Table 1c – Exception classes

	Comparative preferential requirements	**Progressional preferential** requirements
Compliance rules	A check is to be made as to which requirement is preferred, and then that requirement is to be complied with.	A check is to be made as to which requirement is preferred first, and then that requirement is to be complied with before successive preferences.
Example requirement	**prefer** the **faster** of a and b	**prefer** a **to** b
Colour coding	**Pink**	
Summary word	**prefer**	**prefer, to**
Typical words and phrases	**Comparative preference** • **desire** • **descending, ascending** • **fancy** • **first, second, third, etc** • **greater, shorter, longer, etc** • **greatest,**	**Progressional preference**

	shortest, longest, etc • last • middle, median, average • prefer • preference • preferential • priority • rather • runner up • the most, least • wish •	
Dividers		Successive preference • han • to
Mark up symbols	Triangle Prefer the faster of a and b.   Prefer the faster of a and b. 	Triangle & oval Prefer a to b.    

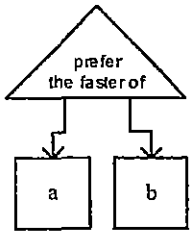
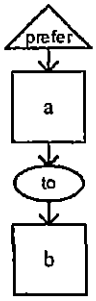
Document mapping layouts		

Table 1d – Preference element classes

	Mandatory requirements	**Optional requirements**
Compliance rules	Compliance with the requirement is mandatory.	Compliance with the requirement is optional.
Example requirement	Xxx shall a and b.	Xxx may a and b.
Colour coding	**Black**	**Black**
Summary word	**shall**	**should**
Typical words and phrases	• as required by • comply with • has had • have to • in accordance with • is • must • normative • shall, shall not • will, will not • may not	• can • could • informative • may • might • ought to • perhaps • should • suitable form of
Mark up symbols	**Solid underline**	**Broken underline**

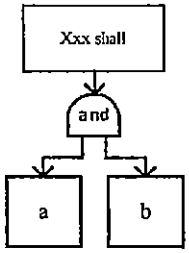
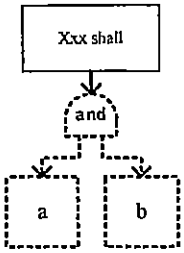
	Xxx shall a **and** b.	Xxx may a **and** b.
Document mapping layouts		

Table 1e – Mandatory and optional classes

	Structure of requirements
	Sequential requirements
Compliance rules	Each of the unique requirements must be satisfied for compliance
Example requirement	a then b
Colour coding	grey
Summary word	then
Typical words and phrases	• chain • cycle • first, firstly • in the following order • progression • run • second, secondly • sequence • series • string • succession • train
Dividers	Successive requirements

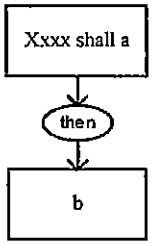
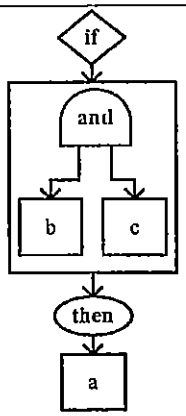
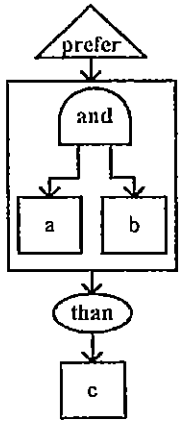
	• then • ,(comma)
Mark up symbols	Oval Xxxx shall a, **then** b. Xxxx shall a (then) b. Xxxx shall a, (then) b. Xxxx shall a, then b. ◯ Xxxx shall a [then] b.
Document mapping layouts	

Table 1f – Sequential element class

The embedded requirements are typically inserted into the requirement box when there is a following requirement. For example

5

Multiple requirements embedded within the **condition** of a **leading conditional** requirement. **If** [b **and** c], **then** a.	
Multiple requirements embedded in the primary **preference** of a **progressional preference.** **Prefer** [a **and** b] **to** c.	

An example of analysing a document in accordance with a preferred aspect of the invention will now be described. As a first step, an original document is reviewed to identify any words and/or phrases (and associated punctuation) that match members of the element classes. The document elements identified form dividers between text blocks, typically containing the requirements for those document elements, that allow the document to be divided into sections. In terms of producing a document map, the actual text or expression of a requirement is unimportant.

A map of the document can then be constructed using the mapping rules for the element classes to which the identified document elements belong. A simplified example of the process is illustrated in Figure 2 where the individual requirements within a contract 40 have been

mapped as depicted in the document map 50. In producing the map 50, the system has identified the document elements "or" 41, "and" 42 and "if"/"then" 43 and produced the map 50 in accordance with the format and mapping rules listed in Tables 1a-1c for the "Choice Requirements", "Multiple Requirements" and "Leading Conditional Requirements" classes respectively.

In constructing the map 50 the "and" element 42 is identified, as are the three requirements headed "a)", "b)" and "c)". An "AND" node 51 is created, being the symbol for elements in the "Multiple" class, on the map with three branches extending from beneath it to receive the respective requirements "a)", "b)" and "c)". Within sub-clause "b)", the element "or" 41 is identified between two requirements. The "or" element is classified in the "Choice of" class and is depicted on the map as an "OR" node 52, being the map symbol used for "Choice of" elements. The "OR" node 52 is displayed on the second branch from the "AND" node 51 with the two identified requirements being displayed in respective boxes branching from the "OR" node 52.

Within sub-clause "c)" the "if" and "then" elements 43a, 43b are identified with the leading conditional requirement 44 between the "if" and "then" elements and the consequential requirement 45 following. The map 50 depicts the requirement of sub-clause "c)" text box 53 on the third branch extending from the "AND" node 51 . The requirement of sub-clause "c" is connected to the "Leading Condition" symbol 54. The text of the conditional requirement 44 is connected to the text box 55. The "Consequential condition" symbol 56 is connected to the "Leading Condition" text box 55. The "Consequential condition" symbol 56 is then connected to the 'Consequential condition" text box 57. The text of the consequential requirement is placed in a box 57 branching from the "THEN" node.

The output of the document analysis system is thus a formatted document that allows the document to be more clearly navigated. In one embodiment, the formatted document is a map of the type illustrated in Figure 2. In an alternative embodiment, the formatted document may be in substantially the same form as the original document but with the document elements highlighted with the appropriate colours, fonts, etc in accordance with the format rules. An example of such a system is where a contract clause as in Table 2 is converted into a formatted document as shown in Table 3 below.

4.2 METHOD OF QUALIFICATION OF WELDING PROCEDURE

A welding procedure shall be qualified by one of the following methods:

a) A prequalified procedure in accordance with Clause 4.3.

b) Production of documentary evidence of relevant prior experience by the fabricator.

c) Production of a suitable length of test piece of the same joint type, material type, material thickness and edge preparation as the requirement upon which the procedures are to be applied and testing it in accordance with Clause 4.6 where the type of joint allows such testing. The test piece may be made as a production test piece.

d) Preparation of a special test piece, such as shown in Figure 4.6.3, which simulates as closely as practicable the weld penetration, material type and direction of rolling, material thickness, edge preparation, welding conditions including welder access and conditions of restraint to be used in production, and testing it in accordance with Clause 4.6.

e) Destructive testing of a prototype joint, structure or component.

4.3 PREQUALIFIED WELDING PROCEDURES

Welding procedures shall be deemed to be prequalified where—

a) the joint preparations are prequalified in accordance with Clause 4.4

b) the consumables are prequalified in accordance with Clause 4.5;

c) the workmanship and welding techniques, including the preheat and interrun temperature requirements comply with this Standard; and

d) where required by Clause 4.6.2, //documentary evidence is available of a satisfactory macro test in accordance with Clause 4.6.5.

4.4 .....

4.5 QUALIFICATION OF WELDING CONSUMABLES

4.5.1 Prequalified welding consumables

.....

4.5.2 Qualification of welding consumables by testing

Table 2

4.2 METHOD OF QUALIFICATION OF WELDING PROCEDURE

A welding procedure shall be qualified by **one of the following** methods:

a) A prequalified procedure in accordance with Clause 4.3.

b) Production of documentary evidence of relevant prior experience by the fabricator.

c) Production of a suitable length of test piece of the same joint type, material type, material thickness and edge preparation as the requirement upon which the procedures are to be applied and testing it in accordance with Clause 4.6 where the type of joint allows such testing. The test piece may be made as a production test piece.

d) Preparation of a special test piece, such as shown in Figure 4.6.3, which simulates as closely as practicable the weld penetration, material type and direction of rolling, material thickness, edge preparation, welding conditions including welder access and conditions of restraint to be used in production, and testing it in accordance with Clause 4.6.

e) Destructive testing of a prototype joint, structure or component.

4.3 PREQUALIFIED WELDING PROCEDURES

Welding procedures shall be deemed to be prequalified **where**—

a) the joint preparations are prequalified in accordance with Clause 4.4

b) the consumables are prequalified in accordance with Clause 4.5;

c) the workmanship and welding techniques, including the preheat and interrun temperature requirements comply with this Standard; **and**

d) **where** required by Clause 4.6.2, //documentary evidence is available of a satisfactory macro test in accordance with Clause 4.6.5.

Table 3

Whether or not a word within a document is identified as a document element can depend upon its context, that is, on any surrounding words, phrases or punctuation. For example, not every occurrence of the word "AND" within a document will necessitate a node to be created for it. It may only be required in cases of the following form:

Condition Statement

a) First requirement

b) Second requirement; and

c) Third requirement.

Thus the identification of an occurance of the word "AND" as a document element may be conditional on it being immediately proceeded by a semicolon ";" or like divider.

A further and more complex example of a conditional classification of an identified element into an element class occurs with the terms "two or more", "three or more" and the like. Such terms will form a subset of the "Choice of" class as some but not all of the requirements pertaining to the element will need to be met for the entire requirement to be complied with. Thus, after identifying the presence of an "or" within a document, it is necessary to analyse the surrounding words and/or punctuation to determine the context of the element to ascertain whether the element falls within a subset of the broader element class.

Punctuation in the original document will play an important role in determining the relationship between text sections and the document elements. An example is the contract condition "(a) and (b) or (c)" which, as shown in Figures. 10a and 10b, can be represented in two different ways depending on how the statement is punctuated. Punctuation, headings and numbered sections e.g. a,b,c...; i, ii, iii...; can be used to correctly divide the document into text sections.

In a preferred embodiment of the invention, the system is a software application executed on a computer processor that accesses an electronic database storing the element class information. The system includes application programs and an electronic database and/or electronic libraries storing the element classes and rules, as shown and described with respect to Figure 1. The software application is adapted to run on a computer terminal such as a personal computer or work station.

Initially, an original document to be analysed and mapped is obtained in an electronic format such as in Microsoft Word™ format. A highlighted version can be obtained by identifying within the original document the document elements as they appear in the document and converting those elements according to the highlighting rules for the element class. Document elements within the document are identified using any suitable known search algorithm, of which many are known to the skilled addressee. Similarly, many algorithms for highlighting the identified document elements in accordance with the appropriate formatting rules would be known to the skilled addressee. The following is a simple search and formatting algorithm for locating the document element "and" and displaying the element in green, as specified by the rule for "Multiple Requirements" class :

```
Sub CTB_Contract_HIGHLIGHTS()
'
' CTB_Contract HIGHLIGHTS Macro
' Macro recorded 06/11/00 by Nathan McDonald
'MULITPLE REQUIREMENTS (GREEN)
    Selection.Find.ClearFormatting
    Selection.Find.Replacement.ClearFormatting
    Selection.Find.Replacement.Font.ColorIndex = wdGreen
    With Selection.Find
        .Text = "and"
        .Replacement.Text = "and"
        .Forward = True
        .Wrap = wdFindContinue
        .Format = True
        .MatchCase = False
        .MatchWholeWord = True
        .MatchWildcards = False
        .MatchSoundsLike = False
        .MatchAllWordForms = False
    End With
    Selection.Find.Execute Replace:=wdReplaceAll
```

Similar algorithms can be used to identify and highlight the other document elements. The process is repeated for each document element stored in the database. The output of the search and format procedure is a formatted version of the original document containing the appropriate highlights. The formatted document is displayed on the graphic user interface (GUI) of the computer terminal.

The display of the formatted document includes a user selectable macro tool that allows the user to override the automatic formatting where a document element has been inappropriately identified.

A further macro allows a user to manually highlight document elements and the associated requirements that use a language outside of the stored document elements. The user selects the text in the document and then selects the macro tool for a document element class. The macro tool then applies the highlighting rules for that class to the selected text.

The formatted document is then used as the basis for creating a document map through a mapping interface. A sample interface of a mapping application is shown in Figure 3. One preferred mapping application is based on the Microsoft Visio™. The mapping application interface 30 includes a menu 31 and a mapping area 32. The menu displays graphical symbols 33 for each of the element classes. Each symbol represents an element class and includes a class header 35 and linked requirement boxes 36.

The interface of the computer terminal on which the application runs can be manipulated to display the formatted document and mapping applications simultaneously on the GUI.

The user creates the document map clause by clause according to the formatted document. To commence, a user identifies a clause in the formatted document that is to be mapped. The clause will contain a document element and one or more requirements. In the mapping application, the user selects from the menu an appropriate element class symbol matching the document element of the clause and drags and drops the symbol into the mapping area. The user then highlights and selects a clause requirement from the formatted document and places the text of the requirement in a text box of the symbol in the mapping area. As stated previously, in terms of producing a document map, the actual text or expression of a requirement is unimportant and thus the requirement can be manipulated as a single text block.

Further requirements are placed in additional text blocks of the symbol.

The text of a symbol may be displayed within the symbol in its entirety or alternatively, a summary label may be produced and displayed as hyperlink with the full text of the requirement being displayed when the text box is selected by the user.

A class symbol will have a default number of requirements. For example, "and" and "or" clauses will each have a minimum of two requirements and thus the symbol displays a default number of two text boxes. Further requirement boxes can be added as necessary, for example by selecting an option from a drop down menu available for that element class symbol. Alternatively, when the user selects a symbol from the menu, the user may be prompted to supply the number of requirement boxes to be included in the symbol.

To continue building the map, the user selects a further clause of the formatted document which may be an individual clause or an embedded clause, and drags the appropriate symbol into the mapping area. The user creates a graphical link between the first and second clauses by dragging the screen cursor between the two symbols to draw the link. The process of filling the requirement boxes is then repeated accordingly.

When a new symbol is added to the map, an element record is created. Referring to Figure 4, an element record 450will have a field for the element class 451 and fields 452 for linking to the requirements related to that element. The element record will further include a validation field 453 for each of the linked requirements and a complete validation field. The complete validation field is governed by a rule for the element class as will be described in greater detail below.

When a requirement is added, a requirement record is created. The element record becomes a parent record to each of the subordinate requirement records. An exemplary requirement record structure is shown in Figure 4. Each requirement record 401 will have a text field 402 that stores the text of the requirement. Further record fields include a clause identifier 403, clause title 404, summary or label field 405, compliance box 406 and evidence field 407. The record will also contain addressing fields for linking to the parent element record and subsequent element or requirement records, and embedded requirements.

The map is thus stored electronically as a series of linked requirement records and element records.

Contracts often have requirements embedded within others. A set of brackets may be used to highlight the embedded requirements in the highlighted document. With reference to

Figure 10, when creating the document map, embedded requirements are to be either inserted into the unique requirement box or connected directly to the embedded structural symbol.

The following example shows a choice of requirement embedded in a multiple requirement. Figure 10a shows embedding into the requirement box, and the second shows direct linking to the embedded structural symbol.

The software application also supports an automatic navigation or validation process. Whilst the compliance rules would ordinarily be specified and stored with the element class rules, the compliance rules are detailed in Table 4 below.

Structure of requirements

Multiple requirements	Each of the unique requirements must be satisfied for compliance.	a **and** b
Choice of requirements	The choice must be satisfied for compliance.	a **or** b
Requirements with **Leading conditions**	A is to be made as to whether the condition is complied with, and whether the consequence needs to be complied with. If the consequence is to be complied with, then it must b.	**if** b, **then** a
Requirements with **Trailing conditions**		a **if** b
Requirements with **leading exceptions**	A requirement must be complied with, except when certain requirements are complied with.	**Except** for b, a
Requirements with **trailing exceptions**		a **except** for b
Comparative preferences	A check is to be made as to which requirement is preferred, and then that requirement is to be complied with.	**prefer** the **faster** of a and b
Progressional preferences	A check is to be made as to which requirement is preferred first, and then that requirement is to be complied with before successive preferences.	**prefer** a **to** b

Sequential requirements	Each of the unique requirements must be satisfied for compliance.	**a then b**

Intensity of requirements

Mandatory requirements	Compliance with the requirement is mandatory.	shall a
Optional requirements	Compliance with the requirement is optional.	should a

Table 4

Using the validation process, a user can navigate through a document by validating that particular requirements of the document have been met. A user validates a requirement by selecting an appropriate icon on the interface pertaining to that requirement. The software then updates the compliance field of that requirement record with a compliance status, eg TRUE, FALSE etc. The compliance field of the parent element record relating to that requirement is also updated. The system further prompts the user to provide compliance evidence, eg a receipt number, link to a document or other evidence that the requirement is met. In this way, the user's own record keeping is enhanced. Validation of a requirement can be indicated on the user interface, eg by shading the requirement.

In an alternative embodiment, a non-blank compliance evidence field is deemed to indicate that evidence of compliance with the requirement exists and accordingly, a non-blank validation field triggers the updating of the compliance field to a validated status.

Validation of an entire clause can be indicated by shading the entire clause. The element classes may store validation rules that allow the formatting of validated clauses to be performed automatically. For example, the validation rules for an "AND" clause will specify that all requirements must be validated, the validation rules for an "OR" clause will specify that any one of the requirements must be validated whilst the validation rules for an "IF/THEN" clause specify that if the leading condition requirement is validated, then the consequence condition must also be validated. To determine whether the entire clause can be validated, the software checks the compliance fields of the element record for a compliance status. If the necessary requirement records have a complied status in the compliance field, the entire clause

is validated and the complete compliance field of the element record is updated with a validated status. A validated status in the complete compliance field triggers the display of the entire clause to be highlighted accordingly.

In Table 2 is an example of a contract which a user wishes to navigate. Figures 5 and 6 show a document map that is presented to the user on the system interface. Table 5 illustrates the steps in navigating the document, the step numbers are also indicated in Figures 5 and 6. In the present example, the user is attempting to determine whether clause 4.2 is satisfied. Starting at the first branch of the "Multiple" or AND node in Figure 5, and as written at step 1 in Table 5, the user has verified that the cross referenced clause 4.4 has been complied with and thus 4.3(a) has been complied with. At step 2, the user has validated that cross referenced clause 4.5 has been complied with and thus the map shows that clause 4.3(b) also indicates compliance. At step 3, investigation reveals that the requirements of clause 4.3(c) have been met and the user indicates this through the interface (Figure 6). Clause 4.3(d) contains a leading conditional requirement (IF/THEN) cross referenced to clause 4.6.2. The user tests clause 4.6.2 to see if the conditional requirement is met. If the conditional requirement is not met, then 4.3(d) will be automatically satisfied, otherwise, if the conditional requirement of 4.6.2 is met, then the consequential requirement of 4.6.5 will also have to be satisfied. These are steps 4 5 and 6 in Figure 6.

Step	Compliance Validation
1	Investigation validates that the joint preparation is prequalified in accordance with Clause 4.4. As this requirement complies, the subclause 4.3(a) complies.
2	Investigation validates that prequalified consumables to Clause 4.5 are being used. So, subclause 4.3(b) complies.
3	Investigation validates that workmanship and welding techniques comply. So, subclause 4.3 (c) complies.
4	Investigation into Clause 4.6.2 reveals that a macro test is required.
5	Investigation validates that there is documentary evidence of a macro test in accordance with Clause 4.6.5 So, subclause 4.3 (d) complies.

6	Clause 4.3(d) complies as the condition in step 4 is required and the consequence in step 5 is complied with.
7	Clause 4.3 complies as the embedded requirements in trailing consequence has been complied with. That is Clauses 4.3 (a), (b), (c) and (d) have been complied with.
8	Clause 4.2(a) complies as it refers to Clause 4.3, which compiles.
9	Clause 4.2 complies as one of its options, Clause 4.2(a) complies.

Table 5

Because section 4.3 identifies a "Multiple requirement embedded in a trailing conditional requirement" element (IF[AND]) all requirements within the trailing consequence requirement of section 4.3 will need to be complied with before clause 4.3 is complied with. Step 7 shows that Clause 4.3 complies as the embedded requirements in trailing consequence has been complied with. That is Clauses 4.3 (a), (b), (c) and (d) have been complied with. As indicated at steps 7 and 8, compliance of 4.3 validates compliance of clause 4.2(a). The sections of clause 4.2 are linked by "Choice of" element and thus only one requirement need be validated as complied with in order to validate clause 4.2 as at step 9.

In Figure 7 ,the effect of a non-conforming clause 4.3(a) step 1 on clause 4.2(a) is shown. The latter clause consequently does not conform in that the multiple requirements of this clause are not complied with. In order to comply either of the following needs to be complied with 4.3(a) or, 4.2 (b), (c), (d), or (e).

Each unique requirement box on the validation interface may be divided. With one side detailing the requirement and the other detailing compliance evidence. Figure 8 expands on Figure 7. It details compliance evidence for steps 2,3,4 and 5. Table 6 also details these compliance steps.

Step	**Clause**	**Compliance evidence**
2	4.3b	Pre-qualified consumable xxxx, from Supplier xxxx.
3	4.3c	Client inspection record dated dd/mm/yy.

4	4.3(d) Condition	Visual inspection details that a macro test is required
5	4.3d Consequence	Macro test record dated dd/mm/yy.

Table 6

The above validation example uses past actions known to the user to determine whether a clause or indeed an entire document has been complied with. In a further embodiment, the validation application allows a user to determine an action plan for validating a clause or document. Clause 4.2 identified in Figure 9 is an "OR" clause, that is only one requirement of the clause needs to be validated. In the example, the user is formulating an action plan for validating the clause according to subclause 4.2(a). This sub-clause cross references clause 4.3 which contains four "Multiple" class requirements. Moving further through the document map, it can be determined that in order to validate the Multiple requirements clause, five individual requirements must be validated. These five requirements can be considered the terminating requirements and can be mapped into an action plan such as in the table 7 below.

Step	Action
1	**Ensure** that joint preparation are prequalified in accordance with Clause 4.4
2	**Ensure** that welding consumables are prequalified in accordance with Clause 4.5
3	**Ensure** that the workmanship and welding techniques, including the preheat and interrun temperature requirements comply with this Standard.
4	**Check** whether Clause 4.6.2 requires macro testing. documentary evidence is available of a satisfactory macro test in accordance with Clause 4.6.5
5	If Clause 4.6.2 requires macro testing, then **ensure** that documentary evidence is available of a satisfactory macro test in accordance with Clause 4.6.5.

Table 7

The action plan concisely identifies the steps a user must undertake in order to prove a clause has been validated. The action plan may further be exported to any known project management software application, for example MS Project™ and MS Outlook™ both produced by Microsoft Corporation.

A further use of the mapping system is in comparing validation options. Action plans for each option available to validate a clause can be determined. The action plans can be compared to determine the most suitable manner for complying with a clause, eg the most cost effective, the least resource or manpower intensive, etc.

Validation of a requirement may occur from the bottom up as described above, from top down or in any other suitable manner as appropriate.

The above described embodiment of Figures 5 to 7 includes an example of cross referencing. Cross referencing may be indicated explicitly on the map as illustrated for example by the link between clause 4.2(a) and the AND node of clause 4.3, or in an alternative embodiment by hyperlinking. Alternatively, cross referencing may be implicit in that it may not be displayed on the document map other than through text. However, validation of an implicit link may still cause the automatic validation of a dependent requirement.

The system of the invention can not only be used in navigating existing documents but can be linked with a newly created document to create a map as a document is drafted. Text sections can be accessed individually and amended with the effects of changes to an existing section or the creation of a new section being readily apparent from the document map.

In a further embodiment, the system database stores construction rules for constructing a document from a document map. An exemplary rule for the Multiple Requirements class would be that the text of each requirement is separated by a comma with the final two requirements being separated by an "and". A rule for the Conditional Requirement class would be that an "if" element is place before the text of the conditional requirement, followed by a "then" element, in turn followed by the consequential requirement. Thus a user may draft a document by first creating a document map and writing the text of the requirements into the requirement boxes. The construction rules then map from the mapping interface into a document interface eg Microsoft Word™ in accordance with the mapping rules. Because the mapping is conducted as the initial step, the user can be sure that the grammatical construction of the document is correct.

For maximum flexibility, a system user may amend the element class lists to add or delete document elements as required. The user may also amend the format and mapping rules including the way in which links between elements on this map are depicted, the colour, font etc of elements and the symbols used.

The database may store document elements for a plurality of languages in order that the system can be applied to a wider range of documents. For example, Figure 11 shows a document map for a chinese document. The system may store classification and format rules for each language to account for different grammatical structures between languages. Furthermore, the system of the invention can be used to verify the translation of a formatted document from one language to another because the document map for a document should be the same irrespective of the language in which is written. By comparing maps for a document before and after translation, any significant translation errors that cause a change in the document structure can be readily identified.

The system of the invention can be used in drafting and analysing many formatted documents including but not limited to contracts, purchase orders, instructions, standards, specifications, tenders, bids, leases, finance loans, investments, deeds, conveyances, agreements, rules, codes, insurance and other policies, government legislation and statutory regulations. In particular the system of the invention allows a user to readily identify the extent of options within a document and the effect of any changes to the document.

It will be appreciated by the skilled person to which the specification is addressed that numerous modifications and/or variations can be made to the above described embodiments without departing from the spirit or scope of the invention and that all such modifications and/or variations are intended to be embraced herein.

Claims

1. A document analysis system including a set of element classes, each element class having an associated set of document elements and an associated set of element class rules, identifying means for identifying one or more document elements within an original document, said identified document elements dividing the document into a plurality of document sections associated with one or more document elements, and storing means for storing the document sections in a database wherein said document sections are linked according to the element class rules of the document elements associated with a document section.

2. A document analysis system according to claim 1 wherein said document elements are text strings representing grammatical construct elements.

3. A document analysis system according to claim 1 wherein said grammatical construct elements are logical operators.

4. A document analysis system according to claim 1 wherein said document sections are requirements of a clause of said document.

5. A document analysis system according to claim 4 wherein a requirement record is created for a requirement, said requirement record including a requirement field, wherein text of a requirement is stored in said requirement field.

6. A document analysis system according to claim 5 wherein a requirement record further includes a compliance status field.

7. A document analysis system according to claim 4 wherein an element record is created for an identified document element, said element record containing links to the requirement records to which the document element is related.

8. A document analysis system according to claim 7 wherein an element record includes an element class field for storing the element class of a document element.

9. A document analysis method including the steps of :

- identifying one or more document elements within an original document,
- classifying identified elements into one or more of a plurality of element classes,

- dividing the document into a plurality of document sections linked by the document elements in accordance with a set of rules associated with the element classes to which the identified document elements belong, and
- storing the document sections in a database.

10. A document mapping system including a set of element classes, each element class having an associated set of document elements and an associated set of format and mapping rules, identifying means for identifying one or more document elements within an original document, and mapping means for creating and displaying a map of document sections linked by labels representing the respective documents elements associated with those document sections.

11. A document mapping system according to claim 10 wherein an element class includes one or more graphical display symbols for displaying an identified document element belonging to that class and document sections associated with an identified document element.

12. A document mapping system according to claim 11 wherein a graphical display symbol includes one or more text boxes for receiving text of a document section.

13. A document mapping system according to claim 10 wherein said mapping means further includes means for graphically linking two or more element symbols.

14. A document analysis system according to claim 11 wherein an element record is created in a system database when a graphical display symbol is added to said document map, said element record containing an element class field.

15. A document analysis system according to claim 14 wherein a section record is created in said system database when a graphical display symbol is added to said map, said section record including a text section field for receiving text of a document section associated with a document element.

16. A document analysis system according to claim 15 wherein an element record further includes one or more fields containing links to associated section records and/or a section record contains one or more fields containing links to an element record.

17. A document analysis system according to claim 16 further including graphical link means for creating and displaying a graphical link between two or more element symbols

or components of said symbols, wherein the creation of a graphical link between two element symbols creates a database link between the respective element and/or section records.

18. A document analysis system according to claim 10 further including validation means for indicating on said document map a validation status of a document section.

19. A document analysis system according to claim 18 wherein said validation means further includes display means for indicating on said document map a global validation status pertaining to two or more related document sections.

20. A document analysis system according to claim 19 wherein a global validation status is automatically determined from the validation status of the two or more related document sections.

21. A document analysis system according to claim 20 wherein an element class further includes an associated set of validation rules, and wherein a global validation status is automatically determined using the validation rules of the element class or classes associated with said two or more related document sections.

22. A document analysis method including the steps of :

- identifying one or more document elements within an original document,
- classifying identified elements into one or more element classes,
- dividing the document into a plurality of document sections in accordance with a set of rules associated with the element class of the identified document elements, and
- representing the original document as a map of document sections linked by labels representing the respective document elements pertaining to those document sections.

23. An interface for creating a map of a document having a plurality of document sections linked by associated document elements, said document elements belonging to one or more of a plurality of element classes, said interface including a menu of graphic display symbols, each display symbol representing one or more element classes, and a mapping display, wherein one or more display symbols include a text area for receiving text of a document section, wherein a user may select a symbol to display in said mapping section, wherein a user can link two or more symbols.

24. A mapping interface according to claim 23 wherein linking two or more symbols creates a logical link between the respective document sections of associated with said symbols.

25. A system for constructing a document from a document map, said document map including a plurality of logically linked graphical symbols of one or more classes and containing a plurality of text sections, said system including a database storing a plurality of document construction rules for said one or more classes, wherein said system converts said document map into a plain language document according to the document construction rules pertaining to the classes of said graphical symbols of said document map.

26. A system according to claim 25 wherein said system creates said plain language document by placing said text sections in an order determined by said document construction rules and linking said text sections with a text string specified by said construction rules.

27. A document analysis system including a set of element classes, each element class having an associated set of document elements and an associated set of format rules, and identifying means for identifying one or more document elements within an original document, wherein the system creates a representation of the original document formatted in accordance with the format rules for the element class of the document elements identified in the original document.

28. A document analysis method including the steps of:

- identifying one or more document elements within an original document,

- classifying identified document elements into one or more of a plurality of element classes and

- creating a representation of the original document formatted in accordance with a set of format rules associated with the element classes to which the identified document elements belong.