

# Requirements for technical specification

## OASIS Legal XML eContracts TC

Document title: Requirements for technical specification  
Author: Peter Meyer, pmeyer@elkera.com.au  
Contributors:  
Version no: Draft 0.01  
Date submitted: 5 October 2004

### Contents

<b>1</b>	<b>Introduction.....</b>	<b>i</b>
<b>1.1</b>	<b>Purpose of this document.....</b>	<b>1</b>
<b>1.2</b>	<b>Revision history.....</b>	<b>1</b>
<b>2</b>	<b>Interpretation .....</b>	<b>1</b>
<b>2.1</b>	<b>Definitions.....</b>	<b>1</b>
<b>2.2</b>	<b>Use of data flow diagrams.....</b>	<b>2</b>
<b>3</b>	<b>Relevant contract transactions .....</b>	<b>3</b>
<b>3.1</b>	<b>Sources of information .....</b>	<b>3</b>
<b>3.2</b>	<b>Types of contracts.....</b>	<b>3</b>
<b>3.3</b>	<b>Contracts with negotiated narrative terms .....</b>	<b>3</b>
3.3.1	Description.....	3
3.3.2	Handshake agreement process.....	5
3.3.3	Draft contract process .....	5
3.3.4	Negotiate contract terms process .....	6
3.3.5	Assent to contract process .....	7
3.3.6	Precedent harvesting process.....	7
3.3.7	Manage contract process .....	8
3.3.7.1	Description.....	8
3.3.7.2	Contract transaction activity management.....	9

3.3.7.3	Contract document management .....	10
3.3.8	Dispute resolution process.....	11
3.3.9	Archive completed contract process.....	11
3.3.10	Data flows.....	11
3.3.11	Interfaces .....	11
<b>3.4</b>	<b>Ticket contracts.....</b>	<b>11</b>
<b>3.5</b>	<b>Standard form business and consumer contracts.....</b>	<b>12</b>
<b>3.6</b>	<b>Click through contracts.....</b>	<b>12</b>
3.6.1	Description.....	12
3.6.2	Submit contract terms process.....	13
3.6.3	Assent to terms process .....	14
3.6.4	Download contract terms.....	14
3.6.5	Other sub processes .....	14
<b>3.7</b>	<b>Electronic commerce contracts .....</b>	<b>14</b>
3.7.1	Description.....	14
3.7.2	Assent to master contract process.....	15
3.7.3	Transaction instances process.....	15
<b>3.8</b>	<b>Computer negotiated contracts .....</b>	<b>15</b>
<b>4</b>	<b>The use of XML markup for contract documents.....</b>	<b>16</b>
4.1	The purpose of this section .....	16
4.2	The use of XML markup in contract authoring today.....	16
4.3	Available tools for authoring contract documents using XML.....	16
4.4	Current use of contract metadata .....	17
4.5	Current use of embedded data value markup for contracts .....	17
4.6	Current use of contract semantic languages .....	17
4.7	Machine generated XML markup .....	18
4.8	The need to transform XML markup into display formats.....	18
4.9	Likely future developments .....	18
4.10	Conclusions on the use of XML markup for contract documents .....	19

<b>5</b>	<b>Problems, objectives and scope.....</b>	<b>19</b>
<b>5.1</b>	<b>Purpose of this section .....</b>	<b>19</b>
<b>5.2</b>	<b>Contract drafting and negotiation .....</b>	<b>20</b>
5.2.1	Reported problems.....	20
5.2.2	Problem statement .....	20
5.2.3	Objectives .....	20
5.2.4	Scope assessment.....	21
<b>5.3</b>	<b>Click through contacts .....</b>	<b>21</b>
5.3.1	Reported problems.....	21
5.3.2	Problem statement .....	21
5.3.3	Objectives .....	21
5.3.4	Scope assessment.....	21
<b>5.4</b>	<b>Contract management – transaction activities.....</b>	<b>21</b>
5.4.1	Reported problems.....	21
5.4.2	Problem statement .....	22
5.4.3	Objectives .....	22
5.4.4	Scope assessment.....	22
<b>5.5</b>	<b>Contract management – contract document.....</b>	<b>22</b>
5.5.1	Reported problems.....	22
5.5.2	Problem statement .....	22
5.5.3	Objectives .....	22
5.5.4	Scope assessment.....	22
<b>5.6</b>	<b>Electronic commerce contracts .....</b>	<b>23</b>
5.6.1	Reported problems.....	23
5.6.2	Problem statement .....	23
5.6.3	Objectives .....	23
5.6.4	Scope assessment.....	23
<b>5.7</b>	<b>Computer negotiated contracts .....</b>	<b>23</b>
5.7.1	Reported problems.....	23
5.7.2	Problem statement .....	24
5.7.3	Objectives .....	24
5.7.4	Scope assessment.....	24

<b>6</b>	<b>Overview of the proposed specification .....</b>	<b>24</b>
<b>7</b>	<b>Functional requirements .....</b>	<b>25</b>
<b>7.1</b>	<b>Approach to requirements development .....</b>	<b>25</b>
<b>7.2</b>	<b>Precedent contract documents .....</b>	<b>28</b>
7.2.1	Overview .....	28
7.2.2	Precedent creation, storage & retrieval.....	28
7.2.2.1	Background to requirements .....	28
7.2.2.2	Issues for contracts with negotiated terms .....	28
7.2.2.3	Issues for standard form contracts .....	28
7.2.2.4	Issues for click through contracts .....	28
7.2.2.5	Issues for computer negotiated contracts.....	28
7.2.2.6	The role of XML markup .....	29
7.2.2.7	The role of a standard .....	29
7.2.2.8	Specific requirements .....	29
<b>7.3</b>	<b>Contract document creation .....</b>	<b>30</b>
7.3.1	Document assembly for new drafts .....	30
7.3.1.1	Background to requirements .....	30
7.3.1.2	Issues for contracts with negotiated terms .....	30
7.3.1.3	Issues for standard form contracts .....	31
7.3.1.4	Issues for click through contracts .....	31
7.3.1.5	Issues for computer negotiated contracts.....	31
7.3.1.6	The role of XML markup .....	31
7.3.1.7	The role of a standard .....	31
7.3.1.8	Specific requirements .....	31
7.3.2	Variables substitution in new drafts .....	31
7.3.2.1	Background to requirements .....	31
7.3.2.2	Issues for contracts with negotiated terms .....	32
7.3.2.3	Issues for standard form contracts .....	32
7.3.2.4	Issues for click through contracts .....	32
7.3.2.5	Issues for computer negotiated contracts.....	32
7.3.2.6	The role of XML markup .....	32
7.3.2.7	The role of a standard .....	33
7.3.2.8	Specific requirements .....	33

7.3.3	Custom authoring of contract terms .....	33
7.3.3.1	Background to requirements .....	33
7.3.3.2	Issues for contracts with negotiated terms .....	33
7.3.3.3	E commerce master .....	33
7.3.3.4	The role of XML markup .....	33
7.3.3.5	The role of a standard .....	33
7.3.3.6	Specific requirements .....	33
7.3.4	Collaborative editing of draft contract terms .....	33
7.3.4.1	Background to requirements .....	33
7.3.4.2	Issues for contracts with negotiated terms .....	34
7.3.4.3	The role of XML markup .....	34
7.3.4.4	The role of a standard .....	34
7.3.4.5	Specific requirements .....	34
7.3.5	Publish draft contract documents in multiple output formats .....	34
7.3.5.1	Background to requirements .....	34
7.3.5.2	Issues for contracts with negotiated terms .....	34
7.3.5.3	Issues for standard form contracts .....	34
7.3.5.4	Issues for click through contracts .....	34
7.3.5.5	The role of XML markup .....	34
7.3.5.6	The role of a standard .....	34
7.3.5.7	Specific requirements .....	34
7.3.6	Manage document versions during drafting .....	35
7.3.6.1	Background to requirements .....	35
7.3.6.2	Issues for contracts with negotiated terms .....	35
7.3.6.3	The role of XML markup .....	35
7.3.6.4	The role of a standard .....	35
7.3.6.5	Specific requirements .....	35
7.3.7	Exchange draft contract documents .....	35
7.3.7.1	Background to requirements .....	35
7.3.7.2	Issues for contracts with negotiated terms .....	35
7.3.7.3	Issues for standard form contracts .....	35
7.3.7.4	Issues for click through contracts .....	35
7.3.7.5	The role of XML markup .....	35
7.3.7.6	The role of a standard .....	35
7.3.7.7	Specific requirements .....	35

7.3.8	Produce assent document.....	36
7.3.8.1	Background to requirements.....	36
7.3.8.2	Issues for contracts with negotiated terms.....	36
7.3.8.3	Issues for standard form contracts .....	36
7.3.8.4	Issues for click through contracts .....	36
7.3.8.5	The role of XML markup .....	36
7.3.8.6	The role of a standard .....	36
7.3.8.7	Specific requirements .....	36
<b>7.4</b>	<b>Assent by parties to the contract .....</b>	<b>36</b>
7.4.1	Assent to contract terms .....	36
7.4.1.1	Background to requirements.....	36
7.4.1.2	Issues for contracts with negotiated terms.....	36
7.4.1.3	Issues for standard form contracts .....	37
7.4.1.4	Issues for click through contracts .....	37
7.4.1.5	Issues for e commerce master contracts .....	37
7.4.1.6	The role of XML markup .....	37
7.4.1.7	The role of a standard .....	37
7.4.1.8	Specific requirements .....	37
<b>7.5</b>	<b>Contract management – contract activities.....</b>	<b>37</b>
7.5.1	Extract contract states, obligations, rights etc into exchange format .....	37
7.5.1.1	Background to requirements.....	37
7.5.1.2	Issues for contracts with negotiated terms.....	37
7.5.1.3	Issues for standard form contracts .....	37
7.5.1.4	Issues for e commerce master contracts .....	37
7.5.1.5	The role of XML markup .....	37
7.5.1.6	The role of a standard .....	37
7.5.1.7	Specific requirements .....	38
7.5.2	Communicate contract states, obligations & rights to interested parties.....	38
7.5.2.1	Background to requirements.....	38
7.5.2.2	Issues for contracts with negotiated terms.....	38
7.5.2.3	Issues for standard form contracts .....	38
7.5.2.4	The role of XML markup .....	38
7.5.2.5	The role of a standard .....	38
7.5.2.6	Specific requirements .....	38
<b>7.6</b>	<b>Contract management – contract document.....</b>	<b>38</b>

7.6.1	Retain or preserve assent document .....	38
7.6.1.1	Background to requirements .....	38
7.6.1.2	Issues for contracts with negotiated terms .....	38
7.6.1.3	Issues for standard form contracts .....	38
7.6.1.4	Issues for click through contracts .....	39
7.6.1.5	Issues for e commerce master contracts .....	39
7.6.1.6	Issues for computer negotiated contracts .....	39
7.6.1.7	The role of XML markup .....	39
7.6.1.8	The role of a standard .....	39
7.6.1.9	Specific requirements .....	39
7.6.2	Prepare variations to contract .....	39
7.6.2.1	Background to requirements .....	39
7.6.2.2	Issues for contracts with negotiated terms .....	39
7.6.2.3	The role of XML markup .....	39
7.6.2.4	The role of a standard .....	39
7.6.2.5	Specific requirements .....	39
7.6.3	Maintain variations and contract versions .....	40
7.6.3.1	Background to requirements .....	40
7.6.3.2	Issues for contracts with negotiated terms .....	40
7.6.3.3	The role of XML markup .....	40
7.6.3.4	The role of a standard .....	40
7.6.3.5	Specific requirements .....	40
7.6.4	Publish contract terms to interested persons .....	40
7.6.4.1	Background to requirements .....	40
7.6.4.2	Issues for contracts with negotiated terms .....	40
7.6.4.3	The role of XML markup .....	40
7.6.4.4	The role of a standard .....	40
7.6.4.5	Specific requirements .....	40
7.7	<b>Other processes .....</b>	<b>41</b>
7.7.1	Map negotiated parameters to contract terms .....	41
7.7.1.1	Background to requirements .....	41
7.7.1.2	Issues for computer negotiated contracts .....	41
7.7.1.3	The role of XML markup .....	41
7.7.1.4	The role of a standard .....	41
7.7.1.5	Specific requirements .....	41

# 1 Introduction

## 1.1 Purpose of this document

The mission of the TC is to provide a standard for XML markup that should enable “the efficient creation, maintenance, management, exchange and publication of contract documents and contract terms”.

This mission statement provides a broad scope for the TC's work but does not define any problems that are to be addressed or why the use of XML markup or a standard may be relevant. If the TC is to be able to develop a specification that will be useful, it is necessary to identify the real problems that occur in relation to contract transactions and a practical specification that is based on an understanding of the way relevant transactions are carried out and the needs of relevant users.

This document identifies the business problems relating to the preparation and management of various kinds of contracts, the persons affected by those problems and the business needs of those persons to overcome those problems. Within that framework, it defines the functional characteristics an application must have to meet those needs.

Those requirements will enable the TC to design a technical specification to satisfy the identified requirements.

## 1.2 Revision history

Document versions are listed in the table in reverse chronological order.

Date/version	Description	Author
Draft 0.01 5/10/2004	Initial draft after San Francisco face to face meeting with background information, vision statement and outline of requirements topics.	Peter Meyer

# 2 Interpretation

## 2.1 Definitions

In this document:

***assent document*** means the document chosen by the parties to record their assent to the contract terms.

***contract*** means an agreement between parties that is intended to be legally enforceable. A contract may be oral, partly oral and partly written or wholly recorded in writing. The terms of a contract may be incorporate many contract documents.

***contract document*** means a document that records draft or agreed contract terms.



**contract metadata** means information about a contract or particular contract terms that is not embedded in the narrative terms.

**contract semantics language** means a language that can express the rights and obligations of parties to a contract in a form that can be parsed by software applications and processed with other data to determine contract states.

**document** includes information in printed (hard copy) or electronic form.

**embedded data value** means a piece of information such as a product or service description, date, name, address, quantity or monetary amount that is embedded in the narrative terms.

**generic structural markup** defines an ordered hierarchy of narrative components of a document such as clauses and paragraphs. This markup enables processing applications to determine that a marked up component belongs to a particular genus such as clause, paragraph, list item but it does not, on its own define anything about the particular function of those objects. For example, in a contract, the markup would not differentiate between a payments clause or a warranties clause. Generic structural markup is commonly accompanied by metadata and embedded data value markup.


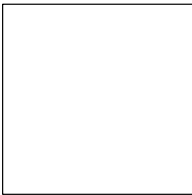

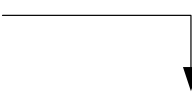
**precedent contract** means a document that is used by the drafter of a new contract document as a starting point or template to assist in creating that new contract document.

**TC** means the OASIS Legal XML eContracts Technical Committee.

## 2.2 Use of data flow diagrams

Data flow diagrams are used to identify the important processes, data flows and interfaces for each type of contract transaction considered in these requirements. They provide a generalised view of common situations for analytical purposes.

The symbols used in the data flow diagrams are explained in the following table.

	<p>Process</p> <p>An action or activity performed on data, either manually or automatically.</p>
	<p>Interface</p> <p>A person or system (document) that interacts with a process to either receive or input data.</p>
	<p>Data store</p> <p>A storage system for data used by the process.</p>
	<p>Data flow</p> <p>This shows the flow of information between an interface or data store and a process.</p>

..... A dotted line shows an optional flow.

---

### **3 Relevant contract transactions**

#### **3.1 Sources of information**

In 2003 various TC members contributed scenario documents describing contract transactions and their anticipated needs from the specification. These are available for inspection at ##.

In August and September 2004, some TC members prepared use case statements to define in more precise terms the actions undertaken by users in particular contract transactions. These used cases are available at ##.

The overall framework used in this document is based on the use case analysis undertaken at the face to face meeting of TC members in San Francisco on 19 and 20 September 2004.

#### **3.2 Types of contracts**

The following sections define the contract types identified as posing business problems that may be addressed by a technical specification.

The TC is not concerned with contracts that are wholly oral. Only those contracts that are at least partly evidenced in writing are considered.

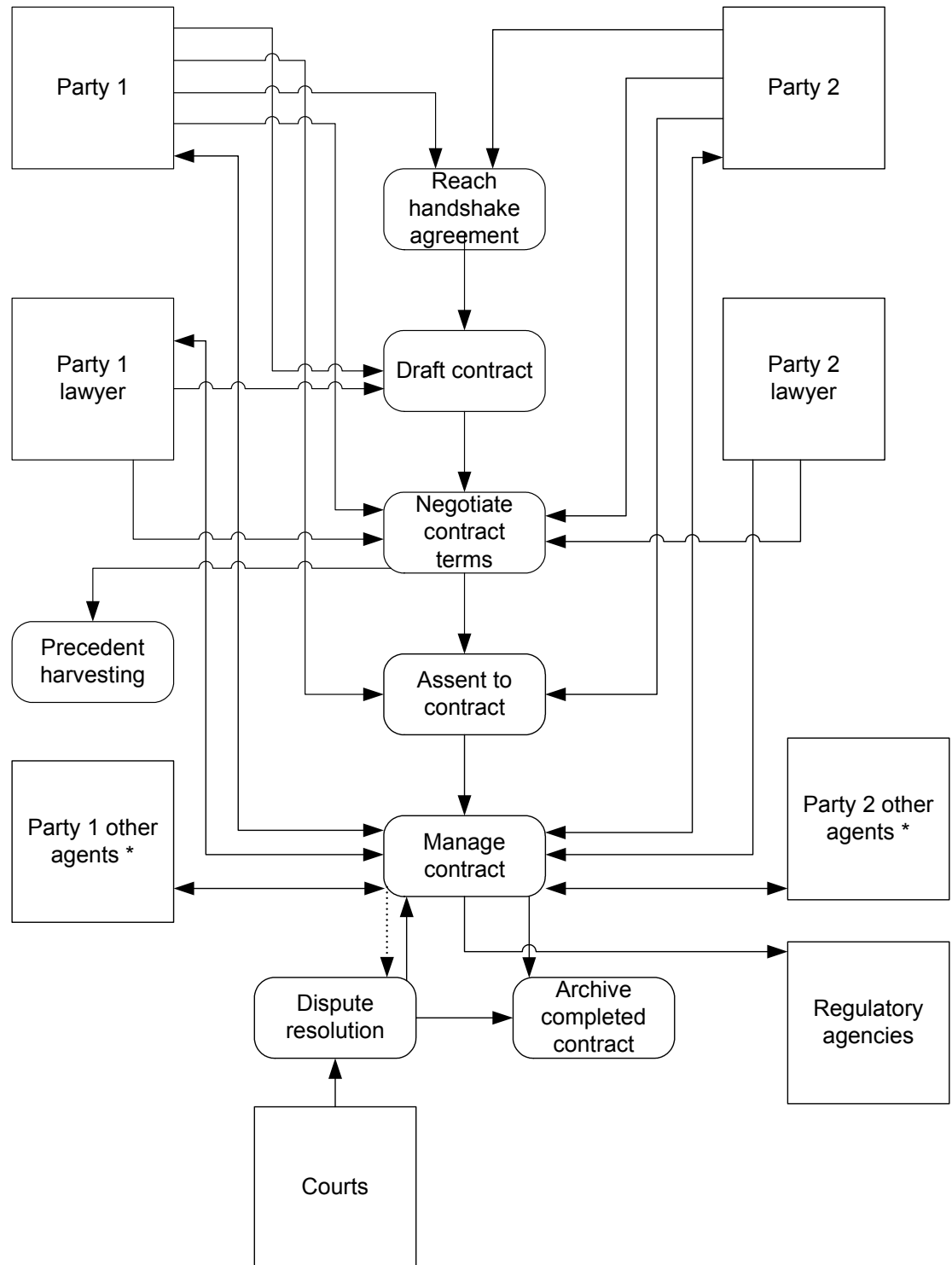
#### **3.3 Contracts with negotiated narrative terms**

##### **3.3.1 Description**

This case describes contracts that involve a negotiation over some or all the narrative terms of the contract.

Contracts with negotiated narrative terms are usually prepared by lawyers on behalf of their clients. They are the most common contract documents created by law firms and corporate and government legal departments.

The life cycle of contracts with negotiated terms is shown in Figure 1.

**Figure 1 Overview of negotiated contracts life cycle**

\* Other agents of the parties includes persons such as architects, accountants and other consultants etc. assisting with or interested in the transaction.

### 3.3.2 Handshake agreement process

This process is included because it is logically the first step. It does not raise any relevant issues for these requirements.

### 3.3.3 Draft contract process

The common components of this process are described in Figure 2.

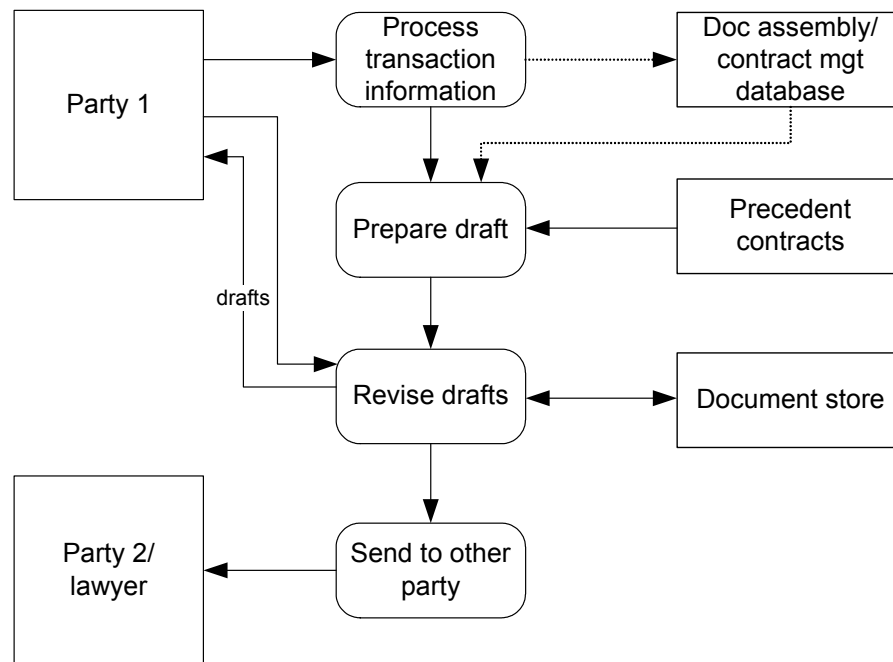
Almost without exception today, the drafting of custom contract terms is carried out using word processing software such as Microsoft Word. For all practical purposes, no lawyers are using XML based editing tools for this purpose. However, some software tools for document assembly may rely on precedent documents being stored in XML format to facilitate software processing. Lawyers using such software normally would be unaware of the use of XML behind the scenes.

Commonly, document assembly systems store transaction data such as names, addresses, monetary amounts and dates. These values may be automatically inserted into the draft contract document as embedded data values.

The draft contract documents so created may be based entirely on precedent contract terms, entirely on custom drafted terms or a mixture of precedent and custom terms.

When creating contract documents, authors commonly need to create other transaction documents such as letters, forms, notices and minutes of meetings. These documents may be created using the same precedent and document assembly systems as are used for contract documents.

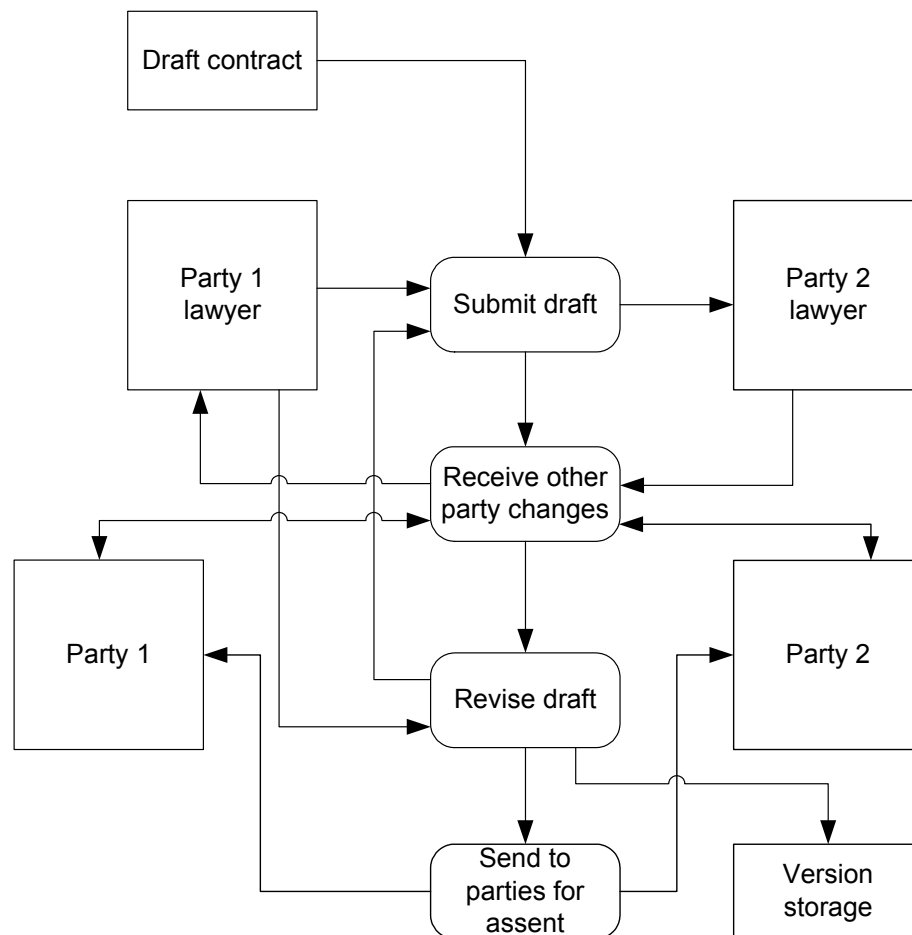
Draft contract documents must be communicated to clients and other interested persons. It is often desired to provide access in various formats such as RTF, PDF, HTML etc according to the circumstances.

**Figure 2 Contract drafting processes**

### 3.3.4 Negotiate contract terms process

This process is described in detail in Figure 3.

In some larger transactions the negotiating parties may want shared access to draft documents so they can contribute amendments, new terms and comments. Software collaboration platforms may be used to assist this process. This approach can be more convenient than the traditional approach where one party makes all changes to a draft document, sends it to the other party for review and awaits return of a package of comments and suggested amendments before repeating the process. This collaboration process involves making contract terms available in a variety of formats such as RTF, PDF and HTML, according to the role of the recipient.

**Figure 3 Negotiating the terms of a draft contract**

### 3.3.5 Assent to contract process

Assent to a contract with negotiated contract terms is usually signified by the parties signing a printed copy or copies of the contract document. It is relatively uncommon for parties to apply digital signatures to electronic contract documents under this scenario.

It is not uncommon for the parties to make last minute changes to contract terms at the time of assent. This may involve correction of a drafting error, an agreed change or the insertion of information that is dependant on the actual time of assent. Such changes may be noted by hand on the printed contract document if there are no facilities for editing the electronic source files and printing new documents.

The document bearing the signatures of the parties becomes the evidentiary contract document.

### 3.3.6 Precedent harvesting process

Lawyers who prepare contracts are often concerned to capture new contract terms for later re-use in other, similar transactions. This is part of their knowledge management process.

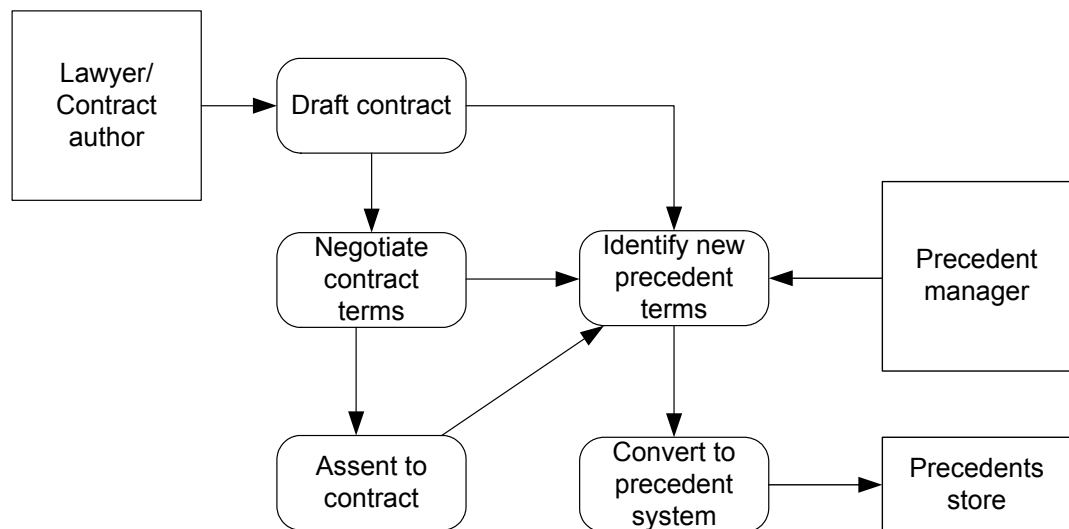
Precedent harvesting may be undertaken at various times in relation to the preparation of contract documents. For convenience it is shown in the overview in a slightly idealised form as occurring after the contract document terms are settled. The main components of the process are shown in Figure 4.

The precedent harvesting process does not apply only to contract documents. Lawyers apply the same process to other documents they prepare, including advices and litigation documents.

The harvesting process is quite difficult because of the bulk of content that must be reviewed. In practice it is difficult to require contract authors to categorise specific contract terms during the drafting process. Their attentions are directed to meeting the immediate needs of their clients and there is often little time to take on ancillary tasks.

If the enterprise uses back end XML for its precedent and document assembly systems, it will need to convert harvested content to that XML markup. This makes it difficult to publish new precedent content quickly and adds to the expense of maintaining the system.

**Figure 4 Harvesting precedent contract terms from transaction documents**



### 3.3.7 Manage contract process

#### 3.3.7.1 Description

Many, but not all, negotiated contracts are executory and involve ongoing obligations of the parties that may need to be monitored and managed until the transaction contemplated by the contract is completed or the comes to an end.

The contract management process is considered to involve two key sub processes:

- (a) manage the transaction activities (rights and obligations), as described in section 3.3.7.2.
- (b) manage the contract document and its publication to the parties, as described in section 3.3.7.3.

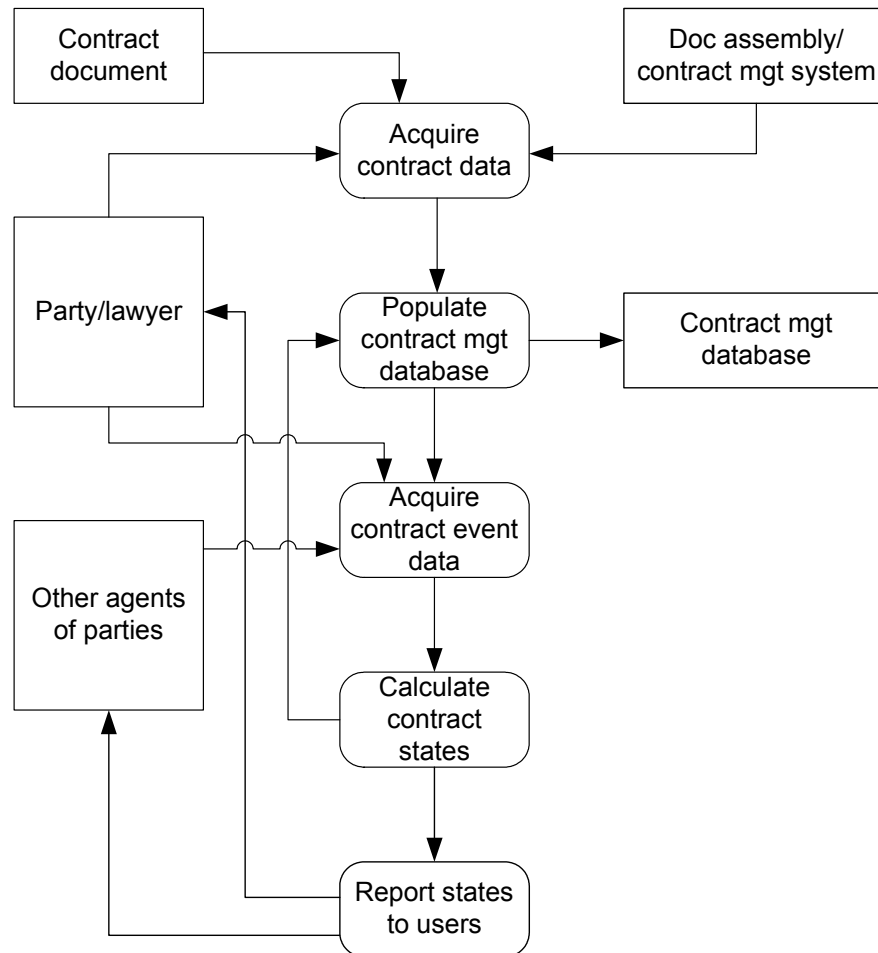
### 3.3.7.2 Contract transaction activity management

This process is described in Figure 5. It involves the management of the continuing contractual obligations of the parties.

Key points about the contract management process include:

- (a) The data needed for contract management may be derived from multiple sources. The contract document is likely to be the source for only some of this information. At the very least, it will be common to input new information as contract events are undertaken and completed and as terms are varied.
- (b) Some information may already exist in a database system used to generate the contract document.
- (c) Information cannot be extracted from the contract document unless that information is in a machine readable form. For all practical purposes, no contract documents produced today under a negotiated contract terms process are in a reliable machine readable format.
- (d) Contract management can be undertaken at certain levels without software tools. At other levels, simple software tools such as spread sheets or desktop project management software (eg, Microsoft Project) may be used. In more complex situations specialised contract management database applications are required to capture and manage the relevant information, calculate contract states and provide the reports desired by the interested persons.

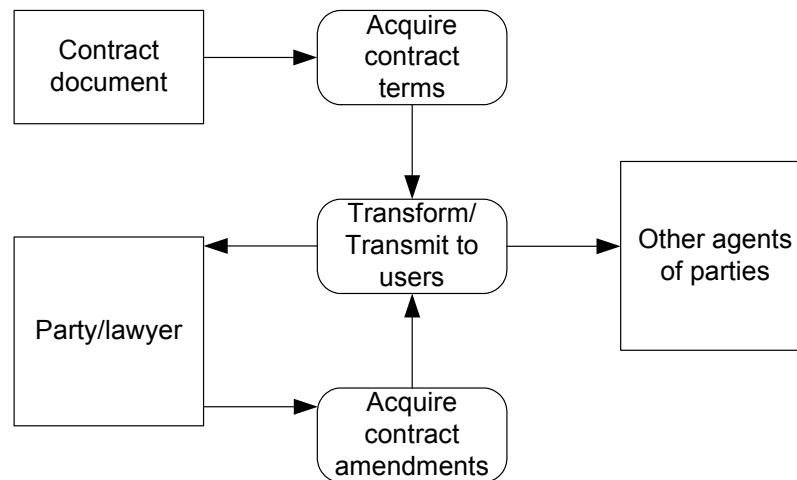


**Figure 5 Overview of transaction activity management**

### 3.3.7.3 Contract document management

This process is described in Figure 6.

Management of the contract document is important in transactions that run for a lengthy period with multiple persons who must perform obligations under the contract. Construction contracts are a good example. During the course of the project, there may be amendments to the contract terms that must be communicated to relevant agents of the parties. These persons may be operating under diverse conditions and may require access to contract documents in formats convenient to their circumstances. These include hard copy, searchable and browseable electronic copies, portable device access and even text to speech access.

**Figure 6 Managing the contract document**

### 3.3.8 Dispute resolution process

If a contract dispute arises, a formal dispute resolution process may be invoked by the parties.

Dr Leff proposed a use case that involves automated dispute resolution for electronic business contracts where the contract rights and obligations are expressed in machine readable form and there is no factual dispute between the parties. This is considered out of scope of the TC's work at this time.

### 3.3.9 Archive completed contract process

This step is shown for completeness. It raises no issues for the development of the TC's specification.

### 3.3.10 Data flows

At this stage, little is to be gained from a detailed analysis of the data flows. Sufficient information is provided by Figure 1 and the process analysis.

### 3.3.11 Interfaces

At this stage, little is to be gained from an analysis of specific interfaces. Sufficient information is provided by the process analysis.

## 3.4 Ticket contracts

Ticket contracts are those where a printed set of contract terms is offered to a buyer at or near to the time of purchase of a good or service. Examples include travel and parking station tickets.

This category also covers contracts such as “shrink wrap” software licenses that are included in physical software packaging.

The contract documents are normally prepared for the supplier by a process that is effectively the same as that described in Figure 2 (Contract drafting processes).

If the contracts are offered online in electronic form, ticket contracts are the same as click through contracts.

Printed ticket contracts raise no issues relevant to the work of the TC and are not considered further in these requirements.

### 3.5 Standard form business and consumer contracts

This case describes contracts offered with a service in circumstances where the offeror will not accept negotiation of contract terms. Examples include housing finance, car financing and most insurance contracts. The offered services may be available from many suppliers on different terms. Consequently, the negotiation is effected by shopping around different service providers.

Unlike the ticket contracts, contract documents produced under this case bear more resemblance to negotiated contracts in the way they are produced. Some may be generated by document assembly systems to align the contract document with selected service options. Data values may be directly incorporated into or attached to the contract terms.

These contracts may be assented to in paper form or in an online transaction. If the contracts are offered online in electronic form, these contracts are the same as click through contracts discussed in the next section.

The standard form contract documents are normally prepared for the supplier by a process that is effectively the same as that described in Figure 2 (Contract drafting processes). These contracts raise no distinct issues for the TC's work.

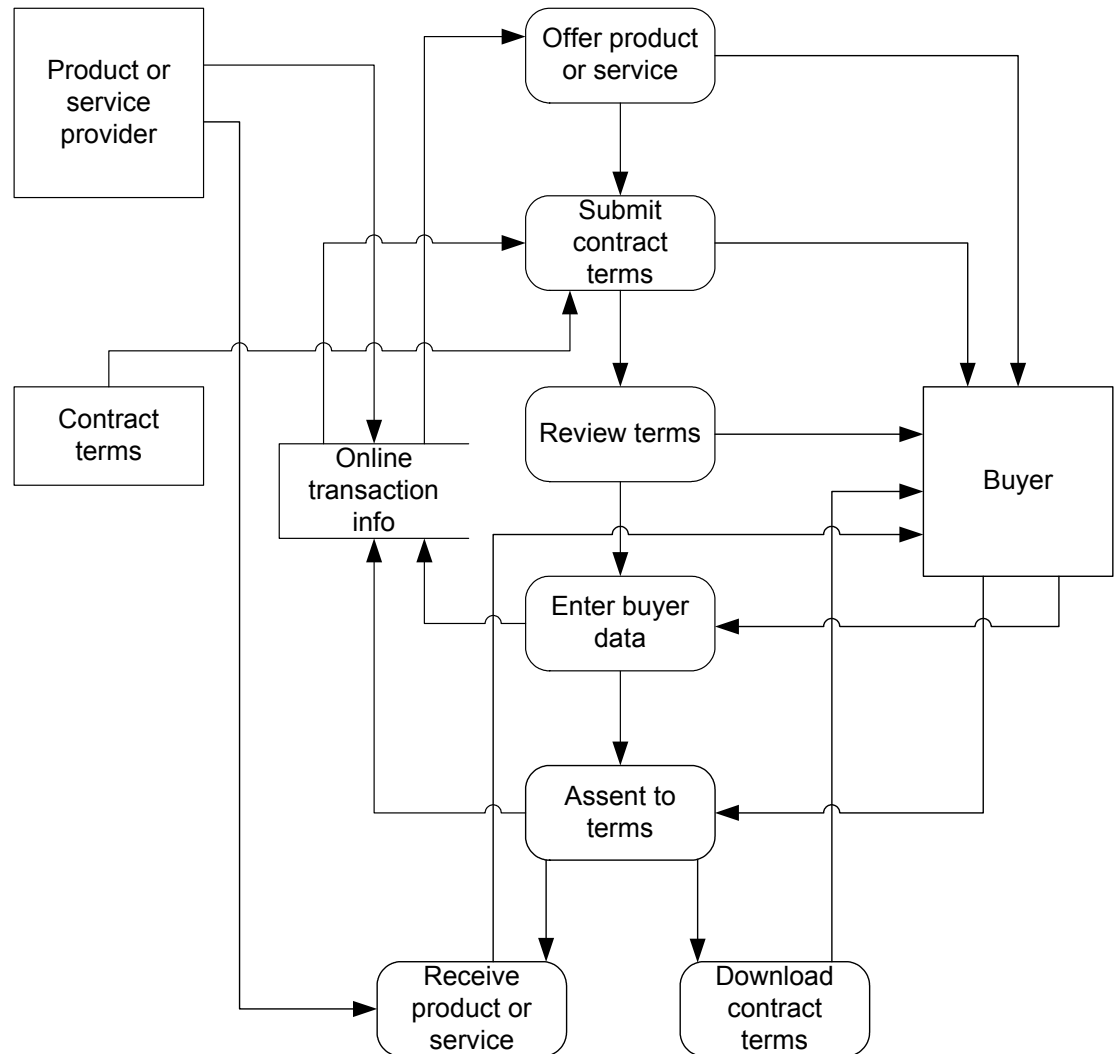
### 3.6 Click through contracts

#### 3.6.1 Description

Click through contracts are really just electronic versions of ticket contracts or standard form business or consumer contracts. They involve a standard form contract document that is offered to a purchaser of goods or services during an on-line transaction or during software installation. The processes involved in an on-line click through contract are described in Figure 7.

Click through transactions are widespread on the World Wide Web.

**Figure 7 Click through contract overview**



### 3.6.2 Submit contract terms process

The service provider presents its standard contract terms for the selected service. The buyer is not able to negotiate these terms and must either accept the terms in full to receive the product or service or reject them and do without.

Some sites provide a way to access the standard contract terms at any time so they can be reviewed in advance, if desired. This is not invariable.

The contract terms presented are normally prepared for the service provider under a process that is basically the same as that described in Figure 2 (Contract drafting processes). The contract document is prepared once and used many times.

### 3.6.3 Assent to terms process

Assent is usually signified by the buyer clicking an “I agree” button on screen with a copy of the contract terms displayed. The online transaction system record this event to initiate the next steps in the transaction.

### 3.6.4 Download contract terms

The buyer may or may not have a convenient method to download a copy of the contract terms as a record of the transaction. Commonly, sites allow users to save a copy of the terms in HTML or PDF format and to print them. The document so created usually does not represent the complete terms of the contract between the parties. The transaction particulars such as product, quantity, price and date are likely to be set out in a separate, automatically generated invoice or receipt.

### 3.6.5 Other sub processes

At this stage it appears sufficient information is provided by this high level representation of click through transactions.

## 3.7 Electronic commerce contracts

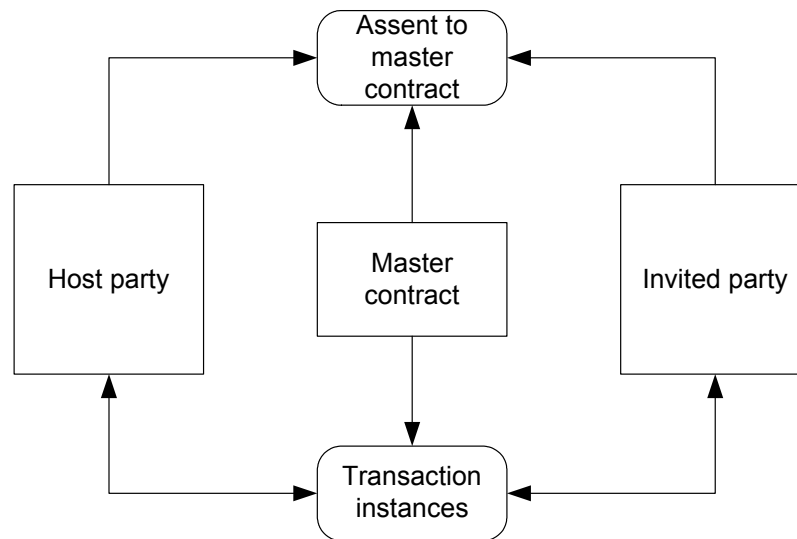
### 3.7.1 Description

Commonly, electronic commerce contracts are set up between a large enterprise that wishes to do business electronically with its customers or suppliers for particular goods and services. Under this model, there is a host party that initiates the formation of the system and invited parties who must join the system if they wish to do business with the host.

Alternate models involve peer to peer exchanges under which all parties are essentially equal.

Electronic commerce transactions of this kind are not yet widespread. They are mainly confined to use by very large corporations such as aircraft or motor vehicle manufacturers and their suppliers.

An overview of the hosted model is described in Figure 8.

**Figure 8 Electronic commerce contracting system overview**

### 3.7.2 Assent to master contract process

The master contract is typically prepared by the host party and presented to the invited parties. In the early stages of the establishment of the system, it is likely that negotiation will occur over the terms of the master contract. This will take place outside the system.

The parties could assent to the master contract by signing paper copies in much the same way as the negotiated contract terms scenario or by signifying their assent online in a similar way to the click-through contracts scenario. This may involve use of digital signatures.

The master contract is normally in narrative form and would be prepared in the same way as a draft contract described in Figure 2 (Contract drafting processes).

### 3.7.3 Transaction instances process

Transaction instances occur when the parties wish to buy or sell a product or service provided by the master agreement. The master agreement specifies the procedures that must be followed through enquiry, offer and acceptance to create a binding transaction. The electronic trading system allows the parties to transmit machine readable messages to effect these transactions. Protocols for the machine readable messaging are provided by standards such as ebXML.

## 3.8 Computer negotiated contracts

Several researchers have proposed systems that will enable computer systems to automatically negotiate contracts to buy or sell commodity goods. These systems involve the specification of transaction parameters that can be understood by both sides to the negotiation.

The set of parameters that results from the negotiation must be mapped to a set of contract terms to provide a complete contract. This could occur simply by referring to a master agreement in the same way as for the electronic commerce case. Alternatively, the parameters may map to a set of contract terms derived from an agreed set of available terms in a way that is similar to a document assembly process for contract document drafting.

Processes involved in automated negotiation are outside the scope of the TC's work.

## **4 The use of XML markup for contract documents**

### **4.1 The purpose of this section**

It is envisaged that various aspects of the TC's specification will be based on the use of XML for contract document preparation and contract management. This section provides a brief review of the key factors affecting the use of XML markup for contract documents.

### **4.2 The use of XML markup in contract authoring today**

As far as the TC is aware, almost all contract documents today are prepared in formats used by conventional word processing software such as Microsoft Word, Corel WordPerfect and other similar applications.

The TC has no evidence that lawyers in law firms and enterprise legal departments are using XML authoring applications to create contract documents.

There is evidence that XML is used to markup precedent contract documents for use in some document assembly systems. These systems must rely on the product vendor's proprietary XML schema since there are no applicable standards at this time. Authors interact with the precedents through an interface that shields them from the underlying document markup. Once the assembly process is complete, the document is converted to a word processing format so that any custom editing can be completed using common applications familiar to the authors. For convenience, this is referred to as "back end XML".

It is possible for users of Microsoft Office 2003 to create XML documents using the Microsoft WordML schema, provided that the enterprise concerned develops the necessary style mappings and import and export scripts. The WordML schema is a proprietary schema. The TC has no evidence that there is significant use of WordML format to create legal documents at this time.

Where XML markup is used for documents it is usually a generic, structural markup designed to support content chunking for document assembly, data value insertion and automated single source publishing.

### **4.3 Available tools for authoring contract documents using XML**

There are commercial XML authoring applications that can be configured to create contract documents using XML markup. Use of an XML authoring

application requires a suitable schema or DTD that will allow the contract document to be transformed into the desired layout formats such as RTF, PDF or HTML etc.

The TC is not aware of any public standard schema that are specifically designed for marking up contract documents. All available schema would require substantial modification to be useful.

Commercial XML authoring applications must work with many different schemas or DTDs. Many of the authoring conveniences found in word processing applications are not available out of the box with commercial XML authoring applications. Usually, it is necessary to undertake a great deal of configuration development to create a tool that is convenient for authors to use. This configuration is usually specific to each DTD or schema.

Based on currently available tools, this development requires specialised expertise and involves considerable effort and expense.

#### 4.4 Current use of contract metadata

Contract metadata is used extensively to categorise content for retrieval purposes, to manage versions and to support publishing processes.

Word processing applications make it difficult to associate metadata with individual content chunks.

XML markup makes it easy to add metadata to any chosen component. Metadata is commonly used with generic structural markup in XML documents.

#### 4.5 Current use of embedded data value markup for contracts

Embedded data value markup is not easily implemented in word processing documents but it is commonly used with generic structural markup in XML documents.

#### 4.6 Current use of contract semantic languages

The TC is not aware of any instance where contract semantic languages are in commercial use at this time.

##### **Drafting note**

*Is this correct?*

Some TC members and other persons in several parts of the world are undertaking research and development on XML based contract semantic languages. These languages could repeat the semantics of the human readable narrative terms within or outside the contract document. It is conceivable that they could be used instead of that content.

##### **Drafting note**

*We may need to add in an explanation of these languages so we better understand their application and state of development.*



## 4.7 Machine generated XML markup

It is possible to generate a contract document using XML markup, either from another XML document or from chunks of text in a database. This approach is not useful for negotiated contracts where a lawyer or other contract author must modify the draft terms using commonly available desktop software.

## 4.8 The need to transform XML markup into display formats

Transformation or style rules must be applied to XML documents to enable them to be published in the desired format (RTF, PDF, HTML, etc) and according to the desired presentation layouts and styles. A variety of methods and tools exist for doing this. At this time, there is no universal approach that provides all the functionality required. The TC expects that for the foreseeable future, various enterprises may wish to use different tools to transform XML markup to display formats.

The desire to create multiple output formats from a single source is one of the key reasons why enterprises might use XML markup for documents. However, based on currently available tools, the planning and development of transformation rules or styles for a new DTD or schema requires specialised expertise and involves considerable effort and expense.

## 4.9 Likely future developments

Many new standards and products are being developed that may affect the way XML may be used for document production.

On the evidence available to the TC, it is apparent that law firms and other enterprises with corporate legal departments are not demanding to create documents using XML authoring tools. On the contrary, they need to be convinced there are good reasons to do so. They need to be convinced that suitable tools exist to make it easier, rather than harder, for authors to create documents to justify undertaking the change management and development effort.

Microsoft has introduced XML capabilities into Word 2003. It is extending the tools available for the use of XML and it is likely to make further changes to its Office applications in future releases. This will greatly affect future directions because it directly affects the tools most contract authors use today.

The application of contract semantic language markup to a contract is likely to be a difficult exercise for contract authors because specialised skills will be needed to create it. It would be more easily applied to relatively standard form contracts but difficult to apply to custom drafted contracts.

The TC cannot assess the likely directions in the development of contract semantic languages. If they are successful, they could be used in master contracts for electronic commerce transactions and in other long term business contracts where there is an opportunity to set up automated monitoring of activities that occur under the contract. For example, some proponents point to service level agreements for technology systems as an area where the use of contract expression languages would be valuable.

Based on the assumption that parties to contracts will not prepare and exchange contract documents in XML formats in the foreseeable future, there will be little opportunity to add contract semantic language markup to contract documents. It is likely that contract semantic language markup will be more useful and easily implemented if it is separated from the human readable contract documents.

#### **4.10 Conclusions on the use of XML markup for contract documents**

- 4.10.1 There is little likelihood that contract authors will prepare documents using XML authoring applications in the foreseeable future. They will continue to create contract documents using word processing applications and exchange word processing documents.
- 4.10.2 There is no infrastructure in place for parties to exchange and process XML contract documents. This situation is unlikely to change in the foreseeable future.
- 4.10.3 Even where back end XML is used to facilitate document assembly and other automated processing, there is no immediate prospect that XML documents will be distributed outside the enterprise.
- 4.10.4 Metadata and embedded data value markup can be utilised at the contract drafting stage by enterprises that use back end XML systems but this markup is not likely to be available to other parties or interested persons if there is little or no exchange of XML documents.
- 4.10.5 It will not be practicable to incorporate contract semantic language markup to most contracts with negotiated narrative terms. Firstly, such contracts will not use XML markup in the foreseeable future. Secondly, even if they do use XML markup, it is very unlikely that drafters would add contract expression markup during contract drafting of custom contract terms. It would be more practicable to conceive of contract expression language markup being applied to relatively standard form contracts.
- 4.10.6 Contract expression language markup will be more widely implemented if it is maintained separately from the contract documents.

### **5 Problems, objectives and scope**

#### **5.1 Purpose of this section**

This section firstly identifies the problems encountered by various users (interfaces) involved in processes identified earlier.

High level objectives are then defined to deal with those problems. These statements provide a reference point against which all requirements can be validated..

Finally, each contract type or process is assessed to determine its relevance to the TC's mission.

The problem and objective statements are based on the discussions at the face to face meeting in San Francisco in September 2004, with some minor adjustments.

## 5.2 Contract drafting and negotiation

### 5.2.1 Reported problems

It's difficult to track contract changes at the level of contract terms.

It is hard to re-use content from transaction documents, reducing knowledge management benefits and increasing costs.

The drafting cycle during negotiation is slow, adding to cost.

Lawyers are expensive, clients would like to be able to do more themselves.

Precedent maintenance is expensive, thus limiting access to precedents, increasing risk and cost.

How to translate contract terms into machine readable language.

### 5.2.2 Problem statement

Today, contract documents are created using word processing applications. These documents can't easily be processed at convenient levels of granularity. It is difficult to process these documents outside the creating application. This inhibits automated document creation, information reuse, information extraction and change traceability.

### 5.2.3 Objectives

The TC specification will enable:

- (a) contract terms to be managed and manipulated at convenient levels of granularity to facilitate:
  - (i) improved access to precedent contract terms and contract documents
  - (ii) better searching and discovery of relevant contract terms
  - (iii) improved document assembly systems
  - (iv) easier harvesting of re-usable contract terms from transaction documents
  - (v) collaborative authoring and negotiation
  - (vi) Revision history management
- (b) consistent, automated, enterprise wide formatting of contract documents
- (c) contract documents to be automatically generated in multiple output formats.

#### 5.2.4 Scope assessment

[Assess whether this is within the scope of the TC's specification]

### 5.3 Click through contracts

#### 5.3.1 Reported problems

End users don't know:

- (a) what contracts they have entered into;
- (b) their obligations under those contracts.

The service providers have no obvious interest in helping the buyer solve these problems.

All the systems for accessing the contract are provided by the service provider.

#### 5.3.2 Problem statement

Parties to click through contracts cannot easily identify the contracts they have entered into or the terms of those contracts.

#### 5.3.3 Objectives

The TC specification will enable individuals and enterprises to:

- (a) identify click through contracts prior to assent
- (b) Store contracts assented to
- (c) identify the contracts and terms assented to, when dealing with supporting online service providers.

#### 5.3.4 Scope assessment

[Assess whether this is within the scope of the TC's specification]

### 5.4 Contract management – transaction activities

#### 5.4.1 Reported problems

There are frequent disputes over change authorisation in construction contracts and similar transactions where frequent variations occur.

It is difficult to ensure all parties have reliable information about upcoming obligations under the contract.

It is difficult to extract terms from the contract into content management systems.

It is difficult to reference external documents that are incorporated into the contract.

There is no reliable way to determine the state of contract events, obligations and processes.

It is difficult to monitor and analyse performance of parties over extended time periods.

#### 5.4.2 Problem statement

At the moment parties to contracts and other users of contract information have no way to exchange machine readable contract information for importing into contract management systems.

#### 5.4.3 Objectives

The TC specification will enable:

- (a) parties to contracts to exchange machine readable contract information for importing into contract management systems
- (b) machine readable contract information to be associated with relevant human readable contract terms.

#### 5.4.4 Scope assessment

[Assess whether this is within the scope of the TC's specification]

### 5.5 Contract management – contract document

#### 5.5.1 Reported problems

Interested persons don't have access to the terms of the contract in a convenient forms (paper, RTF, PDF, web, text to speech etc).

It is difficult to ensure that all interested persons have access to fully up to date versions of contract documents.

#### 5.5.2 Problem statement

At the moment, it is not practicable to provide interested persons with contract documents in formats that suit their access needs.

#### 5.5.3 Objectives

The TC specification will enable parties to contracts and authorised agents of the parties to access contract documents in formats convenient to them where the contract document is prepared using XML markup and maintained in an up to date form as amendments are made.

#### 5.5.4 Scope assessment

[Assess whether this is within the scope of the TC's specification]

## 5.6 Electronic commerce contracts

### 5.6.1 Reported problems

Electronic commerce standards such as ebXML provide for only part of the contract. The master agreements or transaction protocol agreements are only available in human readable form.

There is no way for new parties to automatically accede to the master agreement.

### 5.6.2 Problem statement

In electronic commerce transactions:

- (a) there is currently no way to map or validate electronic transactions against their master agreement
- (b) There may be no way to automatically determine if there is a master agreement
- (c) Human negotiation of bi-lateral master agreements is too time consuming.

### 5.6.3 Objectives

The TC specification will enable:

- (a) electronic commerce systems to establish machine readable master contracts that will permit automatic validation of subordinate electronic transactions
- (b) automatic contractual opt-in to participate in the system
- (c) (consider whether the eContracts standard can provide a lingua franca for processing data created from various e commerce standards such as ebXML etc).

### 5.6.4 Scope assessment

[Assess whether this is within the scope of the TC's specification]

## 5.7 Computer negotiated contracts

### 5.7.1 Reported problems

Contract negotiation is slow and expensive.

Human contract negotiation limits the value to the parties, particularly where many parameters are involved.

Currently, there is no way to ensure that both parties to the negotiation can generate identical contract documents from the negotiated parameters.

### 5.7.2 Problem statement

There is no standard way to map negotiated contract parameters to contract terms.

### 5.7.3 Objectives

### 5.7.4 Scope assessment

[Assess whether this is within the scope of the TC's specification]

## 6 Overview of the proposed specification

### Drafting note

This is merely a suggested overview from Peter Meyer. It goes to the heart of the TC's work and the issues raised must be resolved by the TC before the requirements can be developed much further.

The TC aims to develop a specification that can be used by the widest range of users at all stages of the contract life cycle. The specific needs of users and the systems they use are diverse. There is only very limited use of XML for document markup and processing systems are immature. The proposed specification will set out to achieve core, common objectives with minimal prescriptiveness. Feedback from use of the specification will guide its future development.

The specification will define an XML conforming schema for the generic, structural markup of contract documents. This will include a framework to add contract metadata and embedded data values markup to suit particular contract transactions. It will be up to interested industry sectors to define particular vocabularies for metadata and embedded data values markup relevant to those industry sectors.

The generic, structural markup will support all document creation and publishing processes described earlier.

For the foreseeable future, the TC expects that structural markup will be used mainly inside contract authoring enterprises and that there will be little, if any exchange of XML contract documents between parties or other interested persons. Where exchange of XML document does occur, it may be based on industry specific schema or on the Microsoft WordML schema.

The structural markup schema will facilitate the use of XML markup in contract documents that can be exchanged between the parties. It can be adopted to support contract management requirements as the relevant exchange protocols and processing systems are implemented by software developers and the parties.

The initial version of the specification will define normative components only as far as necessary to promote support from software vendors.

The specification will define protocols for the exchange of contracts metadata and embedded data values between the parties and other interested persons for contract management purposes. These protocols make no assumptions about

the use of XML markup for contract documents or the schema used for XML markup of those documents. It will be up to the parties to extract this information from database systems, XML documents or to create the exchange data manually, according to the systems available.

Exchange protocols for metadata and embedded data values will be of limited utility unless implemented by contract management system vendors. The TC does not have any contract management vendor representation. This part of the specification will be non normative.

The specification will define protocols for the specification of contracts semantics markup independently from the generic structural markup of the contract documents and for the exchange of contracts semantics data between interested parties. Again, the TC does not have any contract management vendor representation. This part of the specification will be non normative.

## **7 Functional requirements**

### **7.1 Approach to requirements development**

The following table summarises how the processes discussed in section 3 relate to each relevant contract type. A check mark is shown under a contract type if the process can occur with that contract type, although it may not be common. The points shown by the check marks are the key intersections for which requirements are to be considered.



**Functional processes and contract types**

	Negotiated	Standard form	Click through	E commerce master	Computer negotiated
Processes					
(1) precedent creation, storage & retrieval	✓	✓	✓		✓
(2) document assembly for new drafts	✓	✓	✓		✓
(3) variables substitution in new drafts	✓	✓	✓		✓
(4) custom authoring of contract terms	✓			✓	
(5) collaborative editing of draft contract terms	✓				
(6) publish draft contract documents in multiple output formats	✓	✓	✓		
(7) manage document versions during drafting	✓				
(8) exchange draft contract documents with other parties prior to assent	✓	✓	✓		
(9) produce the assent document	✓	✓	✓		
(10) assent to contract terms	✓	✓	✓	✓	
(11) retain or preserve assent documents	✓	✓	✓	✓	✓
(12) extract contract states, party obligations and rights into exchange format	✓	✓		✓	
(13) communicate contract obligations, rights and states to interested persons	✓	✓			
(14) prepare variations to contracts	✓				
(15) maintain variations and contract versions	✓				

	Negotiated	Standard form	Click through	E commerce master	Computer negotiated
(16) publish contract terms and related information to interested persons	✓				
(17) validate electronic commerce transactions against the transaction protocol agreement				✓	
(18) map negotiated contract parameters to contract terms					✓

**Drafting note**

The TC needs to decide if the listed processes are adequate or within scope.

Requirements will be stated once, in the first process in which they arise. Later processes that would support an existing requirement will refer to the applicable requirements in a note.

## 7.2 Precedent contract documents

### 7.2.1 Overview

There is nothing distinctive about precedent contract documents compared to other precedent documents prepared by lawyers.

The systems used for contract documents should be applicable to other kinds of precedent documents.

### 7.2.2 Precedent creation, storage & retrieval

#### 7.2.2.1 Background to requirements

If precedent documents are to be maintained in XML format, users will need to be able to convert word processing documents into XML markup.

The key problem identified in relation to document precedent documents in section 5.2 is the lack of ability to work with word processing content in various levels of granularity to suit automated processing needs.

Precedents may be created to work with specialised document assembly software. The TC is not attempting to analyse these in depth or to define a standard for the way in which that software operates.

#### 7.2.2.2 Issues for contracts with negotiated terms

Precedent documents are derived from transaction documents and enhanced by subject area experts to provide reliability and flexibility for users. Precedent documents must be regularly updated. New precedent terms are added as transactions evolve and new experiences encountered.

#### 7.2.2.3 Issues for standard form contracts

A standard form contract may be managed as a complete document or as a collection of discrete terms that can be assembled into a contract document to reflect transaction options. This does not appear to raise any issues different from those applicable to precedents for negotiated contracts.

#### 7.2.2.4 Issues for click through contracts

As for standard form contracts.

#### 7.2.2.5 Issues for computer negotiated contracts

It is understood that the precedent terms would need to be accessible in a way that a system could access them by reference or copy them into a contract document instance. Current understanding is that it will be sufficient if contract terms can be stored and retrieved as distinct objects.

#### 7.2.2.6 The role of XML markup

XML markup of precedent documents would provide these benefits:

- (a) It reduces the risk of obsolescence of valuable data as software is updated or replaced.
- (b) It enables content to be managed as convenient components that can be stored, retrieved and shared separately by software systems.
- (c) It enables documents to be automatically transformed into any desired output format according to standard enterprise styles.

There is no evident difference in the needs of the markup for each contract type considered.

#### 7.2.2.7 The role of a standard

A standard would provide these benefits:

- (a) The cost of implementation of XML based authoring and publishing applications could be reduced if applications are developed around a standard schema.
- (b) The costs of switching between document assembly and other processing software applications would be reduced if those applications are developed around a standard schema. However, enterprises may still incur some costs in adapting precedents to suit the more specialised interfaces of document assembly applications if they are not standardised.
- (c) Document assembly systems should be accessible to more enterprises who create contract documents.
- (d) It may be easier to develop and maintain processes to convert word processing data to XML markup if the target is standardised.

#### 7.2.2.8 Specific requirements

- R-1** The specification must include a schema for generic, structural markup of contract documents.
- R-2** The generic, structural markup schema must conform to W3C XML 1.0.  
*[note: We are not specifying here whether the schema is expressed in RELAX NG, XML Schema or DTD syntax.]*
- R-3** The generic, structural markup schema must define common content objects such as paragraphs and clauses that may be processed as distinct objects in document assembly systems.
- R-4** It must be an aim in design of the generic, structural markup schema that the common content objects defined by the schema could be adopted by another standards body responsible for developing a schema applicable to other legal documents that are commonly prepared by the same people using the same systems as contract documents.

**R-5** The generic structural markup schema must provide a model for users to add metadata to contract documents and to distinct content objects defined by the schema. The schema must make provision for common metadata fields required by document management, document assembly and publishing applications such as:

- (a) document identifiers, the author, version and dates;
- (b) the legal subject matter or categorisation of distinct content objects.

The schema must not restrict the metadata that users may add to contract documents or content objects.

The specification will not define a legal classification vocabulary for metadata values. These may be developed as required by industry or regional users.

**R-6** The specification must not require the use of any particular XML processing technology. As far as practicable, it must be designed to allow users to adopt any processing technology of their choosing.

## 7.3 Contract document creation

### 7.3.1 Document assembly for new drafts

#### 7.3.1.1 Background to requirements

There are various approaches to document assembly. Essentially, the process involves a software application presenting a drafter with choices about the terms to be included in a draft contract or other document. As the user signifies his or her choices, the application builds a document to reflect those choices.

The proposed specification will not promote any particular approach to document assembly. Different applications will have particular requirements for markup in the precedent documents. While this may diminish the value of a standard to users, it is unavoidable at this time. If software vendors can work with the proposed generic, structural markup there will be significant advantages to user enterprises who use that schema.

It is theoretically possible that someone could use an XML document as an assent document. The writer is unable to conceive of a situation where this would occur at this time. Consequently, in all contract types, the XML precedent document is transformed into a display format after document assembly.

#### 7.3.1.2 Issues for contracts with negotiated terms

Typically, the document assembly process produces only a first draft contract document. The author must then modify the draft to tailor it to the particular transaction at hand. Since authors work with word processing applications, it is assumed that an XML precedent must be translated to a word processing format for further drafting. If this is to work, the resulting word processing document must be of the same standard as one created from scratch in a word processing application. For example, it must have named styles and automatic numbering.

#### 7.3.1.3 Issues for standard form contracts

In this case, the document produced by a document assembly process will be the final document submitted for assent. It is assumed in this case that there is no negotiation over narrative terms of the contract.

In this case, it is necessary only to translate the XML precedent document into a format suitable for assent, for example print, PDF or HTML.

#### 7.3.1.4 Issues for click through contracts

As for standard form contracts.

#### 7.3.1.5 Issues for computer negotiated contracts

As for standard form contracts.

#### 7.3.1.6 The role of XML markup

While document assembly processes are undertaken without XML, there are many disadvantages, as previously stated. XML provides the benefits described in section 7.2.2.6.

#### 7.3.1.7 The role of a standard

The benefits of a standard are the same as those listed in section 7.2.2.7.

#### 7.3.1.8 Specific requirements

Requirements R-1 to R-6 apply to this process.

**R-7** The generic structural markup schema must provide sufficient definition of content objects in contract documents that user enterprises can:

- (a) define and apply automatic numbering schemes to those objects;
- (b) automatically transform or style contract documents into chosen output formats, including but not limited to print, RTF, PDF, HTML and text to speech ready formats.

**R-8** The generic structural markup schema must enable users to include all content necessary to create a contract document that could be used as an assent document. This does not require that attachments such as plans and other documents created by other persons must be marked up using the schema.

### 7.3.2 Variables substitution in new drafts

#### 7.3.2.1 Background to requirements

Variables substitution involves the insertion of data values into placeholders in the narrative text of a precedent document or contract.

Variables substitution may occur with or as a separate process to document assembly. When highly standardised documents are used, variables substitution with a precedent contract may be the only process needed to prepare a contract document.

#### 7.3.2.2 Issues for contracts with negotiated terms

Typically, the contract author will enter or import data values into a database format suited to the available processing software. Precedent documents may contain named placeholder markers which are replaced by the data values. The software provides a facility for the author or a system administrator to map named placeholders to data values in the database. When the draft document is otherwise complete, the data values are inserted into the document.

This process does not require the use of XML. It is commonly performed with word processing documents using a range of different software applications. The proposed specification is not concerned with word processing based operations.

If precedent documents are maintained in XML format, it is possible that variables substitution may be performed on the XML document or on a word processor version after translation. The proposed specification is not concerned about how or when variables substitution is carried out.

In word processing formats, some software vendors can read placeholder markers from other software vendors. This mitigates the lack of standards but does not reduce the need for costly data conversion from time to time.

There would be advantages to user enterprises if the placeholder markers in XML documents are standardised to provide greater interoperability of precedent documents with software from multiple vendors. It is not clear if this is practicable at this time. Consequently, it is proposed that the specification for markup of placeholders is non normative in the first version.

The placeholders for variables substitution in precedent documents are essentially placeholders for embedded data values. If marked up in the XML, the values can be read out by downstream processing applications.

#### 7.3.2.3 Issues for standard form contracts

The issues are essentially the same as for contracts with negotiated terms.

#### 7.3.2.4 Issues for click through contracts

As for standard form contracts.

#### 7.3.2.5 Issues for computer negotiated contracts

As for standard form contracts.

#### 7.3.2.6 The role of XML markup

XML markup may provide a way to standardise placeholder variables in precedent documents, if supported by software vendors.

XML markup of placeholders in precedents would make it easy to automatically markup embedded data values for later retrieval in downstream processing applications.

#### 7.3.2.7 The role of a standard

A standard would be advantageous, for reasons canvassed earlier. It would be difficult to expect compliance with so few software vendors represented on the TC. It is proposed that this part of specification will be non normative.

#### 7.3.2.8 Specific requirements

Requirements R-1 to R-6 apply to this process.

**R-9** The specification must define one or more elements as placeholders in XML precedent contracts that can be used for variables substitution. Variables substitution includes the insertion of markup to permit the extraction of embedded data values from contract documents.

**R-10** It is necessary to determine how the specification should allow users to define the content that is required to be inserted for each placeholder.

### 7.3.3 Custom authoring of contract terms

#### 7.3.3.1 Background to requirements

#### 7.3.3.2 Issues for contracts with negotiated terms

#### 7.3.3.3 E commerce master

#### 7.3.3.4 The role of XML markup

#### 7.3.3.5 The role of a standard

#### 7.3.3.6 Specific requirements

### **R-11**

### **R-12**

### **R-13**

### 7.3.4 Collaborative editing of draft contract terms

#### 7.3.4.1 Background to requirements



#### 7.3.4.2 Issues for contracts with negotiated terms

#### 7.3.4.3 The role of XML markup

#### 7.3.4.4 The role of a standard

#### 7.3.4.5 Specific requirements

### **R-14**

### **R-15**

### **R-16**

#### 7.3.5 Publish draft contract documents in multiple output formats

##### 7.3.5.1 Background to requirements

##### 7.3.5.2 Issues for contracts with negotiated terms

##### 7.3.5.3 Issues for standard form contracts

##### 7.3.5.4 Issues for click through contracts

##### 7.3.5.5 The role of XML markup

##### 7.3.5.6 The role of a standard

##### 7.3.5.7 Specific requirements

### **R-17**

### **R-18**

### **R-19**

### 7.3.6 Manage document versions during drafting

#### 7.3.6.1 Background to requirements

#### 7.3.6.2 Issues for contracts with negotiated terms

#### 7.3.6.3 The role of XML markup

#### 7.3.6.4 The role of a standard

#### 7.3.6.5 Specific requirements

### **R-20**

### **R-21**

### **R-22**

### 7.3.7 Exchange draft contract documents

#### 7.3.7.1 Background to requirements

#### 7.3.7.2 Issues for contracts with negotiated terms

#### 7.3.7.3 Issues for standard form contracts

#### 7.3.7.4 Issues for click through contracts

#### 7.3.7.5 The role of XML markup

#### 7.3.7.6 The role of a standard

#### 7.3.7.7 Specific requirements

**R-23****R-24****R-25****7.3.8 Produce assent document****7.3.8.1 Background to requirements****7.3.8.2 Issues for contracts with negotiated terms****7.3.8.3 Issues for standard form contracts****7.3.8.4 Issues for click through contracts****7.3.8.5 The role of XML markup****7.3.8.6 The role of a standard****7.3.8.7 Specific requirements****R-26****R-27****R-28****7.4 Assent by parties to the contract****7.4.1 Assent to contract terms****7.4.1.1 Background to requirements****7.4.1.2 Issues for contracts with negotiated terms**

7.4.1.3 Issues for standard form contracts

7.4.1.4 Issues for click through contracts

7.4.1.5 Issues for e commerce master contracts

7.4.1.6 The role of XML markup

7.4.1.7 The role of a standard

7.4.1.8 Specific requirements

## **R-29**

## **R-30**

## **R-31**

## **7.5 Contract management – contract activities**

7.5.1 Extract contract states, obligations, rights etc into exchange format

7.5.1.1 Background to requirements

7.5.1.2 Issues for contracts with negotiated terms

7.5.1.3 Issues for standard form contracts

7.5.1.4 Issues for e commerce master contracts

7.5.1.5 The role of XML markup

7.5.1.6 The role of a standard

#### 7.5.1.7 Specific requirements

### **R-32**

### **R-33**

### **R-34**

## 7.5.2 Communicate contract states, obligations & rights to interested parties

### 7.5.2.1 Background to requirements

#### 7.5.2.2 Issues for contracts with negotiated terms

#### 7.5.2.3 Issues for standard form contracts

#### 7.5.2.4 The role of XML markup

#### 7.5.2.5 The role of a standard

#### 7.5.2.6 Specific requirements

### **R-35**

### **R-36**

### **R-37**

## 7.6 Contract management – contract document

### 7.6.1 Retain or preserve assent document

#### 7.6.1.1 Background to requirements

#### 7.6.1.2 Issues for contracts with negotiated terms

#### 7.6.1.3 Issues for standard form contracts

7.6.1.4 Issues for click through contracts

7.6.1.5 Issues for e commerce master contracts

7.6.1.6 Issues for computer negotiated contracts

7.6.1.7 The role of XML markup

7.6.1.8 The role of a standard

7.6.1.9 Specific requirements

#### **R-38**

#### **R-39**

#### **R-40**

7.6.2 Prepare variations to contract

7.6.2.1 Background to requirements

7.6.2.2 Issues for contracts with negotiated terms

7.6.2.3 The role of XML markup

7.6.2.4 The role of a standard

7.6.2.5 Specific requirements

#### **R-41**

#### **R-42**

#### **R-43**

### 7.6.3 Maintain variations and contract versions

#### 7.6.3.1 Background to requirements

#### 7.6.3.2 Issues for contracts with negotiated terms

#### 7.6.3.3 The role of XML markup

#### 7.6.3.4 The role of a standard

#### 7.6.3.5 Specific requirements

### **R-44**

### **R-45**

### **R-46**

### 7.6.4 Publish contract terms to interested persons

#### 7.6.4.1 Background to requirements

#### 7.6.4.2 Issues for contracts with negotiated terms

#### 7.6.4.3 The role of XML markup

#### 7.6.4.4 The role of a standard

#### 7.6.4.5 Specific requirements

### **R-47**

### **R-48**

### **R-49**

## 7.7 Other processes

### 7.7.1 Map negotiated parameters to contract terms

#### 7.7.1.1 Background to requirements

#### 7.7.1.2 Issues for computer negotiated contracts

#### 7.7.1.3 The role of XML markup

#### 7.7.1.4 The role of a standard

#### 7.7.1.5 Specific requirements

### **R-50**

### **R-51**

### **R-52**