

# The Contract Expression Language – CEL

Xin Wang

ContentGuard, Inc.  
222 N. Sepulveda Blvd., Suite 1400  
El Segundo, CA 90245, USA  
[xin.wang@contentguard.com](mailto:xin.wang@contentguard.com)

Eddie Chen

ContentGuard, Inc.  
222 N. Sepulveda Blvd., Suite 1400  
El Segundo, CA 90245, USA  
[eddie.chen@contentguard.com](mailto:eddie.chen@contentguard.com)

Dmitry Radbel

Universal Music Group, eLabs Division  
2220 Colorado Ave.  
Santa Monica, CA 90404, USA  
[dmitry.radbel@umusic.com](mailto:dmitry.radbel@umusic.com)

Tsutomu Horioka

Nippon Telegraph and Telephone Corp.  
1-1 Hikarinooka Yokosuka-Shi  
Kanagawa 239-0847 Japan  
[horioka.tsutomu@lab.ntt.co.jp](mailto:horioka.tsutomu@lab.ntt.co.jp)

Jim Clark

Microsoft Corp.  
One Microsoft Way  
Redmond, WA 98052, USA  
[jimclark@microsoft.com](mailto:jimclark@microsoft.com)

Greg Wiley

Banjo Creek  
1534 N. Moorpark Rd. Suite 250  
Thousand Oaks, CA 91360, USA  
[greg@orthogony.com](mailto:greg@orthogony.com)

**Abstract** — This paper describes the Contract Expression Language (CEL), currently being developed at the industry consortium Content Reference Forum. The CEL is an XML-based language designed to express contractual agreements between different parties for the purposes of capturing and communicating contractual information, and facilitating contract execution and enforcement by machines with respect to granted permissions, mandated obligations and stipulated prohibitions. In addition to modeling contractual agreements using the deontic concepts of rights, obligations and prohibitions, it has distinct features for specifying statements of intentional, factual and exclusive types, defining preference rules for resolving conflicts, and supporting lifecycles and trust management of contracts in open and distributed environments. This paper presents an overview of the CEL in terms of its data model, expressiveness and processing models, and illustrates its application in the Content Reference Framework for content distribution, and its compliance to the Business Collaboration Framework for e-business transactions.

**Keywords** - *contract expression language, lifecycle, trust management, content references, business collaboration framework, standardization.*

## I. INTRODUCTION

This paper describes the Contract Expression Language (CEL) [4], an XML-based language designed to express contractual agreements between different parties.

The CEL is currently under development within the Content Reference Forum (CRF) [6], and is based on the ISO/IEC MPEG standard Rights Expression Language [19]. The CRF is an industry consortium to develop specifications for a content distribution framework. The CEL is designed to specify contractual agreements that resemble those of actual contracts (rather than logic rule sets found in most business rules languages, e.g., [21]) used in content distribution and to automate processing of these contractual agreements as contained in CEL expressions. In particular, CEL expressions are used to ensure interoperable, consistent and predictable

behaviors of distribution value-chain participants (e.g., content owners, packaging providers, distributors, retailers and consumers) in terms of their rights, obligations and prohibitions and to express the declaration of distribution constraints and applicable regulations. A specific usage of the CEL is to regulate generation of standard licenses which grant usage rights to content [8].

In order to validate applicability, to promote adoption and to facilitate interoperability, a large set of requirements has been collected for the CEL [5]. The primary function of CEL expressions is to serve the following purposes: 1) evidence: to communicate information conveyed within a contract that can be easily understood and unambiguously interpreted; 2) execution: to facilitate permissive, obligatory or prohibitory performance within a contract in the appropriate context, integrated with the contracting parties' business processes. This includes determining whether one is allowed to exercise some rights, or is required to fulfill some obligations or obey some prohibitions; and 3) evaluation: to check permissive, obligatory, or prohibitory performance by contracting parties. The secondary function of CEL expressions is to facilitate: 1) formation of contracts and automate the conventional process of offer, acceptance, agreement, and consideration; 2) dynamically generated and updated contracts based on, for instance, negotiation and future business models; and 3) enforcement of contracts with respect to their granted permissions, mandated obligations, and stipulated prohibitions.

This paper presents an overview of the CEL in terms of its data model (Section II), expressiveness (Section III) and processing models (Section IV), and illustrates its application to the Content Reference Framework for content distribution, and its compliance to the Business Collaboration Framework for e-business transactions (Section V).

## II. OVERVIEW OF CEL

The CEL is a general purpose language that can be used to express contractual agreements in an unambiguous, machine-interpretable form. It is designed to provide an extensible

architectural framework. It also includes a set of baseline common and application-specific vocabularies for specifying contracts in many potential applications. The role of contracts written in the CEL is to describe the business agreements (generally contractual) between or among parties participating in the business value chain in a way that, when the contracts are specified, the interests and obligations of the value chain participants can be communicated, interpreted, and executed by machines.

### A. Basic Concepts

What is a contract? One definition is that a contract is “a promise, or set of promises, for the breach of which the law gives a remedy, or the performance of which the law in some way recognizes as a duty.[3]” Two or more parties can draft a contract to declare their consent to any act or thing to be done or forbome by some of them or others. For instance, “Alice must sell her house to Bob upon receiving a payment of \$500,000 from him by April 30, 2002, signed by both Alice and Bob” constitutes a contract.

In the CEL, a contract is considered as an agreement between two or more parties over a number of promises made by some parties. The nature of being an agreement requires that a valid contract be signed by all parties involved. Each promise contains several clauses, each of which states a relationship as follows:

- some parties possess a right (or permission) to what the other grants,
- some parties bind an obligation to what the other requires,
- some parties follow a prohibition to what the other imposes,
- some parties see an intention to what the other expresses, or
- some parties know an assertion to what the other makes.

A right (or permission), an obligation or a prohibition means performance or non-performance of some act. The difference among them is on the types of modality of the act they address [14][15][24][25]. A right or permission is of type of modality characterized by the word “may”, an obligation categorized by the word “must”, and a prohibition by “must not”. In contrast, an intention means a desire to perform some act, and an assertion describes a fact about the state of affair as to whether or not some act was performed, is being performed, or will be performed.

The relationship stated in a clause may be contingent upon some event to occur or some condition to be true. An event captures the changes in a system context or environment that may require the examination or execution of the clause. For instance, an obligation clause can be triggered when an event occurs, and when this is the case, the parties must fulfill the obligation. Event driven processing makes it possible to build more efficient systems to manage (e.g., consult, execute and enforce) contracts.

Clauses may be also mutually dependent. Simple kinds of dependency are the ones based on the existence and validity of another. More complex ones are the ones based on performance or non-performance of others. These kinds of dependency can be used to model rewards for fulfilling contracts and remedies for breaching contracts.

It is also important that a contract is modeled as a dynamic object with its lifecycle, and operations of creating, negotiating, executing and revoking contracts themselves need also be managed in a trusted manner.

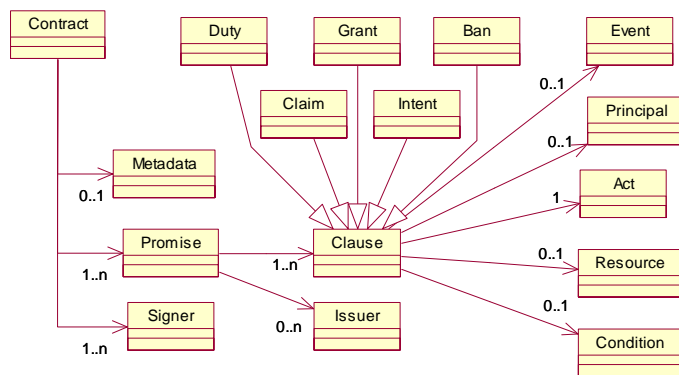
### B. Approach

The CEL uses the XML technology to define its syntax and semantics. It is based on the MPEG REL [19]. It uses relevant concepts, elements and types defined in the REL as building blocks. It also adopts REL’s extensibility mechanisms provided by the XML schema [26], which, for instance, enables the separation of the CEL core elements and their relationships specified in the form of its data model from the application specific vocabularies captured in the forms of a variety of extensions to the core. This kind of design consistency ensures interoperability with the MPEG REL.

Like the MPEG REL, the CEL is defined as a declarative language. What this means is that the semantics of expressions written in CEL has no dependency on procedural or control aspects of interpreting the expressions, and the interpretation of CEL expressions yields no side effects – that is, the state of a system that uses CEL does not change because of the evaluation of any CEL expression.

### C. Data Model

The following diagram illustrates the CEL data model:



In this model, the root element of a CEL expression is the Contract element. It contains one or more Promise elements and one or more Signers elements representing parties who pledge to abide by the statements made by the Promise elements in the Contract. Each Promise is further defined as a conglomeration of Clauses and may contain one or more Issuers, each of which issues and optionally signs the statements conveyed by the Clauses within the Promise. An Issuer is different from a Signer in that the former makes the statement in the Clause it issues, whereas the latter agrees to that statement. Each of the Clauses may contain Event, Principal, Act, Resource, and Condition elements, but all these

elements are optional except for Act. The general framework of the CEL is based on this EPARC model of a Clause. More specifically, “EPARC” represents the following key components:

1. Event: an optional element representing a triggering event.
2. Principal: an optional element representing a party or a set of parties that may perform the act specified in the peer element Act.
3. Act: a required element representing an act or a set of acts.
4. Resource: an optional element representing a resource or a set of resources to which the Act applies.
5. Condition: an optional element representing a condition, subject to which the Act may be performed.

Currently CEL has provided five specific types of Clauses that are syntactic and semantic extensions to the Clause: Grant, Duty, Ban, Intent and Claim. Their semantics are as follows:

1. Grant: whenever the Event is triggered, permits the Principal the right to perform the specified Act over the Resource provided the Condition is met.
2. Duty: whenever the Event is triggered, demands an obligation on the Principal to perform the Act on the Resource when the Condition is met.
3. Ban: whenever the Event is triggered, imposes a prohibition on the Principal to perform the Act on the Resource under the Condition.
4. Intent: whenever the Event is triggered, expresses the Principal’s intent to perform the Act on the Resource under the Condition.
5. Claim: whenever the Event is triggered, asserts a fact that the Principal has performed the Act on the Resource when the Condition was met.

These specific Clauses make individual contractual statements to convey permissive, obligatory, prohibitory, intentional, and assertive information, and that, when grouped into Promises made by their Issuers, comprise a CEL Contract agreed upon by its Signers.

*D. A Simple Contract Example*

A simple goods-for-sale example is stated as the following: “Alice and Bob agree on November 11, 2001 that Alice must sell her house to Bob if he pays her \$500,000.” The CEL expression below specifies this agreement between Alice and Bob (signed on 11/11/2001) that imposes an obligation on Alice to sell the property to Bob when she receives \$500,000 from Bob. The Duty Clause contains the Event element “receivePayment”, the Principal elements “keyHolder” for Alice and Bob, the Act element “sell” with the constraint “sellTo”, and the Resource element “property”.

```

<paymentFlat>
  <rate currencyCode="USD">500000</rate>
</paymentFlat>
<payer>
  <keyHolder licensePartId="Bob"/>
</payer>
<payee>
  <keyHolder licensePartId="Alice"/>
</payee>
</receivePayment>
<keyHolder licensePartId="Alice"/>
<sell>
  <sellTo>
    <keyHolder licensePartId="Bob"/>
  </sellTo>
</sell>
<property licensePartId="AliceHouse">
  <location>
    <address>SOME ADDRESS</address>
  </location>
</property>
</duty>
</promise>
<signer licensePartId="Alice">
  <Signature/>
  <details>
    <timeOfIssue>2001-11-11T11:11:11</timeOfIssue>
  </details>
</signer>
<signer licensePartId="Bob">
  <Signature/>
  <details>
    <timeOfIssue>2001-11-11T11:11:11</timeOfIssue>
  </details>
</signer>
</contract>

```

III. EXPRESSIVENESS OF CEL

In order to meet the requirements in [5], the CEL provides a number of noticeable features that make it very expressive, especially in terms of interdependence of Clauses, exclusivity, preference for resolving conflict and multiplicity, trust and binding relationship, and support for contract lifecycles.

A. Interdependence of Clauses

Contract clauses may be interdependent. The following are some forms of the interdependence of Clauses that the CEL supports through a number of predefined Conditions:

1) Existence. Validity of Clauses may simply depend on existence of other Clauses. For instance, a Claim for a fact that a person was an employee of some company is dependent on the existence of another Claim that the employee is included an employment record issued by the company. The Condition, ExistClause, can be used for specifying this kind of dependence.

2) Validity. Validity of Clauses may be dependent on validity of other Clauses. For instance, a Grant for a right to view an R-rated movie is dependent on a Claim for an assertion that the Principal is at least 18 years old. The Condition, ValidClause, can be used for specifying this kind of dependence. Using this Condition, logical implication can be specified in particular between Claims when they are considered as logical propositions.

3) Performance and non-performance. In order to specify remedy and warranty, one need conditions to specify whether or not an obligation or prohibition has been fulfilled in the past. If not, some other obligations, prohibitions and/or rights may

```

<contract>
  <promise>
    <duty>
      <receivePayment>

```

incur. The Conditions, ExecutedClause and NotExecutedClause, can be used for this purpose.

### B. Exclusivity

It is often desirable that a contract clause can be specified in a mutually exclusive way that does not allow anything else to happen. For instance, someone has an exclusive right to publish a digital work for a specific territory, or a guest (non-registered) user can only browse the public area of a website.

As a CEL Clause is modeled using the EPARC model and it involves an Event, a Principal, an Act, a Resource and a Condition, the exclusivity restriction may vary from one component element to another. For instance, the exclusive statement, "Lycos is the only authorized seller of 'I Remember Mama' in Australia", grants that Lycos sells "I Remember Mama" in Australia, and it also forbids anyone else (other than Lycos) to sell it in Australia. This is somehow different from the exclusive statement, "Lycos is the authorized seller of only 'I Remember Mama' in Australia", which implies that Lycos can sell only "I Remember Mama" (but not others) in Australia.

Thus, each component element of a Clause is augmented with an optional attribute of Boolean type, Exclusivity, to indicate whether or not the element is to be exclusive. By default, the attribute takes the "False" value. Only when it is specified with the attribute Exclusivity being "True", the element is meant to be exclusive. Semantically, this attribute is a "syntactic sugar" for convenience. This is because one can perform a "normalization" transformation to convert exclusive Clauses into non-exclusive Clauses. For instance, the first example above can be converted into a Grant for Lycos and a Ban for all others.

### C. Preference for Conflict and Multiplicity

With the expressiveness of prohibitions and exclusivity, it becomes possible to have potential conflicts in expressed contracts. For instance, one contract may require an obligation on Alice to do something and another contract may demand a prohibition on Alice not to do the same thing. Moreover, sometimes when multiple Clauses of the same type (say, "Grant") are applicable from a same Contract or different Contracts, it is also desirable to determine which one of them is preferable to the others.

The real issue with conflict and multiplicity resolution is how to identify potential conflict and multiplicity and resolve them according to some policies specified in some (possibly higher level) contracts. Note that conflicts and multiplicities may recursively occur at the higher level in the cases where there may be multiple higher contracts.

In order to resolve potential conflicts and multiplicities, some kind of preference relationship needs to be established among clauses. The CEL provides a special Act, Precede, which can be used to specify that one Clause precedes another in terms of preference. With this Act, one can make Claims (such as Clause A "Precede"s Clause B) for specifying any partial order on preference among Clauses in order to resolve potential conflicts and multiplicities.

### D. Trust and Binding Relationship

Applications such as digital rights management, e-commerce and other distributed services require interaction between parties that do not know each other. In such situations, the traditional assumptions for establishing and enforcing contractual agreements do not hold. Not only that principals of interactions may be previously unknown to each other and resources to be controlled may be moving from one system to another (e.g., a movie is downloaded from a distributor site to a user PC and its access on the PC needs also to be controlled), but more importantly the behavior of some principals may be regulated by contracts imposed by other principals (e.g., usage rights to consumers are granted by content owners or distributors). When a contract is to be executed, understanding what to trust and what is bound within the contract is important. For instance, an issuer I grants a principal P a right to play a movie. The principal P needs to establish some level of trust with the issuer I to ensure that I is a legitimate authority to grant P the right. In another case, two householders sign a contract and require their respective family members to bind to the contract. A binding relation needs to be specified in order to determine who are bound to the signers and hence also responsible for the content of the contract.

In CEL, these trust and binding relationships can be specified using Claims with the special Acts "Trust" and "Bind", the former specifying that a Principal trusts a Resource (which can also be another Principal) and the latter specifying that a Principal binds to a Resource (which can also be another Principal). Using these Claims, one is able to define a trust policy or binding policy.

Moreover, the signatures provided by an Issuer of a Promise and a Signer of a Contract facilitate effective trust management. For instance, the signature of an Issuer indicates that it is the Issuer or Signer of the signature who issues the statements conveyed by the Clauses within the Promise. This signature serves as an evidence for a decision that a user of the Clauses needs to make as to whether or not the user trusts the signatures to believe the content and its authenticity carried in the Clauses. The Issuer will also help conducting the issuance path validation for a chain of Clauses in order to establish trust on the Issuer of the end Clause based on trust on the Issuer of the head (or root) Clause in the chain. Similarly, other than providing contract integrity and signer authentication, the signatures of Signers of a Contract convey the consent of these Signers.

### E. Support for Contract Lifecycles

A lifecycle of a contract usually consists of the following phases: creation (including offering, consideration and negotiation), execution (dictating involved parties to exercise rights, fulfill obligations, obey prohibitions, contest intentions and verify assertions), and post management (including amendment, extension, renewal, completion, revocation, termination, expiration, and archiving).

To support lifecycles of contracts, the CEL provides a number of special Acts as meta-acts as they are acts upon Clauses (as well as Promises and Contracts):

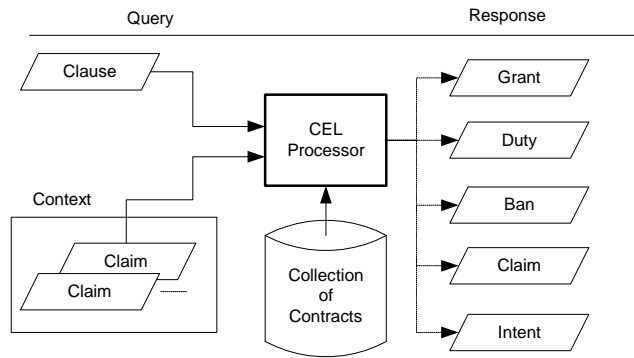
- Issue: used to specify that a Principal issues a Clause (within a contract). With this Act, one can create a specified Clause and be an Issuer of the Clauses.
- Obtain: used to specify that a Principal obtains a Clause (within a contract). With this Act, one can receive a specified clause.
- Revoke: used to specify that one revokes a Clause by revoking one of its signatures as a Signer or Issuer. With this act, one can invalidate an existing Clause made by the Signer or Issuer.
- Possess: used to specify that a Principal can possess some Resource. This is very useful for specifying Claims of the attribute-certificate type, and using such certificate Claims as dependent Claims for others.

#### IV. PROCESSING MODELS OF CEL

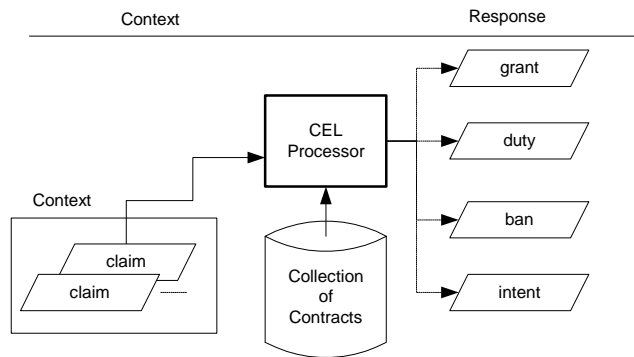
The data model and semantics of the CEL supply a logical foundation for a number of processing models for CEL expressions. They can be broadly categorized into query-driven, context-driven and conflict-driven, corresponding to the backward inference, forward inference and conflict resolution based on the “intuitive” logic semantics of CEL expressions, and any hybrid combinations of these models.

Fundamental to all of these processing models is the concept of context. A context in the CEL is a collection of propositions that describe a state of affair. Semantically, it can be considered as a collection of propositions generated by a collection of CEL Claims that are assumed to be valid. The function of a context is to provide a set of factual statements that can be used to validate conditions related to Events and Conditions, to identify and match Principals, Acts and Resources, and to trust Issuers and Signers. For instance, a context may contains propositions to declare that Principal P possesses a role of “administrator” (issued by an owner as an Issuer) , trusts whatever Issuer I issues (for I being a trusted authority) , and binds to whatever Signer S signs (for S being a legal representative of P).

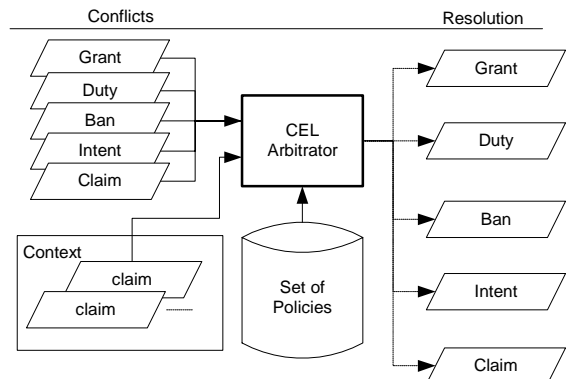
In the query-driven processing model as shown in the diagram below, a collection of contracts serves as a knowledge base, and is consulted when a query for an action or a fact is submitted. In this case, a query in the form of a Clause and a context in the form of a list of Claims are given as inputs, and Clauses (together their signers) which match the query and whose Event and Condition components are simplified based on the information provided in the context are the expected response to the query. Using this query-driven model, sophisticated processing tasks can be built, according to the modality of the special clauses. For instance, when a query is an Intent, the query processing result could be a set of Grants, Duties, Bans and Claims that match the Intent. Similarly, a collection of Grants, Duties and Bans can be generated as a response to a Grant query.



In the context-driven processing model as shown in the diagram below, a collection of contracts is interpreted according to a given context. This is typically triggered when an event occurs that changes the context. In this case, a context in the form of a list of Claims is given, and Clauses (together with their Signers) whose Event and Condition components are satisfied in the context are delivered as the output statements held in the context. Using this context-driven model, applications can be developed to notify valid permissions, obligations, prohibitions and intentions, within any given context, in order to perform and execute a set of contracts.



Finally, in a conflict-driven processing model for a collection of contracts, one conflict set in the form of a collection of Clauses (potentially over common Principals, Acts and Resources – hence conflict) and a context in the form of a list of a list of Claims are given as inputs, and a minimal subset of Clauses are chosen according to some preference or conflict resolution rules/policies defined in the contracts. The output Clauses are the resolution of the conflicts.



## V. APPLICATIONS OF CEL

### A. Content Reference Framework for Content Distribution

The Content Reference Forum (CRF) [5] is an open consortium of companies from diverse industries chartered to facilitate the distribution of legitimate content governed by the real world of contractual arrangements advantaged in the peer-to-peer distribution environments.

The CRF defines a content distribution framework built on the notion of content references (CRs) [8][9]. A CR is a data package that uniquely identifies content and the context in which was distributed and will be used, and it is resolved by a "Reference Service" (RS) that determine the right content for user context (including rendering environment, language and location) and commercial terms of usage. The RS facilitate the seamless acquisition of appropriate content (e.g., matching consumer's preferences and platform capabilities) by providing an offer or offers for the consumer to acquire the content, or redirect the consumer to the appropriate retail source, per contractual agreements for content distribution.

The CEL is used in the Content Reference Framework for specifying these contractual agreements that the RS needs to interpret and fulfill in order to determine an appropriate action (e.g., a list of appropriate offers) in response to a consumer request. A simple agreement is that the RS has the obligation to issue an offer (expressed in the MPEG REL) if a CR is received (as a triggering event) and the CR matches certain criteria. More complex examples for distribution contracts, marketing initiatives, legal and operational constraints placed on content managed by the RS can be found in [9]. CEL expressions are used to automate the determination of the appropriate offerings or other actions. The CEL also enables dynamic commerce and accommodates many existing and foreseen business models.

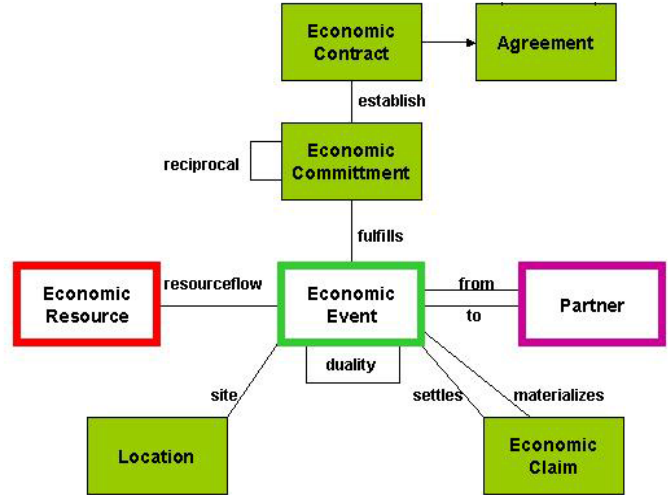
### B. Business Collaboration Framework for e-Business Transactions

The Business Collaboration Framework (BCF) [2] is an international effort to enable business process and information models to be specified in a technology- and implementation-neutral manner that can then be implemented in software using the information exchange syntax and structures of choice. The BCF consists of a collection of specifications defining electronic business exchange for two or more business partners. Because of the common interests in the technologies to describe enforceable e-Business contractual agreements, an analysis on the compliance of the CEL with the BCF is commissioned by the CRF. The result of the analysis is provided as a reference in [7]. It turns out that the CEL is a compliant technology for the BCF within the scope of e-contract expression and interpretation. The following are the evidences to support that claim:

1. the CEL contract data model is mapped to the agreement portion of the BCF meta model,
2. the CEL core elements are aligned to the BCF transaction ontology,

3. it is illustrated how the CEL can be used to specify BCF example contracts,
4. it is demonstrated how the CEL can be used to model BCF business collaborations,
5. it is shown how types of CEL expression templates can be designed to specify BCF business transaction patterns, and
6. it is explained how CEL extensions and profiles can support different BCF views.

For instance, in the BCF, a commercial trading agreement is modeled, as shown in the following diagram, as part of the business collaboration model.



This diagram shows that a business collaboration involves two dual types of Economic Events, each of which details the Economic Resources involved in an exchange between two Partners. A Commitment is a promise by a Partner to initiate an Economic Event in the future. Performing the Economic Events fulfills that Commitment. Commitments should always be reciprocated by the other Partner who commits to initiate another type of Economic Event in return. An Economic Contract is a bundle of reciprocal commitments between Partners. A contract is a subtype of the more general object class called Agreement. The two other objects, Economic Claims (for partially completely exchanges) and Locations (identifying places where Economic Events take place), are not essential to the model and can be considered as attributes of the Economic Event.

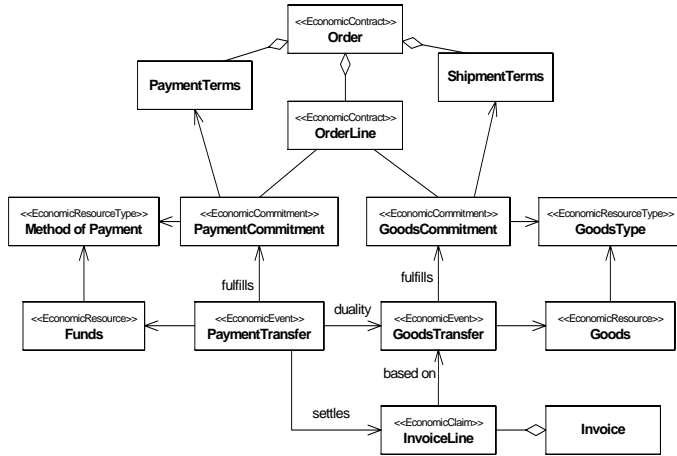
Based on the CEL data model, the following table lists a mapping between the CEL contract elements and the BCF agreement elements.

| CEL Element     | BCF Element                        | Comments  |
|-----------------|------------------------------------|---|
| <i>Contract</i> | <i>Agreement/Economic Contract</i> | A CEL Contract contains elements for specifying agreements involving permission, obligation, prohibition, intention and assertion |
| <i>Contract</i> | <i>Contract</i>                    |   |
| <i>Promise</i>  |                                    | A Promise may specify a Business Collaboration  |

|      |           |                     |   |
|------|-----------|---------------------|---|
| Duty |           | Economic Commitment | A Duty specifies a Commitment             |
|      | Event     |                     | An Event provides triggering a dependency |
|      | Principal |                     | Partner                                   |
|      | Act       |                     | An Event can be converted into Act        |
|      | Resource  |                     | Resource                                  |
|      | Condition |                     |   |

Note that the duality of the reciprocal events can be specified in CEL with two mutually dependent Duty Clauses.

A BCF example of an economic contract is given as follows:



In this diagram, there are two Economic Events, GoodsTransfer and PaymentTransfer, which represent transfers of ownership or control of Economic Resources (Goods and Funds) from one Partner to another (say, Buyer and Seller). The GoodsTransfer Event fulfills the GoodsCommitment, which is one of a pair of reciprocal commitments, and the PaymentTransfer Event fulfills the PaymentCommitment, which is the reciprocal of the GoodsCommitment.

Based on the data model mapping, a number of CEL Contracts can be formed that consists of a Promise that contains two “reciprocal” Duty Clauses, depending on the relative order of the Acts of ship and pay. One possible scenario is that the shipment of goods happens after receipt of the payment. The following CEL expression specifies the ship-after-pay scenario.

```

<contract>
  <promise>
    <duty>
      <buyer/>
      <pay>
        <methodOfPayment/>
        <paymentTerms/>
      </pay>
      <funds/>
    </duty>
    <duty>
      <paymentTransferredEvent/>
      <seller/>
      <ship>
        <shipmentTerms/>
      </ship>
      <goods/>
    </duty>
  </promise>
</contract>

```

```

</promise>
<signer licensePartId="buyer"/>
<signer licensePartId="seller"/>
</contract>

```

## VI. RELATED WORK

### A. MPEG REL

The MPEG REL [19] is an ISO/IEC international standard for a rights expression language. The REL is an XML-based language for specifying declarative statements about usage rights and their associated conditions that are offered and granted by rights holders of digital content, resources, and services in various forms. A rights statement usually declares that someone MAY exercise a specified right upon some specified resource subject to certain conditions. Though many obligations can be expressed as conditions for exercising rights, it does not directly convey the notion that someone MUST or MUST NOT exercise that right when the condition is met.

The CEL leverages heavily from design principles and extensibility structures of the MPEG REL, and extends it to include elements and types for obligations, prohibitions, intentions and assertions. Integrated with the MPEG REL, the CEL provides flexible, interoperable mechanisms to support many forms of contracts including the business contracts for the supply, distribution, and consumption of content, resources, and services.

### B. OASIS LegalXML eContracts

The purpose of the OASIS LegalXML eContracts Technical Committee (TC) [20] is “to develop open XML standards for the markup of contract documents to enable the efficient creation, maintenance, management, exchange and publication of contract documents and contract terms”. The TC emphasizes on the presentation of contract documents, not on how the agreements expressed by these documents can be used to govern the behaviors of parties involved in the contracts. Whereas the CEL is to focus on the contract semantics, the two efforts are complementary.

OASIS also hosts some of the ebXML Technical Committees, such as Collaborative Protocol Profile and Agreement (CPP/CPA) TC [21]. At the present, CPP/CPA is focused on describing the technical capabilities required for electronic collaboration, not on specifying contracts.

### C. Deontic Logic, Rule, Policy and Event based Approaches to Specifying Contracts

There has been a large body of literature on using deontic logic, policies, rules and event calculus to specify contracts (e.g., [11][12][13][16][17][18][22][24][25]).

A distinction between the CEL and the deontic logic based languages is at the granularity of a permissive or obligatory statement. The CEL Clause concerns with its components, especially Principal, Act and Resource, whereas such statements in deontic logic are general propositions. Other distinctions are the support for lifecycles and trust management of contracts provided by the CEL.

Policy and rule specification languages often provide mechanisms to specify authorization and obligation policies. In general, policies are rules governing the choices in behaviors of a system [10]. They are often used as a means of implementing flexible and adaptive systems for management of internet services, distributed systems, and security systems. In this sense, the CEL can also be considered as a policy specification language. What makes the CEL different from these policy languages are mainly the support for interdependence of Clauses, lifecycles, and trust management of contracts.

Finally, events as broadly understood are things that happen. The event-based approach has its limitation because it is mainly descriptive rather than prescriptive. For instance, a BCF Event specifies a process of changing states that, when it happens, fulfills a Commitment. The CEL Act identifies an act, whose performance generates an Event. Thus, the CEL emphasizes an Act, while the BCF emphasizes an Event as the performance of an Act. In comparison to software programming, specifying Act is like writing a program, while specifying an Event is like describing behavior of a program in terms of processes that are the execution instances of the program.

## VII. CONCLUSION

The CEL is a machine interpretable language for expressing contractual agreements with precise semantics, rich expressiveness, solid logic foundation, flexible extensibility structure and sound processing models. It has been adopted and used the Content Reference Forum for content distribution. It has also been assessed as a BCF compliant technology.

For the latest development of the CEL and the latest information on the Content Reference Forum, visit the CRF Web site at [www.crforum.org](http://www.crforum.org).

## REFERENCES

- [1] L. Brownston, R. Farell, E. Kant, and N. Martin, Programming Expert Systems in OPS5: An Introduction to Rule-Based Programming. Addison-Wesley, 1985.
- [2] The Business Collaboration Framework, [www.unbcf.org](http://www.unbcf.org).
- [3] J. D. Calamari and J. M. Perillo, *Contracts*, 3rd edition, Black Letter Series, West Group, St. Paul, MN, 1999.
- [4] CEL: Contract Expression Language, revision 1, May 2, 2003. [http://www.crforum.org/candidate/CELVG002\\_celspec.pdf](http://www.crforum.org/candidate/CELVG002_celspec.pdf).
- [5] CEL Requirements, version 0.93, June 30, 2003. [http://www.crforum.org/candidate/CELVG001\\_celreq.pdf](http://www.crforum.org/candidate/CELVG001_celreq.pdf).
- [6] Content Reference Forum, <http://www.crforum.org/>.
- [7] Content Reference Forum, "Contract Expression Language: An UN/CEFACT BCF Compliant Technology", January 21, 2004. <http://www.crforum.org/papers/CEL-BCF-Whitepaper.pdf>.
- [8] Content Reference Forum, "Overview", March 17, 2003. [http://www.crforum.org/crfreppub/CRF004\\_002\\_cr\\_forum\\_overview.pdf](http://www.crforum.org/crfreppub/CRF004_002_cr_forum_overview.pdf).
- [9] Content Reference Forum, "Baseline Profile Specification 1.0", candidate specifications, December 2003. [http://www.crforum.org/candidate/BP10\\_baseline\\_profile.zip](http://www.crforum.org/candidate/BP10_baseline_profile.zip).
- [10] N. Damianou, A. Bandara, M. Sloman, and E. Lupu, "A Survey of Policy Specification Approaches", April 2002, <http://www.doc.ic.ac.uk/~mss/Papers/PolicySurvey.pdf>.
- [11] A.K. Daskalopulu, and M.J. Sergot, "A Constraint-Driven System for Contract Assembly". Proc. Fifth International Conference on AI and Law, Univ. of Maryland, May 1995. ACM Press, pp 62--70.
- [12] A. Daskalopulu and M.J. Sergot, "The Representation of Legal Contracts", AI and Society 11 (Nos. 1/2), pp. 6-17. 1997.
- [13] G. Geerts and W.E. McCarthy, "An ontological analysis of the economic primitives of the extended-REA enterprise information architecture". International Journal of Accounting Information Systems, (3):1-16, 2002.
- [14] W.N. Hohfeld, Fundamental Legal Conceptions as Applied in Judicial Reasoning. Greenwood Press Publishers, 1978.
- [15] W. N. Hohfeld, Some Fundamental Legal Conceptions as Applied in Judicial Reasoning. Yale Law Journal. 23. 1913.
- [16] S. Jajodia, P. Samarati, and V.S. Subrahmanian, "A logical language for expressing authorizations", in the proceedings of the IEEE Symposium on Security and Privacy, Oakland, CA, USA, pp.31-42, 1997.
- [17] L. Kagal, T. Finin and A. Joshi, "A Policy Language for Pervasive Systems", Fourth IEEE International Workshop on Policies for Distributed Systems and Networks, Lake Como, 4-6 June, 2003. <http://umbc.edu/~finin/papers/policy03.pdf>
- [18] W.E. McCarthy, "The REA accounting model: A generalized framework for accounting systems in a shared data environment". The Accounting Review, LVII(3):554-578, July 1982.
- [19] MPEG REL, ISO/IEC 21000-5:2003, Information Technology – Multimedia Framework – Part 5: Rights Expression Language. <http://www.iso.ch/iso/en/CombinedQueryResult.CombinedQueryResult?queryString=21000-5>.
- [20] OASIS LegalXML eContracts TC, [http://www.oasis-open.org/committees/tc\\_home.php?wg\\_abbrev=legalxml-econtracts](http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=legalxml-econtracts).
- [21] OASIS ebXML Collaboration Protocol Profile and Agreement TC [http://www.oasis-open.org/committees/tc\\_home.php?wg\\_abbrev=ebxml-cppa](http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=ebxml-cppa).
- [22] Rule Markup Language, <http://www.ruleml.org/>.
- [23] J. Widom and S. Ceri, *Active Database Systems*, Morgan-Kaufmann, 1995.
- [24] G.H. von Wright, "Deontic Logic", Mind v60(237):1-15, 1951.W3C XML Schema, <http://www.w3.org/2001/XMLSchema>.
- [25] R.J. Wieringa and J.-J. Ch. Meyer, "Applications of Deontic Logic in Computer Science: A Concise Overview". In J.J.Ch. Meyer and R.J. Wieringa, editors, Deontic Logic in Computer Science, chapter 2, pages 17-40. John Wiley & Sons, 1993.
- [26] W3C XML Schema, <http://www.w3.org/2001/XMLSchema>.