On Formalization and Reasoning Algorithm in Distributed XBRL System

Ding Pan, Yingmin Zhang and Dong Wang

School of Management, Jinan University, Guangzhou, 510632, China
School of Business Administration, Guangzhou University, Guangzhou, 510006, China

Abstract: Traditional description logic usually used to represent static knowledge. To solve the problem of metadata interoperability in distributed XBRL, the distributed interact model of XBRL was constructed, the distributed characteristics and formal representation requirement of distributed XBRL interact model was analyzed. A distributed temporal description logic DTDL_{on} and its syntax and semantics was proposed. Finally, the reasoning mapping algorithm was given.

Key words: Distributed XBRL, formalization, reasoning algorithm, distributed temporal description logic.

INTRODUCTION

XBRL (eXtensible Business Reporting Language) is an XML-based markup language designed specifically for financial reporting, disclosure and use. By giving financial accounting data specific category tags, financial reports can be recognized and understood by computer and through the built-in authentication mechanism, reports can be processed and analyzed automatically. XBRL has advantages in cross-platform using, multiple output formats supporting, faster and more accurate searching. It provides a convenience in financial reports' preparation, generation, analyzing, transmission and comparing and enhances the accuracy and reliability of financial data.

Recent work on XBRL is mainly focus on the regulatory of Specification and Taxonomy and the disclosure of XBRL reports, while there is rarely improvement in data intelligent processing and analyzing. The primary cause is XBRL’s semantics is achieved by markup language whose readability stuck in the level of natural language and lack of formal representation.

Metadata in XBRL has precise semantics and has apparent distributed, isomerism and temporal characteristics. How to represent the above features is the base of XBRL’s intelligent reasoning.

Description Logic (DL) is recommended by W3C for the logical basis of OWL. It is a knowledge representation formal method based on object and is a decidable subset of first-order predicate logic, with strong expressivity and decidability therefore can reason implicit knowledge from explicit knowledge in knowledge base. However, DL is mainly used to represent static knowledge and the corresponding reasoning. XBRL business reports dealing with a large number of tenses data which is cannot be represented by traditional DL.

Scholars have proposed various extended ideas to formalize temporal knowledge. Schmiedel (1990) firstly proposed to combine description logic and interval temporal logic with adding tense operator such as “at”, “some-time” and “all-time” to represent temporal knowledge. But this simple combination causes reasoning undecidable. Schild (1993) proposed a temporal description logic ALCT which embedding ALC and point-based tense operators. Lutz (2004) defined the interval-based description logic and gave a general TBox which expand the expressivity and proved reasoning decidable but cannot express the change of tense. Based on interval and Allen’s time interval temporal domain, Lutz and Milicic (2007) proposed the algorithm of Tableau under the general concept contain axiom. However, there are few researches on the formalization and especially in the formalization of temporal characteristics of XBRL. Pan et al. (2012) proposed a description logic TDL_{on} which can match the representation of local XBRL tense metadata but not considered the reasoning problem of distributed knowledge. Borgida and Serafini (2003) proposed a distributed description logic (DDL) for the distributed, heterogeneous information to achieve automation of intelligent reasoning.

This study analyzes the distributed and heterogeneous features of XBRL, extends the Pan’s TDL_{on} and Borgida’s DDL theory, proposes a distributed temporal description logic (DTDL_{on}) to formalize the terms, attributes and relationships in distributed XBRL reports.

DISTRIBUTED INTERACTIVE MODEL OF XBRL

Introduction of XBRL: XBRL is an open international standardization method for unconstrusted information

Corresponding Author: Ding Pan, School of Management, Jinan University, Guangzhou, 560632, China
Fig. 1: Distributed XBRL interactive model

process. Since its inception in 1998, XBRL has become one of the most effective forms of internet financial reporting. Major stock exchanges in the world require the listed companies to provide XBRL financial reporting, thus forming a distributed environment of it. XBRL's extensible feature making different countries, regions and industries can extend the basic elements to build suitable taxonomy elements according to their own situation and even in the same company, there are different requirement in metadata representation in different times. This extensible feature is determined by its architecture framework which is consist of Specification, Taxonomy and Instance Document.

Where, Specification is the basic of XBRL architecture framework, mainly used to interpret the structure of XBRL documents, detailing the syntax and semantics of XBRL Taxonomy and Instance Document. Based on the Specification, while the Taxonomy is the region's industrial information “dictionary” that being used to define the attributes and relations of items in the business reports. XBRL Taxonomy is consist of Schema (*.xsd) and Linkbase (*.xml) files. Schema file defines the structure and content schema includes the definition of name, datatype, timetype and related attributes of elements. Linkbase uses XLink to define the relations between elements. Instance Document is the enterprise business reports instances created based on Specification and Taxonomy. Taxonomy makes financial reports computer-readable and it is the core and foundation of XBRL financial reports generating and disclosing. Therefore, formalizing the semantics of Taxonomy can realize XBRL reports' identifying, analyzing and reasoning automatically.

Distributed interactive model of XBRL: In order to make distributed and heterogeneous XBRL reports enable to compose and analyze automatically and interact intelligently, we first need to realize the semantic formalization of XBRL Taxonomy and Instance Documents.

This study references the subsumption bridge rule of Borgida's DDL to realize the interact operation of distributed metadata of XBRL, establishes the distributed XBRL semantic metadata interacting model which shown in Fig. 1.

As shown in Fig. 1, based on the specified taxonomy, enterprises extract, integrate data from their own information systems and convert them to local XBRL instance documents and through consistency checks to ensure the correctness of local XBRL data. And then uses the mapping rules to achieve interoperate between metadata in XBRL Taxonomy and instance documents of distributed XBRL.

DESCRIPTION LOGIC

Description Logic is an object-based formal representation method. The basic elements are Concepts and Roles which are used to describe the field of objects and their binary relations. A DL reasoning system contains a Knowledge Box (KBox) and the reasoning mechanism. While KBox includes a terminology set (TBox) and an instance set (ABox), so KBox=<TBox, ABox>. Where TBox is the domain-specific axioms set composed of $C_i \subseteq C, R_i \subseteq R$.

Depending on the characteristics of the field of knowledge representation and application requirements, we can extend the description logic to increase the
expressivity. TDLe is the temporal extension of traditional
description logic for temporal knowledge representation
in XBRL. DDL extends the single description logic to
time and temporal knowledge and the related
reasoning mechanism.

DISTRIBUTED TEMPORAL DESCRIPTION
LOGIC DTDL_ne

According to the distributed feature of XBRL
metadata, this study extends the temporal description
logic TDLe, proposes a concept-identified-supported
distributed description logic DTDL_ne. DTDL_ne provides a
good representative to the XBRL abstract model and its
structuring metadata. It is equipped with a distributed
reasoning procedure so that it can provide rigorous formal
representation and reasoning framework to distributed
XBRL.

Syntax of DTDL_ne: DTDL_ne reasoning system consists of
two components: a plurality of local TDLe (where, i = 1, 2,
..., n, is the serial number of a single
temporal description logic); a distributed knowledge
(KBox) which includes distributed TBox and ABox; a
reasoning mechanism of distributed TBox and ABox.

Here we introduce the symbols in TDLe (Fan et al.,
2012), in which instance, concept and role express the
same as classical description logic. Specific symbols as:
C, D is the general concept (and T is the top concept, i is
the bottom concept, A is the atom concept), R for a role
that expresses a binary relation (P is an atom relation), T
is for a temporal concept.

For the distributed XBRL interoperating model, in order to
integrate two or more different XBRL reports we need to
resolve the mapping problem of different XBRL. As
described, various relations in description logic can be
transformed into subsumption relation; therefore, here
imports the subsumption rule as the mapping rule for
distributed XBRL. Defined as follows:

Definition 1: (Concept Subsumption rules) Suppose C, D
are the concept of TDLe, TDLe respectively, the
concept subsumption rules from i to j (i → j) defined as
follow:

- i : C → j : D, former concept subsumption into-
  bridge rule CIBR which states that concept C of the
  i-th TDLe is the subset of D of the j-th TDLe
- i : C → j : D, consequent concept subsumption
  into-bridge rule COBR which states that concept D
  of the j-th TDLe is the subset of C of the i-th TDLe

Definition 2: (Role Subsumption rules) Suppose R, S are
the role of TDLe, TDLe respectively, the role
subsumption rules from i to j defined as follow:

- i : R → j : S, former role subsumption into-bridge
  role RIBR which states that role R of the i-th TDLe
  is the subset of S of the j-th TDLe
- i : R → j : S, consequent role subsumption onto-
  bridge role ROBR which states that role S of the j-th
  TDLe is the subset of R of the i-th TDLe

where subsumption bridge rules has directionality
denotes that bridge rules from j to i are not necessarily
the inverse of the rules from i to j and in fact there may be no
rules in one or both the directions (Jiang et al., 2006; Kriger, 2008).

Definition 3: (Instance Subsumption rules) Suppose a is
an instance set of TDLe, b, b1, b2, ... are instances of
TDLe, i to j (i → j) instance subsumption rules include the
following two definitions:

- i : a → j : b, where, b ∈ {b1, b2, ...}, is part instance
  subsumption rules PIR, it denotes that b is one of the
  instance that related to a
- i : a → j : {b1, b2, ...}, is complete instance
  subsumption rules CIR, that instance a of TDLe
  is complete equal to instance set {b1, b2, ...} of TDLe

Definition 4: The knowledge base of distributed temporal
description logic KBox is compose of distributed
terminology set DTBox and distributed instance set
DABox, that is: KBox = (DTBox, DABox).

Definition 5: DTBox = \{TBox_{i} \mid I\}, BR, where, I is the
serial set, \{TBox_{i} \mid I\} is TBox set of TDLe, BR = \{BR\}, is
the subsumption bridge rules set. For any k ∈ I,
descriptions in TBox\_k must comply the specification of
TDLe and for any subsumption bridge rule CIBR, COBR, 
RIBR, ROBR of BR\_k and the related concept C, D, related
role R, S must comply the specification of TDLe and
TDLe respectively.

Definition 6: DABox = \{ABox_{i} \mid IR\}, where, I is the serial
set, \{ABox_{i} \mid IR\} is ABox set of TDLe, IR = \{IR\} is instance
subsumption bridge rules set from i to j. For any k ∈ I,
descriptions in ABox\_k must comply the specification of
TDLe and for any instance subsumption bridge rule PIR,
CIR of IR\_k and the related instance a and b, b1, b2, ... must
comply the specification of TDLe and TDLe respectively.
Semantics interpretation of DTDL$_{neg}$: DTDL$_{neg}$ is integrated by a number of TDL$_{neg}$ through subsumption bridge rules, so its semantics interpretation can be brought up from TDL$_{neg}$ and the domain association.

$I$ denoted as the i-th interpretation of TDL$_{neg}$, it is compose of the non null interpretation set $\Delta^1$ and interpretation function $\star$ ($I=(\Delta^1, \star)$), mapping with $\star$ that concept $i:C$ of TDL$_{neg}$ is the subset of $C$, binary relation $i:R$ is $R^1$ which is the subset of $\Delta^1 \times \Delta^1$. Detailed semantic interpretation of TDL$_{neg}$ shown in article (Schmiel, 1990).

Definition 7: In order to illustrate the interpretation relation between TDL$_{neg}$ and TDL$_{neg}$ (i.e.), we import the binary relation $r_{ij}:\Delta^1 \times \Delta^1, i,j \in I$, if $r_{ij}(C^1)$ to designate the relation between interpretation of concept $i:C$ and the concept of TDL$_{neg}$, then the distributed interpretation of DTDL$_{neg}$ defined as $DI=(\{I\}_{i}, \{r_{ij}\}_{ij})$.

Definition 8: the satisfiability of a distributed interpretation $DI=(\{I\}_{i}, \{r_{ij}\}_{ij})$, denoted as $DI \models_s$, it means we can find an interpretation in the Knowledge Box, whatever in the distributed TBox or ABox and has the following forms:

- **Satisfiability of concept**: $DI \models_s: C \rightarrow D$, if the interpretation $r_{ij}(C^1) \subseteq D^i$ (where, $\Sigma = \{\approx, \equiv\}$, $C$, $D^i \equiv (\star)(\wedge)$ holds, then DI satisfies the concept of DKBox that concept $i:C$ to j:D is satisfiable.

- **Satisfiability of subsumption relation**: if concept $C$ is included in $D$, then for $DI$ satisfies $C \subseteq D$, denoted as $DI \models_s: C \subseteq D \equiv (\star)(\wedge)$

- **Satisfiability of terminology**: if a distributed interpretation $DI \models_s$ satisfies all the assertions of each TBox (denoted as $DI \models_s$: TBox) and meets all bridge-rules interpretation (denoted as $DI \models_s$: BR$_{rij}, BR_1, BR_2 \in BR$), called that the terminology set satisfiable.

- **Satisfiability of instance relations**: if the interpretation of relation $b^i \in r_{ij}(a^i)$ satisfied then call instance $i:a$ partly contain $j:b$ satisfied, denoted as $DI \models_s: a \rightarrow j: b$; if:

$$\tau_{ij}(a^i) = \{b^i_1, b^i_2, ...\}$$

established, then instance $i:a$ completely contain $j:b = (b_1, b_2, ...)$ is satisfiable, denoted as $DI \models_s: a \rightarrow \rightarrow j: b_1, b_2, ...$

- **Satisfiability of DABox**: if distributed interpretation $DI \models_s$ all assertions of each ABox, that $DI \models_s: I(C), DI \models_s: P(a, b)$, we call that $DI$ is satisfiable to ABox, denoted as $DI \models_s$ ABox, if $DI$ meets interpretation of all instance in, denoted as $DI \models_{in, in} \in \in$, that DI is satisfiable to distributed instance set DABox, denoted as $DI \models _s$ DABox

- **Satisfiability of DKBox**: if $DI \models _s$ satisfy all assertions and instances of DKBox, call that $DI \models _s$ is a model of DKBox, denoted as $DI \models _s$ DKBox

**REASONING ALGORITHM OF DTDL$_{neg}$**

Whether a concept $C_i$ in ALCN is satisfiable is to find the interpretation $I = (\Delta^1, \star)$ that make $C_i$ satisfied. If can find it that under the interpretation $I$, then concept $C_i$ is satisfiable, otherwise it is unsatisfiable. In fact, the interpretation is uncountable that it is impossible to traverse all interpretations to judge whether it is satisfiable. In this context, Tableau algorithm being put forward, its basic idea is constructing a model to prove $C_i$ is satisfiable.

Reasoning in DTDL$_{neg}$ includes local and global parts. Representation and reasoning of version migrating in taxonomy belongs to local XBRL consistency which goal is to find a local interpretation $I$, meets $\models I_{\subseteq} C_i \subseteq D_i$, because in the ALC language, reasoning problems can be transformed into satisfiability judgment.

This study concerns the interoperability after local consistency checking, namely the global inference. In the distributed XBRL interactive model, local XBRLs interoperate based around the subsumption rules, therefore, the global reasoning in DTDL$_{neg}$ is through subsumption rules to find a distributed interpretation $DI$ satisfies $DI \models_s: C \subseteq D$, denoted as $(C \subseteq \neg D)^i = \emptyset$.

**Reasoning algorithm of DTDL$_{neg}$**: According to distributed reasoning feature this study extends traditional Tableau algorithm, establishes the interactive Tableau algorithm for distributed XBRL. Main idea is: firstly, constructing a local Tableau tree, secondly, checking whether can end the local Tableau tree through subsumption mapping rules. If cannot end it, then the concepts is satisfiable, otherwise it is unsatisfiable (Serafini and Tamlin, 2005). Specific steps designed as follows:

1. Start input: $I_0 = \{C_i(x_0)\}$
2. Root note: $x_0$
3. Output: whether TBox, $\models I_{\subseteq} C \subseteq D$ can be established

Distributed Tableau tree building process:

- Build a local complete tree through general Tableau algorithm, if there were conflict then concept in local KBox is unsatisfiable and end algorithm, otherwise go to step (2)
- For any $j \in I_{\subseteq} J_{\subseteq}$ in TDL$_{neg}$ and TDL$_{neg}$, judge whether the following statements can be met: If exists the COBR rules from $j$ to $i$, the concept $C$ in TDL$_{neg}$ being contained to $G$ in TDL$_{neg}$; IF exists the CIBR rules from $j$ to $i$, the concept $H$ in TDL$_{neg}$ being contained to $D$ in TDL$_{neg}$. Go to step (3)
Nature of DTDL_{enr} ’s reasoning algorithm: DTDL_{enr} is a subset of ALCN, the Tableau algorithm has been proved that reasoning of ALCN-concepts is reliability, terminability and completeness. DTDL_{enr} is a collection of multiple DTDL_{enr}, so it has reliability, terminability and completeness either.

De Giacomo (1995) proved the inference algorithm of ALCNis exponential time complete. DTDL_{enr} did not change the complexity of algorithm, so its complexity is exponential time complete either.

CONCLUSIONS

XBRL architecture framework determines that metadata in XBRL has typical distributed, heterogeneous and temporal features. This study extends the DTDL_{enr}, proposes a distributed and temporal-extended description logic DTDL_{enr}, uses it to represent XBRL’s distributed metadata, constructs the corresponding reasoning algorithm and proves its decidability.

ACKNOWLEDGMENT

This study was supported by the National Nature Science Foundation of China (No. 71171097, 70872020, 70771044).

REFERENCES


