PhD Thesis Summary – Dr. Lim Shin Huei

Thesis Title

Automated Verbalization of Conceptual Data Models in Bahasa Melayu and Mandarin

---

Degree Information

Degree Awarded: Doctor of Philosophy (PhD)

Field: Information Systems / Conceptual Modeling

University: INTI International University, Malaysia

Thesis Submitted: 2015

Graduation Year: 2017

Abstract

This thesis focuses on the automated verbalization of conceptual data models using Object Role Modeling (ORM), specifically within the context of Asian languages—Bahasa Melayu and Mandarin. While ORM has proven effective in capturing complex business semantics, verbalizing such models in non-English languages remains a challenge in both academic and business practice.

The research investigates how native linguistic structures affect the automated transformation of ORM diagrams into natural language sentences, aiming to bridge communication between domain experts and developers in multilingual environments. By extending the verbalization engine in the NORMA tool, the study introduces a prototype that supports Malay and Mandarin verbal outputs based on refined grammar rules, classifier patterns, and contextual sentence structures.

Results from implementation and evaluation highlight improved clarity, simplicity, and precision in ORM verbalizations in both target languages, supporting the broader goal of making conceptual modeling more accessible to non-English-speaking stakeholders.

Related Publications

1. *Automated Verbalization of ORM Models in Malay and Mandarin* – Published by IGI Global, 2016
2. *Automated Verbalization of ORM Models in Malay and Mandarin* – Published in Springer, 2013

Archival Permission Statement

I hereby grant INTI International University permission to re-archive this thesis within its internal or public thesis repository system, in accordance with the university's current policy.

(Signed: Dr. Lim Shin Huei)

Date: 19th June 2025

Automated Verbalization of Conceptual Data Models in Bahasa Melayu and Mandarin

by

Lim Shin Huei

being a thesis submitted to
INTI International University
in candidature for the degree of
Doctor of Philosophy

Faculty of Science, Technology, Engineering and Mathematics

2015

COPYRIGHT

DECLARATION

I hereby declare that the work has been done by myself and no portion of the work contained in this thesis has been submitted in support of any application for any other degree or qualification of this or any other university or institute of learning.

_____

Candidate Signature

Candidate Name: Lim Shin Huei

ACKNOWLEDGEMENTS

I am deeply thankful to my supervisor, Professor Terry Halpin, for his sage advice, kindness and willingness in supervising me for this thesis. As the leading expert on Object-Role Modeling (ORM), his deep knowledge of this conceptual modeling approach proved invaluable in addressing technical issues that arose in developing algorithms to transform complex ORM constraints into Bahasa Melayu (BM) and Mandarin verbalizations.

Next, I want to show my appreciation to Professor Tony Morgan. He encouraged me to first build my own prototype system in C# to explore basic, automated verbalization of ORM models in BM and Mandarin, in addition to developing comprehensive verbalization software that exploits and extends the basic verbalization framework provided by the Natural ORM Architect (NORMA) tool.

Matthew Curland, the main software architect of NORMA, has my gratitude for his assistance in acquainting me with the internal structures of NORMA's current verbalization framework.

Lastly, I wish to thank to my precious family and school mates for their support and patience in helping me to validate the correctness and naturalness of the verbalizations in BM and Mandarin generated by the software I developed for this thesis.

ABSTRACT

This thesis is mainly concerned with automatically verbalizing conceptual data models expressed in the graphical notation of Object-Role Modeling (ORM) into readable text expressed in two Asian languages: Bahasa Melayu (BM) which is the national language of Malaysia, Brunei and Indonesia, and one of the four official languages of Singapore; and Mandarin (the most widely used Chinese language).

In developing information systems, the most critical aspect is to specify a conceptual model that correctly and completely captures the semantics of the relevant business domain. This model can then be used to drive the later phases of the information systems engineering process. Industrial practice has shown that the most reliable way to validate the model with business domain experts is to communicate the model clearly in their native language.

Object-Role Modeling (ORM) is an approach for conceptual data modeling specifically designed to optimize communication between data modelers and business users. In Malaysia, the most common non-English languages spoken are Bahasa Melayu and Mandarin. The main objectives of this research are to develop procedures for automating the verbalization of ORM data models (including constraints and business rules) in Bahasa Melayu and Mandarin, and to implement the approach in a software tool that allows the models to be entered and verbalized in these languages.

After conducting a literature review to identify existing, relevant research, to ensure that our work will provide a novel contribution, we will specify the theoretical underpinnings by developing the relevant metamodels and information structures needed to capture the relevant logical and linguistic forms underlying the verbalizations and the transformations from ORM structures to the native languages.

We provide one full and one partial implementation of the approach by coding the structures and transformations in appropriate computing languages. Our full implementation is an independently developed prototype, mainly programmed in C#. Our partial implementation is an extension to the public domain version of the NORMA software tool for ORM, mainly coded in XML and XSLT. As the internal verbalization framework in NORMA will soon be replaced by another tree-based

verbalization framework, we limited our Mandarin and Bahasa Melayu extension of the NORMA's current verbalization framework to just a few ORM constraints.

Some of our implementation aspects (e.g. vocal rendition of our verbalizations in Mandarin) also make use of other existing public domain tools.

Evaluation of the end product involved a number of native speakers of Bahasa Melayu and Mandarin to determine the quality of the generated verbalizations in terms of clarity, unambiguity and precision.

Contents

List of Tables

List of Figures

1 Introduction

1.1 Problem Statement

In this section we point out the scope and motivation of this thesis. The next section summarizes the main contributions of this work, and the last part of this chapter provides a structural overview.

This research is concerned with the specification of conceptual modelling verbalizations in Bahasa Melayu (BM) and Mandarin, which we automatically generate, mainly by coding the relevant transforms in our independently developed prototype as well as in extensions to the NORMA system. In addition to verbalization of fact types, we cover verbalization of all the kinds of constraints currently expressible in ORM's graphical notation. We first translate ORM constraints into a logical form, based mainly on first-order logic, and then transform the logical form into linguistic forms conforming to the grammar of BM and Mandarin. The verbalizations we generate have been checked with other experts in the relevant language (BM or Mandarin) to determine their correctness and naturalness.

The main motivation for this research work is to improve the quality of conceptual data models produced by native speakers of BM or Mandarin, by allowing them to model directly in their own language, and more easily validate their models with business users who naturally communicate in those languages. Additionally, this work should enable speakers of those Asian languages to more easily master the ORM modeling approach.

To our knowledge, our work is the first to address the task of automatically generating verbalization of ORM models into BM. While our literature search did locate one paper on translating ORM models into Mandarin (5), this previous work examined only a basic fragment of ORM, made some incorrect translations, and did not include an implementation.

After determining the required translation patterns for verbalization, we tested our initial approach by coding in C# a prototype system that covered a basic subset of the ORM modeling constructs, that allowed users of BM and Mandarin to select an appropriate noun classifier (NC) for each object type, which we then incorporated in the generated verbalizations. Afterwards, we implemented verbalizers for BM and Mandarin that covered all the ORM constructs by extending our prototype as well as building extensions to the NORMA system. This involved declaring, mapping, coding, testing and debugging the verbalization transforms, as well designing and coding the user interface for relevant windows properties (e.g. for users to select relevant noun classifiers). This implementation underwent extensive testing to help ensure that it met the required functionality, correctness, and performance objectives before release.

1.2 Research Objectives

The main research that has been conducted for this thesis is summarized as follows:

a) Transformations from ORM to logical form in BM and Mandarin, and logical form to linguistic form in BM and Mandarin. For explanatory and comparison purposes, this thesis includes many examples of verbalization patterns in English. These English verbalizations have been automatically generated by the NORMA tool, using transformations developed by Halpin and Curland (e.g. [7, 16]). Our transformations adapt and modify those English verbalization patterns to conform to the grammar of BM and Mandarin.

b) Identification of the differences in core verbalization constraint patterns in BM and Mandarin, including quantifiers, modal operators, fact types (primary/inverse readings), object types, value constraints, internal and external uniqueness constraints, mandatory role constraints, subset constraints, equality constraints, exclusion constraints, inclusive-or constraints, exclusive-or constraints, frequency constraints, ring constraints, value-comparison constraints, cardinaility constraints and subtyping.

c) Introduction of a Noun Classifier (NC) property to indicate the kind of object type being used both in BM and Mandarin.
    i) Sampling the examples of types of Noun Classifier that are used in BM and Mandarin with kinds of Noun.
    ii) Highlighting the difference between English, BM and Mandarin in adopting NCs towards their own verbalization.
    iii) Identifying the kinds of noun in BM that influence ORM Verbalization for Noun Classifiers
    iv) Designing the interface to implement the NC plugin as an extension in ORM tools both in BM and Mandarin
    v) Modeling the NC usage in ORM structures to further illustrate the relationship between Noun Phrase, NounType, Classifier etc.
d) Analysis of the structure of various complex aspects of BM and Mandarin grammar
    i) Defining a way to apply the pronouns ‘who’ and ‘that’ in BM and Mandarin, covering both conjunctive and adjectival usages, and how to incorporate them in sentence strcutures.
    ii) Identification of the key differences in singular and plural grammatical number between English, BM and Mandarin.
    iii) Catering for differences in use of indefinite articles such as ‘a’ and ‘an’ in English, BM and Mandarin
    iv) Specifying how to cater for the use of quantifiers such as ‘exactly one’, ‘at most one’ and ‘more than one’ in both active and passive sentences in Mandarin.
    v) Catering for differences in the use of the preposition ‘of’ in English and Mandarin.
e) Implementing automation of the above verbalization aspects in an independent prototype as well as by extending the verbalization framework provided by the NORMA tool.
    i) Providing a language picker (Mandarin, BM) that auto-detects the way of keying in words in the relevant language.

ii) Catering for input of ORM fact types in Mandarin and BM, and other features that support automated verbalization of constraints (for example, introducing brackets around the object type for distinguishing object terms in modelling as well as the way in which a noun classifier is attached to the object type term in verbalization output).

iii) Linking of each object type to the relevant NC through NC window properties, including general and advanced search of type of NounPhrase to give various options to access the NC with further description usage of each NounPhrase and classifier picked by the user.

iv) Introducing a speech narrator feature to voclzie the verbalization output to aid users for better understanding and learning.

v) Discussing sample coding that implements the verbalization, including model creation, classifier setup and verbalization output in the desired language.

f) Performing additional extensions to the NORMA tool

i) Identifying snippets that may be used for the NC plug-in for BM and Mandarin.

ii) Defining the files to store the chosen classifiers, including .dsl, .csproj, and .xsd files.

iii) Coding the C# extension for the NC plug-in and designing the interface for NC window properties.

1.3 Structural Overview

This section summarizes which topics are covered in the written documentation and how they will be discussed. The overall work is distributed into 5 chapters as shown below.

Chapter 1: Introduction

This chapter explains about the research purpose and objetives behind our proposed idea. We also look into to identify the background study of our respected research topic.

Chapter 2: Literature Review

This chapter performs a literature review of related research to determine what has already been done in this area and to ensure that the thesis makes a significantly new contribution.

Chapter 3: Research Methodology

This chapter includes two subsections which are verbalization pattern analysis and structure analysis. The verbalization pattern analysis section discusses patterns and algorithms that may be used to capture relevant ORM structures in logical form and transform them into linguistic form for verbalization in BM and Mandarin. We also compare the verbalization in both logical and ligustic form for a wide variety of ORM constraints and subtyping in English, BM and Mandarin. In the structure analysis section we compare the critical differences between English, Bahasa Melayu and Mandarin grammar that impact verbalization generation, including noun classifiers, singular/plural forms, pronouns (e.g. ‘who’and ‘that’), and indefinite articles (e.g.‘a’ and ‘an’).

Chapter 4: Results and Discussion

Here, we discuss implemention of the model input and transformations in our independently developed prototype, as well as extensions to the Natural ORM Architect (NORMA) software tool by coding in appropriate computer languages

such as C#, XML and XSLT. Where relevant, we make use of existing public domain software to assist with aspects of entering text in Mandarin. We also evaluate the effectiveness of the work by testing it with a range of representative ORM models that reflect the expressive power of ORM, and by surveying native speakers of Mandarin and BM with respect to the intelligibility of the verbalizations generated.

Chapter 5: Conclusion

In this final chapter of the thesis, we summarize the main contributions of our research.

2 Literature Review

2.1 Introduction

A conceptual data model includes a conceptual schema and a population of instances. The fact structures and business rules (constraints or derivation rules) are best validated with subject matter experts, who best understand the business domain. Since such domain experts often lack technical expertise in the graphical languages used by modelers to capture the model, model validation is best performed by verbalizing the model to the domain experts in a controlled natural language (an unambiguous subset of natural language with restricted grammar and vocabulary) and by populating the relevant fact types with concrete examples.

This process of validation by verbalization and population is central to, and facilitated by, fact-oriented modeling approaches, such as Object-Role Modeling (ORM) [12, 13, 15, 18], Cognition enhanced Natural Information Analysis Method (CogNIAM) [25], and Fully Communication Oriented Information Modeling (FCO-IM) [2], which use fact types as their sole data structure, unlike attribute-based approaches such as Entity Reltionship (ER) modeling [4] and the class diagramming technique in the Unified Modeling Language (UML) [26].

While some fact-based modeling tools provide extensive verbalization support in English, comparatively little work exists to provide fact-based model verbalization support for other languages, especially Asian languages. As far as we know, no work apart from our own [24] has yet been published on verbalizing conceptual data models in Bahasa Melayu (Malay) or Mandarin. This thesis describes our work on verbalizing ORM models in Malay and Mandarin (more specifically, the Simplified Chinese version of Mandarin, as adopted in mainland China, Singapore and Malaysia). We discuss aspects of these languages that require special treatment in order to render natural verbalization (e.g. noun classifiers, and the order in which sentence elements are placed), and describe our current

implementation efforts, which involved creating both a prototype and an extension to the NORMA (Natural ORM Architect) tool [6].

2.2 Related Research

To our knowledge, the first controlled natural language for conceptual data modeling was Reference and Idea Language (RIDL) [20], which was developed in the 1980s based on a binary relationship version of NIAM. The model declaration part was implemented in the RIDL* tool, but the query part was never implemented. Afterwards, other fact-oriented languages were developed. In the late 1980s and the 1990s, Halpin specified the first version of Formal ORM Language (FORML) to capture ORM constraints in textual form, and provided patterns to automatically generate verbalizations of constraints in some early ORM tools (InfoDesigner, InfoModeler, VisioModeler, and Microsoft Visio for Enterprise Architects). The Language for Information Structure and Access Descriptions (LISA-D), based on Predicator Set Model (PSM), was specified [20] by researchers at Radboud University, but was not fully implemented. ConQuer, an ORM-based language for conceptual queries, was implemented in the ActiveQuery tool, which generated SQL code from ConQuer queries [3].

More recently, FORML version 2 (FORML 2) was implemented in the NORMA tool [6], providing enhanced, automated verbalization of constraints and derivation rules in second generation ORM (ORM 2) [10] in English [7, 15, 16]. Preliminary work has also been made to extend FORML 2 for full model input and queries [18]. The Constellation Query Language (CQL) [19] is currently under development in the Active Facts tool to provide modeling and conceptual query functionality for ORM. The DogmaModeler tool has been extended to verbalize basic ORM constraints in ten languages (English, Dutch, German, Italian, Spanish, Catalan, French, Lithuanian, Russian and Arabic) [22], but while the range of languages addressed is impressive, the verbalizations generated are not always grammatically correct (e.g. ‘Each Person must Has at least one Name.’, ‘Each Person WorksFor a Company must AffiliatedWith that Company.’).

Outside the fact-oriented modeling community, several high level textual modeling and/or query languages have been developed. The Object Constraint Language (OCL) [31] is used to augment UML class models with rules that cannot be expressed graphically in UML. However, OCL's attribute-based nature leads to semantic instability, its rule contexts are restricted to classes, and OCL expressions are often too technical for business users to understand and hence validate. Some textual languages for ER have been proposed (e.g. see section 16.3 of [17]), but these are limited in scope, and are prone to semantic instability caused by an underlying attribute-based model.

Controlled natural languages are often linguistics-based, employing a formal, executable subset of natural language (typically English). Attempto Controlled English (ACE) [1] supports a wide range of natural statements and queries, relying on interpretation rules (e.g. **and** has priority over **or**) to enable its text to be unambiguously translated into discourse representation structures, a syntactic variant of first-order logic (FOL). John Sowa's Common Logic Controlled English (CLCE) [30] has the full expressibility of FOL, but its use of untyped variables tends to make its expressions look more mathematical than natural. Processable ENGlish (PENG) [27] uses a controlled lexicon of predefined function words as well as domain-specific content words that can be defined by the author on the fly. For further details on controlled natural languages, see [23, 27, 29].

Basic research on multilingual verbalization of ORM conceptual models has been conducted by Jarrar, Keet and Dongili [21, 22]. Using a completely different approach from ours, they proposed basic patterns for multilingual verbalization in around 10 languages [38], but did not cover Mandarin or BM.

We are not aware of any research besides our own on verbalizing ORM models in Bahasa Melayu.

Some very basic research on the structure of Chinese sentences for ORM verbalization was done 20 years ago by Choi as a student project [5]. While this did cover some ORM constraints and made mention of noun classifiers, this previous work examined only a basic fragment of ORM, made some incorrect translations, and did not include an implementation. Our work in this regard goes well beyond

that of Choi as indicated by the following list which summarizes several of the new contributions provided by this thesis.

- Model transformations from ORM to logical form, and logical form to linguistic form
- Identification of core differences between English, BM and Mandarin verbalization with respect to quantifiers, modal operators and various kinds of ORM constraints (uniqueness, ring, subtyping, inclusive-or, etc.)
- Catering for structural differences in word/grammar complexity in BM and Mandarin (singular/plural forms, pronouns, etc.)
- Usage of noun classifiers to designate the relevant kind of object being quantified or counted
- Implementation of the automated verbalization of ORM models in BM and Mandarin.

2.3 Overview of Research Contributions

In this section, we point out various shortcomings in previous work in the relevant field and describe some ways to address those shortcomings. We also introduce some entirely new ideas/approaches that have not been previously explored. Some advantages of our approach compared with previous work are now summarized.

Some distinguishing features of our work compated with previous work in related fields:

i) Automated verbalization of ORM models in BM (no other researchers have attempted this task).

ii) Not all the constraint/predicate wordngs in Mandarin are correctly translated in the previous research done by Annai Choi [5]. For example, the subtype derivation rule '**Each** Professor is **an** Academic **who** has Status 'P' was translated by Choi as '**每一**个教授是一个学术人员, 他有身份'P'. This is better translated as '**每一**个教授是有身份'P **的**'学术人员', instead of being broken into two different parts. Choi translated the constraint '**No** Person is husband of **some** Person **and** is wife of **some** Person' as '**没有**人是**一些**人的丈夫**和一些**人的妻子'. This should

instead be translated as '**没有/无任何一个**人是**某一个**人的丈夫**同时**是**某一个**人的妻子。'. The subset/equality constraint 'Person drives **some** Car **only if/ if and only if that** Person has **some** DrivingLicense.' was translated as '人驾驶**一些**车**只有如果**人有**一些**驾执照。' but should be converted to '人驾驶**某一辆**车**除非**人有**某一个**驾驶执照。'. Moreover, Choi's work also lacks the logical form declaration to provide a formal way of creating the verbalization.

iii) Previous research on Chinese verbalization for ORM covered only some general fragments, and did not include automatic generation of the verbalizations. Our work is far more comprehensive, covering many more varieties of ORM constraints including recently introduced ones, as well including grammatical patterns for both BM and Mandarin. The previous research by Choi [5] basically covered the study of Chinese grammer in terms of singular/plural, past tense/present tense, word ordering, 'a' and 'an' and translation of basic constraints (e.g. only a few of ORM's ring constraints were covered).

iv) More detailed studies in noun classifier usage, such as introducing new patterns such as Δ, [] and ▲ as well as subclasses of noun classifier and different criteria that influence noun classifier applicability toward verbalization in both BM and Mandarin.

v) A modeling prototype system that supports automated verbalization in both Mandarin & BM with some new useful features to aid better verbalization (language picker, speech narrator, NC selector, etc.).

2.4 Google Translate vs our Verbalization

In order to demonstrate that my ORM verbalizations do a better job at capturing the the intended meaning in both BM and Mandarin than provided by translation tools developed by the natural language processing (NLP) community, we used the well known online translator tool Google Translate to test and compare the different

outcomes of the BM and Mandarin statements that are generated when converted directly from English verbalizations.

This translation testing revealed various limitations of Google Translate which often failed to produce grammatically correct BM/Mandarin sentences with the intended meaning. These limitations include word ordering problems, lack of suitable noun classifiers for various kinds of noun, and confusing phraseology.

Examples of these deficiencies are illustrated below, where the translations generated from Google Translate are contrasted with my automated verbalizations. As input to Google Translate, we used the English verbalizations automatically generated by the NORMA tool for varios ORM graphical constraints. This clearly demonstrates that our direct approach yields superior results than could be obtained by feeding the automated English verbalizations into Google Translate as an indirect way of automating verbalizations in BM and Mandarin.

Examples of Translation in our NORMA Verbalization vs Google Translate

1) **It is possible that some** Country is the birthplace of **more than one** Person. (NORMA English Verbalization)

   i) **这可能是一些**国家是**不止一个**人的出生地。(Google translate)

      某一个国家是**多过一个**人的出生地**是可能的**。(Our Verbalization)

   ii) **Ia adalah mungkin bahawa beberapa** Negara adalah tempat kelahiran **lebih daripada satu** Orang. (Google translate)

      Ia adalah mungkin bahawa sesebuah Negara ialah tempat kelahiran bagi **lebih daripada satu** Orang**.** (Our Verbalization)

2) **It is forbidden that some** Immigrant has **no** Passport.
   (NORMA English Verbalization)

   i) **它是被禁止的，有些**移民**没有**护照。 (Google translate)

      某一位移民拥有**无任何一本**护照**是被禁止的**。(Our Verbalization)

ii) **Adalah dilarang bahawa beberapa** imigran **tidak** mempunyai pasport. (Google translate)

Ia adalah dilarang bahawa seseorang Immigrasi mempunyai **tiada mana-mana satu buku** passport. (Our Verbalization)

3) **It is impossible that some** Person was born in **no** Year.
(NORMA English Verbalization)

i) **这是不可能的，有些**人出生在**没有**新年。(Google translate)

某一个人出生于**无任何一个**年份**是不可能的**。(Our Verbalization)

ii) **Ia adalah mustahil bahawa sesetengah** Orang dilahirkan dalam Tahun **tidak**. (Google translate)

Ia adalah tidak mungkin bahawa sese Orang dilahirkan **dalam tiada mana-mana satu** Tahun. (Our verbalization)

4) **For each** Student **and** Course, **that** Student in **that** Course obtained **at most one** Rating.
(NORMA English Verbalization)

i) **对于每一个**学生**和**课程，**最多一个**等级获得学生在课程。(Google translate)

对于每一个 ‘学生。课程’，**那一名**学生在**那一门**课程内获得**最多一个**分数。 (Our Verbalization)

ii) **Bagi setiap** Pelajar **dan** Kursus, bahawa Pelajar dalam Kursus yang diperolehi **paling banyak satu** Rating. (Google translate)

Bagi setiap ‘Pelajar 。Kursus’, Pelajar **itu** di dalam Kursus **itu** memperolehi **paling banyak satu** Markah. (Our Verbalization)

5) **For each** Player **and** Sport, **that** Player obtained **at most one** Ranking in that Sport.
(NORMA English Verbalization)

i) **对于每一个**球员**和**运动，**这得最多只有一个**球员在运动排名。
(Google translate)

对于每一个选手**和**运动项目**的组合，那一位**选手获得**最多一个**名次在**那一项**运动项目内。 (Our Verbalization)

ii) **Untuk setiap** pemain **dan** Sukan, yang diperolehi pemain **paling banyak satu** Ranking dalam Sukan **itu**. (Google translate)

Bagi setiap gabungan dengan Pemain **dan** Sukan, Pemain **itu** memperolehi **paling banyak satu** Kedudukan di dalam Sukan **itu**.
(Our Verbalization)

6) **It is impossible that for some** Driver **and** CarPlatNumber, **there is more than one** Car **such that that** Car is owned by **that** Driver **and that** CarPlatNumber is of **that** Car.
(NORMA English Verbalization)

i) **这是不可能的，对于一些**司机**和**汽车开发平台号码，**有一个以上的**汽车，**这样的**汽车是由司机**和**汽车开发平台没有拥有的是轿车。
(Google translate)

对于每一个驾驶者**和**汽车号牌**的组合，其拥有多过一辆**汽车**为此那一辆**汽车是**那一名**驾驶者所属**和那一个**汽车号牌是**那一辆**汽车所属是**不可能的**。 (Our Verbalization)

ii) **Ia adalah mustahil bahawa bagi sesetengah** pemandu **dan** Sewa Plat Nombor, **terdapat lebih daripada satu** kereta supaya kereta yang dimiliki oleh pemandu **itu dan bahawa** CarPlatNo adalah kereta **itu**.
(Google translate)

Ia adalah tidak mungkin bahawa bagi sesuatu gabungan bagi Pemandu **dan** PlatNomborKereta, **terdapat lebih daripada satu buah** Kereta **di mana** Kereta **itu** dimiliki oleh Pemandu **itu** dan PlatNomborKereta **itu** ialah kepunyaan Kereta **itu**.

(Our Verbalization)

7) **Each population of** Senator contains **at most** 100 **instances**.
(NORMA English Verbalization)

i) 参议员**的每一个群体**含有**至多** 100 **个实例**。(Google translate)

每一个参议员**的群体**包含**最多** 1 0 0 **位**。 (Our Verbalization)

ii) **Setiap penduduk** Senator mengandungi **paling banyak** 100 **kes**.
(Google translate)

Setiap populasi untuk Senator mengandungi **paling banyak** 100 **orang**. (Our Verbalization)

Limitations of Google translate:

1) Mandarin

- No support for a variety of noun classifiers for specific kind of nouns. Instead, only a designated, common noun classifer '个' is used to quantify every single noun, which is unacceptable for formal verbalizations in Mandarin.
- The quantifier 'some' is translated to '一些/有些' (there are exist several things at the same time) which doesn't achieve the same meaning as '某一'.
- Modal operators (e.g. 'it is possible that') are not placed correctly—for better readings in Mandarin, these should be placed right at the end of the phrase. Moreover, complicated sentence structures are produced that fail to capture the meaning in correct manner (e.g. see (6)(i)).

- The quantifier ‘No’ in Mandarin can be translated as ‘没有/无任何一[Δ]; but in the Google translate example 2(i) , the usage of ‘没有’ renders the sentence incorrectly. In this case, ‘No’ means ‘does not exist anything’; thus ‘无任何一[Δ]’ is the better option that I suggested.
- Internal uniqueness constraints that span 2 roles such as ‘for each student and course’ are not tranalsted well. In Mandarin, it means for each individual entity that possess each element itself instead as a group that share an element. Thus **每一个‘学生。**课程**’** will be favoured over **每一个**学生**和**课程 which is shown in example(4)(i)。The first option tends to combine ‘学生’(student) and ‘课程’(course) as one shared property with ‘。’ and encloses them inside ‘’ bracket , thus eventually making the whole sentence clear and straightforward.
- The quantifier ‘that’ which is translated as ‘那一Δ’ is shown in examples 4(i), 5(i) and 6(i). The failure to convert ‘that’ to ‘那一’ loses the intended meaning.
- In example (7)(i), the word ‘instance’ in English means ‘case’ or ‘example’, and is often used for verbalizations involving quantity with no use of noun classifers. In Mandarin, ‘instance’ will instead be replaced by the noun classifier that is specifically appropriate to the indicated noun e.g ‘100 instances → 100 位 / 辆 / 本 etc’.

2) Bahasa Melayu

- No support at all for noun classifers in BM, which typically renders the translation incorrect for formal BM writing.
- The quantifier ‘some’ is translated to ‘beberapa’ or ‘sesetengah’ (there exist several things at the same time) which doesn’t achieve the same meaning as ‘sese[Δ]’ or ‘sesuatu’.
- There are minor sentence segment disorder problems such as those shown in (5)(i).
- The quantifier ‘No’ in BM can be translated as ‘tiada/tidak’ , but in the Google translate example in 2(ii), the sentence structuring that applies

‘tidak’(no) places it in the wrong position, appearing at the back of the entity ‘Tahun’ (Year)—this renders the whole statement vague and meaningless. Of these two sentences ‘**tidak** dilahirkan dalam Tahun’ and ‘dilahirkan dalam **tiada mana-mana satu** Tahun’, the second option more approaches the meaning of ‘was born in **no** Year’.

- For internal uniqueness constraints that span 2 roles such as ‘for each student and course’, in BM it means for each individual entity that possess each element itself instead as a group that share an element, thus **Setiap ‘**Pelajar。Kursus**’** will be favoured over Setiap Pelajar **dan** Kursus which is shown in example (4)(i)。The first option l tends to combine ‘Pelajar’ (student) and ‘Kursus’ (Course) as one shared property with ‘。’ and encloses them inside ‘’ bracket . This eventually makes the whole sentences clear and straight to the point.
- The quantifier ‘that’ is translated as ‘itu’ as shown in examples 4(i), 5(i) and 6(i). The failure to convert ‘that’ to ‘itu’ loses the intended meaning.
- In example (7)(i), the word ‘instance’ in English means ‘case’ or ‘example’. In BM ‘instance’ will be replaced by the noun classifier that specifically suits the indicated Noun e.g ‘100 intances → 100 orang / buah / helai etc’.

In short, oury work is focused on direct, automated, unambiguous verbalization of ORM models in Mandarin and Bahasa Melayu, using specific patterns in a controlled natural language to provide much more specific and accurate translations.

3 Research Methodology

3.1 Verbalization Patterns

3.1.1 Modal Operators and Quantifiers

In this chapter, we explain the fundamental patterns involving the transformation from ORM to logical form, and logical form to linguistic form in BM and Mandarin. We cover verbalization of object types, fact types and a wide range of contraints used in ORM models.

In this first section, we discuss in detail the two main types of modality (alethic and deontic), along with some standard quantifiers and noun classifiers that are used to express various constraints. These are used extensively as logical elements within verbalizations of ORM constraints.

Table 3.1 shows the logical symbols and verbalizations in English, BM and Mandarin for the alethic and deontic modal operators.

Table 3.1 Modal Operators

Modality	Symbol	English Verbalization	BM Verbalization	Mandarin Verbalization
Alethic	□	It is necessary that	Ia adalah perlu bahawa	是必要的
Alethic	◊	It is possible that	Ia adalah mungkin bahawa	是可能的
Alethic	~◊	It is impossible that	Ia adalah mustahil bahawa	是不可能的
Deontic	***O***	It is obligatory that	Ia adalah wajib bahawa	是强制性的
Deontic	***P***	It is permitted that	Ia adalah dibenarkan bahawa	是允许的
Deontic	***F***	It is forbidden that	Ia adalah dilarang bahawa	是被禁止的

Table 3.2 shows the standard readings for the quantifiers that are used in formalizing ORM constraints. We introduce a new symbol ‘Δ’ for noun classifiers in BM and Mandarin ORM verbalization. A square bracket ‘[]’ around ‘Δ’ indicates optionality, to allow that in some cases a noun classifier is not needed. For example,

in BM ‘orang’ can be used as a noun phrase as well as a noun classifier, and when used as a noun phrase no additional classifier is used.

Table 3.2 Noun classifiers and logical quantifiers

Symbol	Name	English Readings	BM Readings	Mandarin Readings
Δ	Noun classifiers	*-None-*	Penjodoh bilangan (orang, ekor, buah, ….)	量词 (位、条、辆、 ...)
∀	Universal quantifier	each, for each	Setiap, Bagi Setiap	每一Δ
∃	Existential quantifier	some, a, an, there is at least one	Sese[Δ], Sesuatu	某一Δ
$\exists^{0..1}$	AtMostOne quantifier	at most one	Paling banyak satu [Δ]	最多一Δ
$\exists^{1}$	ExactlyOne quantifier	exactly one	Hanya satu [Δ]	仅有一Δ
$\exists^{2..}$	MoreThanOne quantifier	more than one, there are at least two	Lebih daripada satu [Δ]	一Δ以上的, 多过一Δ, 至少两Δ
	Identity reference	the same that	Yang sama Itu	同一Δ 那一Δ
	No quantifier	No	Tiada sebarang	没， 无任何一Δ

3.1.2 Object Types

ORM object types are classified into entity types (e.g. Person, Gender), value types (e.g. PersonName, GenderCode) and data types (e.g. string, integer). In this thesis, we confine our verbalization of object types to entity types and value types.

Table 3.3 shows some examples of basic verbalization of object types in English, Mandarin and BM. For entrity types, the reference scheme is also verbalized. The following color scheme is used to distinguish the different components in the verbalization.

Purple→ Object Type

Green→ Predicate

Red → Quantifier or Noun Classifier

Table 3.3 Verbalization of Object Types in English, Mandarin and BM

English Verbalization	Chinese Verbalization	BM Verbalization
Value Types GenderCode **is a value type**. **Entity Types** Gender**is an entity type**. **Reference scheme**: Gender has GenderCode **Reference mode**: .code	**值类型** 性别代码**是个数值类型**。 **实体类性** 性别代码**是个实体类型**。 **参考结构：** 性别有性别代码。 **参考模式：** .代码	**Jenis Nilai** KodJantina **ialah satu jenis nilai.** **Jenis Entiti** KodJantin **ialah satu jenis entiti.** **Skema Rujukan:** Jantina ada KodJantina. **Mod Rujukan:**.Kod

In Mandarin, most nouns and verbs have the same form regardless of whether they are referring to one or multiple entities. The number is inferred from context and is always placed ahead of a noun to distinguish whether a noun is singular or plural (e.g. 一个人, 两个人). As you can see, '是' in Chinese is used to indicate both 'is' and 'are' in English. The same goes for 'has' and 'have' in English; Chinese uses '有' to represent either one of them. Chinese grammars tend to emphasize the right usage of noun classifiers to define the type of noun; thus here noun classifiers are used to classify the noun itself instead of singular and plural articles and nouns ('a','an', '-s' etc.) that apply in English. Sometimes it is optional to omit the 'a' or 'an' in Chinese (e.g. 'she is **a** girl' which can mean '她是个女生' as well as '她是女生').

In Bahasa Melayu, most nouns and verbs have the same form regardless of whether they are refer to one or multiple entities. The number is inferred from context and is always placed ahead of noun to distinguish whether a noun is singular or plural (e.g. satu buah kereta, dua buah kereta). Note that 'ialah' in BM is used to indicate both 'is' and 'are' in English. The same goes for 'has' and 'have' in English; BM uses 'ada' to represent either one of them. BM grammars tend to emphasize the right usage of noun classifiers to define the type of noun; thus here noun classifiers

are used to classify the noun itself instead of singular and plural forms (‘a’, ‘an’, ‘-s’ etc.) that apply in English.

3.1.3 Fact Types

An ORM fact type corresponds to a set of typed predicates. For example, the fact type verbalized as ‘Person was born in Country’ may also be verbalized in the inverse direction as ‘Country is the birthplace of Person’. Table 3.4 shows how this fact type is verbalized in English, Mandarin and BM. In addition, it shows the verbalization of a simple mandatory role constraint (constraints are discussed in details in later sections). The following color scheme is used to distinguish the different components in the verbalization.

Bold Black → Quantifier

Purple→ Object Type

Green→ Predicate

Red → Noun Classifier

Table 3.4 Verbalization of a simple fact type and constraint

English Verbalization	Chinese Verbalization	BM Verbalization
Person was born in Country. -- primary reading / **Country is the birthplace of Person**. -- inverse reading **Each Person was born in at most one Country**. -- constraint reading	人出生于国家 -- primary reading /国家是人的出生地 --inverse reading 每一个人出生于最多一个国家 --constraint reading	**Orang dilahirkan di Negara** -- primary reading /**Negara ialah tempat lahiran bagi Orang** -- inverse reading **Setiap Orang dilahirkan di paling banyak satu buah Negara.** --constraint reading

Note the difference between English and Chinese of the inverse predicate reading structure. For example, the predicate reading ‘是…出生地’ (is birthplace of) includes a slot after the first character for placement of ‘人’ (Person).

In Chinese and BM, a noun classifier must be placed between each quantifier and countable object type. The value of noun classifier varies depending on the type of noun type being quantified. Sometimes an object type can own multiple different noun classifers to identify itself and even share the same identical noun classifier with other object types. Noun classifiers will be discussed in detail in a later section.

3.1.4 Internal Uniqueness Constraints on Binary Fact Types

This section clarifies the differences in logical form patterns that apply for internal uniqueness constraints on binary fact types, as well as the differences between these three languages in the linguistic forms for these constraints. Since BM and Mandarin have their own way of expressing the ORM constraints which differentiate them from typical English, the logical form of the ORM structures are considered first.

3.1.4.1 Simple Uniqueness Constraints on Binary Fact Types

An ORM predicate may have many readings, each of which may be composed of various parts, such as front text and text parts (prebound, postbound, and trailing). The structure of a predicate reading is irrelevant to the logical form, which denotes a predicate by a single predicate symbol (e.g. *R* or *S*). Verbalizations may be displayed in positive form (e.g. Each Person was born on at most one Date.) or negative form (e.g. It is impossible that some Person was born on more than oneDate.).

A simple uniqueness constraint applies to exactly one role of a fact type. Figure 3.1 shows the patterns for various cases of a simple uniqueness constraint applied to an *n*:1 (many-to-one) binary fact type. Alethic constraints display in violet and deontic constraints in blue (with an "o" for "obligation" at one end).

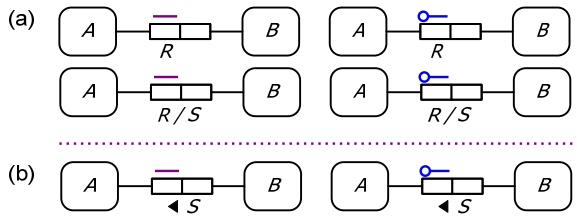


Figure 3.1 Simple uniqueness constraints on binary fact types with/without predicate reading from constrained role

For discussing logical forms os simple uniqueness constraints, we focus on the *n*:1 cases shown in Figure 1, but the forms for 1:*n* and 1:1 cases may be dealt with similarly.

Table 3.5 shows the logical forms of the verbalizations for the cases in Figure 3.1, with positive (+ve) verbalizations shown before the negative (-ve) verbalizations. For each case, the alethic constraint is shown before the deontic constraint. The BM and Mandarin forms include a noun classifier (denoted here by Δ) to categorize the kind of thing being counted.

The square brackets '[]' around 'Δ' in BM indicate that the noun classier may sometimes be omitted (e.g. if the object type is a value type or is the entity type Orang (Person)). In BM, the modal operator appears first (as in English), but in Mandarin it appears last. For positive verbalizations, the modal necessity operator □ **(**It is necessary that**)** at the start of a constraint verbalization may be omitted (this is a user choice).

Table 3.5 Logical forms of constraint verbalizations for the cases in Figure 3.1

Case	Modality	English Logical Form	BM Logical Form	Mandarin Logical Form
+ve(a)	Alethic Deontic	$\Box\forall x{:}A\exists^{0..1}y{:}B\ xRy$ $\boldsymbol{O}\forall x{:}A\exists^{0..1}y{:}B\ xRy$	$\Box\forall x{:}A\exists^{0..1}[\Delta]y{:}B\ xRy$ $\boldsymbol{O}\forall x{:}A\exists^{0..1}[\Delta]y{:}B\ xRy$	$\forall\Delta x{:}A\exists^{0..1}\Delta y{:}B\ xRy\Box$ $\forall\Delta x{:}A\exists^{0..1}\Delta y{:}B\ xRy\boldsymbol{O}$
+ve(b)	Alethic Deontic	$\Box\forall x{:}A\exists^{0..1}y{:}B\ ySx$ $\boldsymbol{O}\forall x{:}A\exists^{0..1}y{:}B\ ySx$	$\Box\forall x{:}A\exists^{0..1}[\Delta]y{:}B\ ySx$ $\boldsymbol{O}\forall x{:}A\exists^{0..1}[\Delta]y{:}B\ ySx$	$\forall\Delta x{:}A\exists^{0..1}\Delta y{:}B\ ySx\Box$ $\forall\Delta x{:}A\exists^{0..1}\Delta y{:}B\ ySx\boldsymbol{O}$
-ve(a)	Alethic Deontic	$\sim\Diamond\exists x{:}A\exists^{2..}y{:}B\ xRy$ $\boldsymbol{F}\exists x{:}A\exists^{2..}y{:}B\ xRy$	$\sim\Diamond\exists[\Delta]x{:}A\exists^{2..}[\Delta]y{:}B\ xRy$ $\boldsymbol{F}\exists[\Delta]x{:}A\exists^{2..}[\Delta]y{:}B\ xRy$	$\exists\Delta x{:}A\exists^{2..}\Delta y{:}B\ xRy\sim\Diamond$ $\exists\Delta x{:}A\exists^{2..}\Delta y{:}B\ xRy\boldsymbol{F}$
-ve(b)	Alethic Deontic	$\sim\Diamond\exists x{:}A\exists^{2..}y{:}B\ ySx$ $\boldsymbol{F}\exists x{:}A\exists^{2..}y{:}B\ ySx$	$\sim\Diamond\exists[\Delta]x{:}A\exists^{2..}[\Delta]y{:}B\ ySx$ $\boldsymbol{F}\exists[\Delta]x{:}A\exists^{2..}[\Delta]y{:}B\ ySx$	$\exists\Delta x{:}A\exists^{2..}\Delta y{:}B\ ySx\sim\Diamond$ $\exists\Delta x{:}A\exists^{2..}\Delta y{:}B\ ySx\boldsymbol{F}$

To illustrate the transformation from logical to linguistic form, the two uniqueness constraints in Figure 3.2 (see next page) will now be verbalized. Constraint C1 has a predicate reading from its constrained role, while constraint C2 has a predicate reading starting only from its corole.

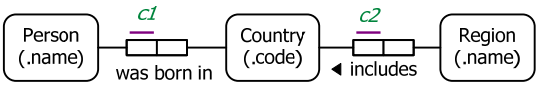


Figure 3.2 Examples of simple uniqueness constraints on binary fact types

Constraint C1 +ve verbalization:

English: **[It is necessary that] Each** Person was born in **at most one** Country.

BM: **[Ia adalah perlu bahawa] Setiap** Orang dilahirkan di **paling banyak satu** buah Negara.

Mandarin: **每一个人出生于最多一个国家[是必要的]**.

Constraint C1 -ve verbalization:

English: I**t is impossible thatsome** Person was born in **more than one** Country.

BM: **Ia adalah mustahil bahawa sese**orang dilahirkan di **lebih daripada satu** buah Negara.

Mandarin: 某一个人出生于多过一个国家**[是不可能的]**.

Constraint C2 +ve verbalization:

English: **[It is necessary that] For each** Country, **at most one** Region includes **that** Country.

BM: **[Ia adalah perlu bahawa] Bagi setiap** Negara, **paling banyak satu** buah Rantau mengandungi Negara **itu**.

Mandarin: 对**于每一个国家,最多一个地区包含那一个国家[是必要的]**.

Constraint C2 -ve verbalization:

English: **It is impossible that for some** Country, **more than one** Region includes **that** Country.

BM: **Ia adalah mustahil bahawa setiap** Negara, **lebih daripada satu** buah Rantau mengandungi Negara **itu**.

Mandarin: 对**于每一个国家, 多过一个地区包含那一个国家[是不可能的]**.

The absence of a simple, alethic uniqueness constraint on a role of an *n*:1 binary is also explicitly verbalized for the fact type (e.g. It is possible that more than one Person was born in the same Country). Such verbalizations translate in a similar way. For example the English logical form of the above verbalization is $\Diamond\exists y{:}B\exists^{2..}x{:}AxRy$. In BM this becomes $\Diamond\exists y{:}B\exists^{2..}[\Delta]x{:}AxRy$; in Mandarin we get $\exists y{:}B\exists^{2..}\Delta x{:}AxRy\Diamond$.

Table 3.6 Logical forms of verbalizations of the absence of an alethic uniqueness constraint for the cases in Figure 3.1

Case	English Logical Form	BM Logical Form	Mandarin Logical Form
(a)	$\Diamond\exists y{:}B\ \exists^{2..}\ x{:}A\ xRy$	$\Diamond\exists[\Delta]y{:}B\ \exists^{2..}\ [\Delta]x{:}A\ xRy$	$\exists\Delta y{:}B\ \exists^{2..}\Delta x{:}A\ xRy\Diamond$
(b)	$\Diamond\exists y{:}B\ \exists^{2..}\ x{:}A\ ySx$	$\Diamond\exists[\Delta]y{:}B\ \exists^{2..}\ [\Delta]x{:}A\ ySx$	$\exists\Delta y{:}B\ \exists^{2..}\Delta x{:}A\ ySx\Diamond$

For case (a), if *A* and *B* are distinct object types, the linguistic forms are as shown below, in English, BM and Mandarin respectively.

It is possible that more than one *A R* **the same** *B*.

Ia adalah mungkin bahawa lebih daripada satu[Δ]*A R B* **yang sama**.

多过一Δ*AR* 同一Δ*B* 是可能的。

For case (a), if *A* = B, then the linguistic form is modified as follows, using A_1 for the common type:

It is possible that for some A_1, **more that one***A R* **that**A_1.

Ia adalah mungkin bahawa bagi seseΔ**/sesuatu** A_1, **lebih daripada satu** [Δ]*A R* A_1 **itu**.

对于某一ΔA_1, 多过一Δ*AR* 那一ΔA_1 是可能的。

For case (b), if *A* and *B* are distinct object types, the linguistic forms are as shown below, in English, BM and Mandarin respectively. If A = B, a similar adjustment is made as for case (a) above.

It is possible that some *B S* **more than one** *A*.

Ia adalah mungkin bahawa seseΔ**/sesuatu** *B S* **lebih daripada satu**[Δ]*A*.

某一Δ*BS* 多过一Δ*A* 是可能的。

Table 3.7 shows some further examples of verbalizing the absence of an alethic uniqueness constraint on the role of a binary fact type.

Table 3.7 Verbalizations of the absence of an alethic uniqueness constraint on a role of a binary fact type

English	BM	Mandarin
It is possible that some Country is the birthplace of **more than one** Person.	**Ia adalah mungkin bahawa sesebuah** Negara ialah tempat kelahiran bagi **lebih daripada satu** Orang**.**	**某一个**国家是**多过一个**人的出生地**是可能的**。
It is possible that more than one Person was born in **the same** Country**.**	**Ia adalah mungkin bahawa sese**Orang dilahirkan di Negara **yang sama**.	**某一个**人出生于**同一个**国家**是可能的**。
It is possible that for some Person$_1$, **more than one** Person **has that** Person$_1$ as father.	**Ia adalah mungkin bahawa bagi sese**Orang$_1$**, lebih daripada satu** Orang mempunyai Orang$_1$ **itu** sebagai ayah.	**对于某一个**人 $_1$,**多过一个**人拥有**那一个**人 $_1$ 作为父亲**是可能的**。
It is possible that some Car is driven by **more than one** Person.	**Ia adalah mungkin bahawa sesebuah** Kereta dipandu oleh **lebih daripada satu** Orang**.**	**某一辆**汽车被**多过一个**人所驾驶**是可能的**。

3.1.4.2 Spanning Uniqueness Constraint on a Binary Fact Type

In ORM, all fact types are set-based, rather than bag-based. Given this understanding, a spanning uniqueness constraint on a binary fact type may be verbalized by combining the verbalizations of the absence of a simple uniqueness constraint on each role of the fact type. The logical forms of the positive verbalizations for the spanning uniqueness constraints in Figure 3.3 are shown in Table 3.8.

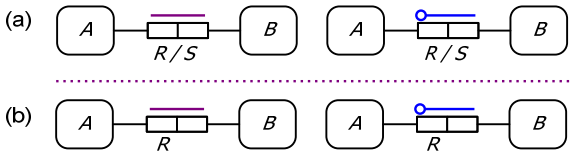


Figure 3.3 Spanning uniqueness constraints on binary fact types

Table 3.8 Logical forms of positive verbalizations of the spanning uniqueness constraint for the cases in Figure 3.3

Case	Modality	English Logical Form	BM Logical Form	Mandarin Logical Form
(a)	Alethic	◊(∃*x:A* ∃$^{2..}$ *y:B xRy* & ∃*x:B* ∃$^{2..}$ *y:A ySx*)	◊(∃[Δ]*x:A* ∃$^{2..}$ [Δ] *y:B xRy* & ∃[Δ]*x:B* ∃$^{2..}$ [Δ] *y:A ySx*)	(∃Δ*x:A* ∃$^{2..}$Δ *y:B xRy* & ∃Δ*x:B* ∃$^{2..}$Δ *y:A ySx*)◊
	Deontic	***P***(∃*x:A* ∃$^{2..}$ *y:B xRy* & ∃*x:B* ∃$^{2..}$ *y:A ySx*)	***P***(∃[Δ]*x:A* ∃$^{2..}$ [Δ]*y:B xRy* & ∃[Δ]*x:B* ∃$^{2..}$ [Δ] *y:A ySx*)	(∃Δ*x:A* ∃$^{2..}$Δ *y:B xRy* & ∃Δ*x:B* ∃$^{2..}$Δ *y:A ySx*)***P***
(b)	Alethic	◊(∃*x:A* ∃$^{2..}$ *y:B xRy* & ∃*x:B* ∃$^{2..}$ y:A *xRy*)	◊(∃[Δ]*x:A* ∃$^{2..}$ [Δ]*y:B xRy* & ∃[Δ]*x:B* ∃$^{2..}$ [Δ] *y:A xRy*)	(∃Δ*x:A* ∃$^{2..}$Δ *y:B xRy* & ∃Δ*x:B* ∃$^{2..}$Δ *y:A xRy*)◊
	Deontic	***P***(∃*x:A* ∃$^{2..}$ *y:B xRy* & ∃*x:B* ∃$^{2..}$ *y:A xRy*)	***P***(∃[Δ]*x:A* ∃$^{2..}$ [Δ]*y:B xRy* & ∃[Δ]*x:B* ∃$^{2..}$ [Δ] *y:A xRy*)	(∃Δ*x:A* ∃$^{2..}$Δ *y:B xRy* & ∃Δ*x:B* ∃$^{2..}$Δ *y:A xRy*)***P***

Figure 3.4 shows three cases for a spanning alethic uniqueness constraint. Case (a) includes both forward and inverse predicate readings. Case (b) has no inverse predicate reading. In case (c), both roles are hosted by the same object type (this corresponds to *A* = *B* in the logical forms above).

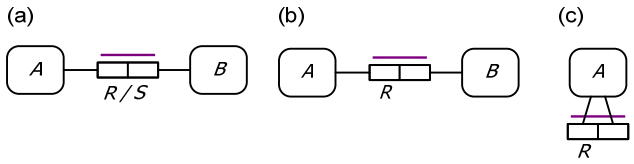


Figure 3.4 Alethic spanning uniqueness constraints on binary fact types

The positive verbalizations for Figure 3.4(a) are shown below. Here "*ft*" indicates front text, for those rare cases where some of the predicate text is placed before the first object term (e.g. consider the fact type 'occasionally Person drives Car').

English: **It is possible that** *ft* **some** *A R* **more than one** *B* **and that** *ft* **some** *B S* **more than one** *A*.

BM: **Ia adalah mungkin bahawa** *ft* **sese** Δ**/sesuatu** *A R* **lebih daripada satu** Δ*B* **dan** *ft* **sese**Δ**/sesuatu** *B S* **lebih daripada satu[**Δ] *A*.

Mandarin: *ft* 某一Δ*AR* 多过一Δ*B* 和 *ft* 某一Δ*BS* 多过一Δ*A* 是可能的。

The positive verbalizations for Figure 3.4(b) are shown below.

English: **It is possible that** *ft* **some** *A R* **more than one** *B* **and that for some**B_1, *ft* **more than one** *A R* **that** B_1.

BM: **Ia adalah mungkin bahawa***ft* **sese**Δ **/sesuatu** *A R* **lebih daripada satu** Δ*B* **dan** bagi **sese**Δ**/sesuatu** B_1, *ft* **lebih daripada satu[**Δ] *A R* B_1 **itu**.

Mandarin: *ft* 某一Δ*AR* 多过一Δ*B* 和对于某一ΔB_1, *ft* 多过一Δ*AR* 那一ΔB_1 是可能的。

The positive verbalizations for Figure 3.4(c) are shown below.

English: **It is possible that** *ft* **some** *A R* **more than one** *A* **and that for some** A_1, *ft* **more than one** *A R* **that** A_1.

BM: **Ia adalah mungkin bahawa** *ft* **sese** Δ**/sesuatu** *A R* **lebih daripada satu** Δ*A* **dan** bagi **sese**Δ**/sesuatu** A_1,*ft* **lebih daripada satu[**Δ] *A R* A_1 **itu**.

Mandarin: *ft* 某一Δ*AR* 多过一Δ*A* 和对于某一ΔA_1,*ft* 多过一Δ*AR* 那一ΔA_1 是可能的。

Some concerte examples of verbalizations of these patterns are shown in Table 3.9.

Table 3.9 Examples of positive verbalizations of spanning uniqueness constraints

English	BM	Mandarin
It is possible that some Person speaks **more than one** Language **and that some** Language is spoken by **more than one** Person.	**Ia adalah mungkin bahawa sese** Orang bercakap **lebih daripada satu buah** Bahasa **dan sesebuah** Bahasa dicakap oleh **daripada lebih satu** Orang.	某一个人讲多过一种语言和某一种语言被多过一个人讲是可能的。
It is possible that some Person speaks **more than one** Language **and that for some** Language, **more than one** Person speaks **that** Language.	**Ia adalah mungkin bahawa sese** Orang bercakap **lebih daripda satu buah** Bahasa **dan bagi sese buah** Bahasa, **lebih daripada satu** Orang bercakap Bahasa **itu**.	某一个人讲多过一种语言和对于某一种语言，多过一个人讲那一种语言是可能的。

It is possible that some Person likes **more than one** Person **and that for some** Person1, **more than one** Person likes **that** Person1。	**Ia adalah mungkin bahawa sese** Orang menyukai **lebih daripada satu** Orang **dan bagi sese** Orang1, **lebih daripada satu** Orang menyukai Orang1i**tu**.	**某一个**人喜欢**多过一个**人**和对于某一个**人 1, **多过一**个人喜欢**那一个**人 1。
It is possible that some Person was awarded **more than one** AcademicDegree **and that for some** AcademicDegree, **more than one** Person was awarded **that** AcademicDegree。	**Ia adalah mungkin bahawa sese** Orang dianugerahkan dengan **lebih daripada satu** Ijazah **dan bagi sese buah** Ijizah, **lebih daripada satu** Orang dianugerakhan dengan Ijazah i**tu**.	**某一个**人荣获**多过一个**学位**和对于某一个**学位, **多过一**个人荣获**那一个**学位。

3.1.5 Internal Uniqueness Constraints on *n*-ary Fact Types

A fact type with n roles where $n \geq 3$ is called an n-ary fact type. In ORM, each internal uniqueness constraint on a fact type with n roles must span either all but one of the roles (n–1 roles) or all the roles (n roles). We consider these two cases in turn.

3.1.5.1 Internal Uniqueness Constraints on *n*-1 roles

Figure 3.5 shows the general case of a fact type with n roles where the internal uniqueness constraint spans n-1 roles ($n \geq 3$). Here, the object types (not necessarily distinct) are labelled $A_1..A_m$, and B, so $n = m+1$. The role that is not spanned by the uniqueness constraint is labelled i and may occur in any position ($1 \leq i \leq n$). This role is hosted by the object type B.

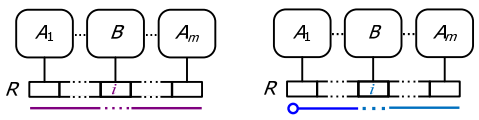


Figure 3.5 Internal uniqueness constraints on *n*-1 roles of an *n*-ary fact type

Table 3.10 shows the logical forms for the positive and negative verbalizations of the uniqueness constraints in Figure 3.5. Here and later, the symbol ▲ denotes a fixed noun classifer specifically used for a combination of entity types that the system will auto-retrieve and assign in Mandarin. For example, the quantification 'For each/some Student and Course' can be rendered as 对于每一/某一个'学生与课程' in Mandarin. In Bahasa Melayu, for some quantifiers 'sesuatu' is used instead of 'sese-' to represent the grouping of two or more entity types, thus no reserved noun classifer is needed (e.g. bagi setiap/sesuatu Pelajar danKursus).

In this table, the symbol 'Δ' denotes a user-picked, typed noun classifier that is used for common entities/non-grouping entities. For an independent entity type that doesn't involve a combination, the quantifier 'sese-Δ' (some) is used.

In BM, the nouns corresponding to 不能被量词所拘束的词 cannot be restrained by a classifier, so 'sesuatu' must be used instead of 'sese' for these.

Table 3.10 Logical forms of verbalizations of the uniqueness constraints in Figure 3.5

Sign	Modality	Language	Logical Form
+ve	Alethic	English	$\Box\forall x_1{:}A_1..x_m{:}A_m \quad \exists^{0..1} x_i{:}B\ Rx_1..x_n$
		BM	$\Box\forall x_1{:}A_1..x_m{:}A_m \quad \exists^{0..1}[\Delta]\ x_i{:}B\ Rx_1..x_n$
		Mandarin	$\forall \blacktriangle x_1{:}A_1..x_m{:}A_m \quad \exists^{0..1}\Delta\ x_i{:}B\ Rx_1..x_n\Box$
	Deontic	English	$\boldsymbol{O}\forall x_1{:}A_1..x_m{:}A_m \quad \exists^{0..1}\ x_i{:}B\ Rx_1..x_n$
		BM	$\boldsymbol{O}\ \forall x_1{:}A_1..x_m{:}A_m \quad \exists^{0..1}[\Delta]\ x_i{:}B\ Rx_1..x_n$
		Mandarin	$\forall \blacktriangle x_1{:}A_1..x_m{:}A_m \quad \exists^{0..1}\Delta\ x_i{:}B\ Rx_1..x_n\boldsymbol{O}$
-ve	Alethic	English	$\sim\Diamond\exists x_1{:}A_1..x_m{:}A_m \quad \exists^{2..}x_i{:}B\ Rx_1..x_n$
		BM	$\sim\Diamond\exists x_1{:}A_1..x_m{:}A_m \quad \exists^{2..}\ [\Delta]x_i{:}B\ Rx_1..x_n$
		Mandarin	$\exists \blacktriangle x_1{:}A_1..x_m{:}A_m \quad \exists^{2..}\Delta x_i{:}B\ Rx_1..x_n\sim\Diamond$
	Deontic	English	$\boldsymbol{F}\ \Diamond\exists x_1{:}A_1..x_m{:}A_m \quad \exists^{2..}x_i{:}B\ Rx_1..x_n$
		BM	$\boldsymbol{F}\ \Diamond\exists x_1{:}A_1..x_m{:}A_m \quad \exists^{2..}\ [\Delta]x_i{:}B\ Rx_1..x_n$
		Mandarin	$\exists \blacktriangle x_1{:}A_1..x_m{:}A_m \quad \exists^{2..}\Delta x_i{:}B\ Rx_1..x_n\boldsymbol{F}$

Figure 3.6 shows the case where an alethic uniqueness constraint spans the first *n*-1 roles of an *n*-ary fact type.

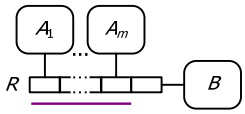


Figure 3.6 Alethic uniqueness constraint on first *n*-1 roles of an *n*-ary fact type

The verbalizations of this constraint are shown below, again using "*ft*" to indicate front text for those rare cases where some of the predicate text is placed before the first object term.

English

+ve: **For each** A_1, … **and** A_m, *ft* **that** A_1…**that** A_m *R* **at most one** *B*.

-ve: **It is impossible that for some** A_1, … **and** A_m, *ft* **that** A_1…**that** A_m *R* **at most one** *B*.

BM

+ve: **Bagi setiap** A_1, …**dan** A_m, *ft* A_1…**itu** A_m ***itu*** *R* **lebih daripada satu** [Δ] *B*.

-ve: **Ia adalah tidak mungkin bahawa bagi sesuatu** A_1, …***dan*** A_m, *ft* A_1…**itu** A_m ***itu*** *R* **lebih daripada satu** [Δ] *B*.

Mandarin

+ve: 对于每一▲ 'A_1, …与 A_m', *ft* 那一ΔA_1…那一ΔA_m*R* 多过一Δ*B*。

-ve: 对于某一▲ '*A1*, …与 A_m', *ft* 那一ΔA_1…那一ΔA_m*R* 多过一Δ*B* 是不可能的。

Some concerte examples of verbalizations of these patterns are shown in Table 3.11. Example 8 demonstrates the use of front text.

Table 3.11 Example verbalizations of uniqueness constraints spanning *n*-1 roles

No.	English	BM	Mandarin
1 +ve	**For each** Student **and** Course, **that** Student in **that** Course obtained **at most one** Rating.	**Bagi setiap** Pelajar **dan** Kursus, Pelajar **itu** di dalam Kursus **itu** memperolehi **paling banyak satu buah** Markah.	**对于每一个** ‘学生**与**课程’，**那一名**学生在**那一门**课程内获得**最多一个**分数。
2 +ve	**For each** Student **and** Course, **at most one** Rating was obtained by **that** Student in **that** Course.	**Bagi setiap** Pelajar **dan** Kursus, **paling banyak satu buah** Markah diperolehi oleh Pelajar **itu** di dalam Kursus **itu**.	**对于每一个**‘学生**与**课程’，**最多一个**分数在**那一门**课程内被**那一名**学生所获得。
3 +ve	**For each** Student **and** Course, **that** Student obtained **at most one** Rating in **that** Course.	**Bagi setiap** Pelajar **dan** Kursus, Pelajar **itu** memperolehi **paling banyak satu buah** Markah di dalam Kursus **itu**.	**对于每一个**‘学生**与**课程’，**那一名**学生获得**最多一个**分数在**那一门**课程内。
4 +ve	**For each** Player **and** Sport, **that** Player obtained **at most one** Ranking in **that** Sport.	**Bagi setiap** Pemain **dan** Sukan, Pemain **itu** memperolehi **paling banyak satu buah** Kedudukan di dalam Sukan **itu**.	**对于每一个**‘选手**与**运动项目’，**那一位**选手获得**最多一个**名次在**那一门**运动项目内。
5 -ve	**It is impossible that for some** Student **and** Course, **that** Student obtained **more than one** Rating in **that** Course.	**Ia adalah tidak mungkin bahawa bagi sesuatu** Pelajar **dan** Kursus, Pelajar **itu** memperolehi **lebih daripada satu buah** Markah di dalam Kursus **itu**.	**对于某一个**‘学生**与**课程’，**那一名**学生获得**多过一个**分数在**那一门**课程内**是不可能的**。
6 +ve	**For each** Person$_1$, Person$_2$ **and** Year, **that** Person$_1$ supervised **that** Person$_2$ in **that** Year for **at most one** Period.	**Bagi setiap** Orang$_1$ **dan** Orang$_2$ **dan** Tahun, Orang$_1$ **itu** meminpin Orang$_2$ **itu** dalam Tahun **itu** bagi **paling banyak satu jangka** TempohMasa.	**对于每一个**‘人$_1$**与**人$_2$**与**年份’，**那一个**人$_1$在**那一个**年份内监督**那一个**人$_2$被为**最多一个**时期。
7 -ve	**It is impossible that for some** Person$_1$, Person$_2$ **and** Year, **that** Person$_1$ supervised **that** Person$_2$ in **that** Year for **more than one** Period.	**Ia adalah tidak mungkin bagisesuatu** Orang$_1$ **dan** Orang$_2$**dan** Tahun, Orang$_1$ **itu** meminpin Orang$_2$ **itu** dalam Tahun **itu** bagi **lebih daripada satu jangka**TempohMasa.	**对于某一个**‘人$_1$**与**人$_2$**与**年份’，**那一个**人$_1$在**那一个**年份内监督**那一个**人$_2$为**多过一个**时期**是不可能的**。
8	**For each** Student **and**	**Bagi setiap**Pelajar **dan** Kursus,	**对于每一个**‘学生**与**课

+ve ***ft***	Course, in normal circumstances **that** Student in **that** Course obtained **at most one** Rating	dalam keadaan biasa Pelajar **itu** di dalam Kursus **itu** memperolehi **paling banyak satu buah** Markah.	程’，在一贯的情况下**那一**名学生在**那一**门课程内获得**最多一**个分数。
9 **+ve**	**It is obligatory that for each** Room **and** HourSlot, **that** Room at **that** HourSlot is booked for **at most one** Course.	**Ia adalah wajib bahawa bagi setiap** Bilik **dan** Masa, Bilik **itu** pada Masa **itu** ditempah bagi **paling banyak satu buah** Kursus.	**对于每一个**‘房**与**时段’，**那一**间房在**那一**个小时时段被预订给予**最多一**门课程**是强制性的**。
10 **-ve**	**It is forbidden that for some** Room **and** HourSlot, **that** Room at **that** HourSlot is booked for **more than one** Course.	**Ia adalah dilarang bahawa bagi sesuatu** Bilik **dan** Masa, Bilik **itu** pada Masa **itu** ditempah bagi **lebih daripada satu buah** Kursus.	**对于某一个**‘房**与**时段’，**那一**间房在**那一**个时段被预订给予**多过**一门课程**是被禁止的**。

Note: In Chinese, when the verbalization involves a combination of 2 or more object types, the noun classifier for ‘个’ followed by‘C1 与 C2 与 Cn’ is used instead of the singular noun classifier that is used to represent each individual object type. For example, ‘each Building, RoomNr’ in English is rendered as ‘每一个‘建筑物与房号’ while In BM, when the verbalization involves a combination of 2 or more object types, it is similar to the English verbalization (most of BM’s grammatical forms originated from English by default), since ‘and’ (in BM, ‘dan’) is understood to be used in a conjunctive sense, thus (*no noun classifier is required), so the pattern C1 **dan** C2 **dan …** Cn is used. Unlike Chinese, there is no need to add single quotes around the combination of terms. For example, ‘each Building and RoomNr’ in English is rendered as ‘Bangunan **dan** NoBilik’ in BM).

Spanning Internal Uniqueness Constraints on n-ary fact types

Figure 3.5 shows the general case of a spanning uniqueness constraint on a fact type with *n* roles.

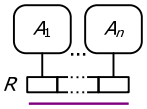


Figure 3.7 Spanning internal uniqueness constraint on an *n*-ary fact type

Table 3.12 shows the logical forms for the positive verbalizations of the uniqueness constraint in Figure 3.7. The negative verbalizations follow a similar pattern to that discussed earlier.

Table 3.12 Logical forms of positive verbalizations of the spanning uniqueness constraint in Figure 3.7

Sign	Modality	Language	Logical Form
+ve	Alethic	English	$\Diamond(\exists x_1{:}A_1..x_{n-1}{:}A_{n-1}\ \exists^{2..}x_n{:}A_n\ Rx_1..x_n$ & …& $\exists x_2{:}A_2..x_n{:}A_n\ \exists^{2..}x_1{:}A_1\ Rx_1..x_n)$
		BM	$\Diamond(\exists x_1{:}A_1..x_{n-1}{:}A_{n-1}\ \exists^{2..}[\Delta]x_n{:}A_n\ Rx_1..x_n$ & …& $\exists x_2{:}A_2..x_n{:}A_n\ \exists^{2..}[\Delta]x_1{:}A_1\ Rx_1..x_n)$
		Mandarin	$\Diamond(\exists\blacktriangle x_1{:}A_1..x_{n-1}{:}A_{n-1}\ \exists^{2..}\Delta x_n{:}A_n\ Rx_1..x_n$ & …& $\exists\blacktriangle x_2{:}A_2..x_n{:}A_n\ \exists^{2..}\Delta x_1{:}A_1\ Rx_1..x_n)$
	Deontic	English	$\boldsymbol{P}(\exists x_1{:}A_1..x_{n-1}{:}A_{n-1}\ \exists^{2..}x_n{:}A_n\ Rx_1..x_n$ & …& $\exists x_2{:}A_2..x_n{:}A_n\ \exists^{2..}x_1{:}A_1\ Rx_1..x_n)$
		BM	$\boldsymbol{P}(\exists x_1{:}A_1..x_{n-1}{:}A_{n-1}\ \exists^{2..}[\Delta]x_n{:}A_n\ Rx_1..x_n$ & …& $\exists x_2{:}A_2..x_n{:}A_n\ \exists^{2..}[\Delta]x_1{:}A_1\ Rx_1..x_n)$
		Mandarin	$\boldsymbol{P}(\exists\blacktriangle x_1{:}A_1..x_{n-1}{:}A_{n-1}\ \exists^{2..}\Delta x_n{:}A_n\ Rx_1..x_n$ & …& $\exists\blacktriangle x_2{:}A_2..x_n{:}A_n\ \exists^{2..}\Delta x_1{:}A_1\ Rx_1..x_n)$

The positive alethic patterns verbalize as follows**.**

English: **It is possible that for some** A_1, …**and** A_{n-1}, *R ft* **that** A_1…**that** A_2…**more than one** A_n **and that** …

…**and that for some** A_2, … **and** A_n, *R ft* **more than one** A_1 **that** A_2…**that** A_n.

BM: **Ia adalah mungkin bahawa bagi sesuatu ‘**A_1, … **dan** A_{n-1}’, *R ft* A_1…**itu** A_2 ***itu*…lebih daripada satu**[Δ] A_n

dan…

…**danbagi sesuatu gabungan dengan** A_2, … **dan** A_n, *R ft* ***lebih daripada satu***[Δ]A_1 A_2…**itu** A_n **itu**.

Mandarin: **对于某一▲** ‘A_1, …与 A_{n-1}’, *R ft* **那一**ΔA_1…**多过一**Δ A_n…**那一**ΔA_2

和…

…**和对于某一▲** ‘A_2, …与 A_n’, *R ft* **多过一**ΔA_1 **那一**ΔA_n…**那一**ΔA_2。

Table 3.13 shows an example of positive verbalizations of these alethic spanning uniqueness constraint patterns for a spanning uniqueness constraint on the ternary fact type Person played Sport for Country.

Table 3.13 Concrete examples of verbalizations of spanning uniqueness constraints

No.	English	BM	Mandarin
1	**It is possible that for some** Person **and** Sport, **that** Person played **that** Sport for **more than one** Country **and that for some** Person and Country, **that** Person played **more than one** Sport for **that** Country **and that for some** Sport **and** Country, **more than one** Person played **that** Sport for **that** Country.	**Ia adalah mungkin bahawa bagi sesuatu** Orang **dan** Sukan, Orang **itu** memainkan Sukan **itu** bagi **lebih daripada satu buah** Negara **dan bagi sesuatu** Orang **dan** Negara, Orang **itu** memainkan **lebih daripada satu bidang** Sukan bagi Negara **itu dan bagi sesuatu** Sukan **dan** Negara, **lebih daripada satu** Orang memainkan Sukan **itu** bagi Negara **itu**.	**对于某一个**‘人**与**体育’，**那一个**人为**多过一个**国家参与**那一项**体育**和对于某一个**‘人**与**国家’，**那一个**人为**那一个**国家参与**多过一项**体育**和对于某一个**‘体育**与**国家’，**多过一个**人为**那一个**国家参与**那一项**体育。

3.1.6 External Uniqueness Constraints

Figure 3.8 depicts an external uniqueness constraint spanning n roles, where $n \geq 2$. The object types $B_1 \ldots B_n$ are not necessarily distinct.

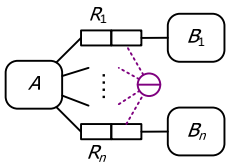


Figure 3.8 External uniqueness constraint spanning n roles ($n \geq 2$).

The formalizations of these constraints are shown in Table 3.14. The English formalizations were previously published in [7].

Table 3.14 Logical formalizations of the spanning uniqueness constraint in Figure 3.8

Sign	Modality	Language	Logical Form
+ve	Alethic	English	$\Box\forall y_1{:}B_1..y_n{:}B_n\ \exists^{0..1}\ x{:}A(xR_1y_1\ \&..\ xR_ny_n)$
		BM	$\Box\forall y_1{:}B_1..y_n{:}B_n\ \exists^{0..1}[\Delta]x{:}A(xR_1y_1\ \&..\ xR_ny_n)$
		Mandarin	$\forall\blacktriangle\ y_1{:}B_1..y_n{:}B_n\ \exists^{0..1}\Delta x{:}A(xR_1y_1\ \&..\ xR_ny_n)\Box$
	Deontic	English	$\boldsymbol{O}\forall y_1{:}B_1..y_n{:}B_n\ \exists^{0..1}\ x{:}A(xR_1y_1\ \&..\ xR_ny_n)$
		BM	$\boldsymbol{O}\forall y_1{:}B_1..y_n{:}B_n\ \exists^{0..1}[\Delta]x{:}A(xR_1y_1\ \&..\ xR_ny_n)$
		Mandarin	$\forall\blacktriangle\ y_1{:}B_1..y_n{:}B_n\ \exists^{0..1}\Delta x{:}A(xR_1y_1\ \&..\ xR_ny_n)\boldsymbol{O}$
-ve	Alethic	English	$\sim\Diamond\exists y_1{:}B_1..y_n{:}B_n\ \exists^{2..}\ x{:}A(xR_1y_1\ \&..\ xR_ny_n)$
		BM	$\sim\Diamond\exists y_1{:}B_1..y_n{:}B_n\ \exists^{2..}\ [\Delta]x{:}A(xR_1y_1\ \&..\ xR_ny_n)$
		Mandarin	$\exists\blacktriangle\ y_1{:}B_1..y_n{:}B_n\ \exists^{2..}\Delta x{:}A(xR_1y_1\ \&..\ xR_ny_n)\sim\Diamond$
	Deontic	English	$\boldsymbol{F}\ \exists y_1{:}B_1..y_n{:}B_n\ \exists^{2..}\ x{:}A(xR_1y_1\ \&..\ xR_ny_n)$
		BM	$\boldsymbol{F}\ \exists y_1{:}B_1..y_n{:}B_n\ \exists^{2..}\ [\Delta]x{:}A(xR_1y_1\ \&..\ xR_ny_n)$
		Mandarin	$\exists\blacktriangle\ y_1{:}B_1..y_n{:}B_n\ \exists^{2..}\Delta x{:}A(xR_1y_1\ \&..\ xR_ny_n)\ \boldsymbol{F}$

The alethic patterns verbalize as follows**.**

English:

+ve: **For each** B_1**,… and** B_n **at most one** A R_1 **that** B_1 **and … and** R_n **that** B_n.

-ve: **It is impossible that more than one** A R_1 **the same** B_1 **and** … R_n **the same** B_n.

BM:

+ve: **Setiap** B_1, …***dan*** B_n **paling banyak satu**[Δ]A R_1 B_1 **itu dan**…**dan** R_n B_n **itu.**

-ve: **Ia adalah tidak mungkin bahawa lebih daripada satu** [Δ] A B_1 **yang sama dan**…R_n B_n **yang sama**.

Mandarin:

+ve: **对于每一▲‘**B_1, …**与** B_n**’最多一**ΔAR_1**那一**ΔB_1**和**…**和** R_n**那一**ΔB_n

-ve: **多过一**ΔAR_2**同一**ΔB_1**和** R_n**同一**ΔB_n。

Table 3.15 Concrete examples of verbalizations of external uniqueness constraints

No.	English	BM	Mandarin
1 +ve	**For each** Building **and** RoomNr, **at most one** Room is in **that** Building and has **that** RoomNr.	**Bagi setiap** Bangunan **dan** NoBilik, **paling banyak satu buah** Bilik terletak di Bangunan **itu** dan mempunyai NoBilik **itu**.	**每一个**‘建筑物**与**房号’，**最多一间**房坐落于**那一栋**建筑物和拥有**那一个**房号。
2 -ve	**It is impossible that more than one** Room is in **the same** Building and has **the same** RoonNr.	**Ia adalah tidak mungkin bahawa lebih daripada satu buah** Bilik terletak di Bangunan **yang sama** dan mempunyai NoBilik **yang sama**.	**多过一间**房坐落于**同一栋**建筑物和拥有**同一个**房号**是不可能的**。
3 +ve	**For each** Node$_1$ **and** Node$_2$, **at most one** Arrow is from Node$_1$ **and** is to Node$_2$.	**Bagi setiap** TitikJalinan$_1$ **dan** TitikJalinan$_2$, **paling banyak satu batang** AnahPanah berasal dari TitikJalinan$_1$ **dan** menunjuk ke Titik Jalinan$_2$.	**对于每一个**‘连接点 $_1$ **与**连接点 $_2$’，**最多一个**箭头从连接点 $_1$ **和**指向连接点 $_2$。
4 -ve	**It is impossible that more than one** Arrow is from the **same** Node$_1$ **and** is to **the same** Node$_2$.	**Ia adalah tidak mungkin bahawa lebih daripada satu batang** AnahPanah berasal dari TitikJalinan$_1$ **yang sama dan** menunjuk ke TitikJalinan$_2$ **yang sama**.	**多过一个**箭头从**同一个**连接点 $_1$ 指向**同一个**连接点 $_2$ **是不可能的**。
5 +ve	**For each** Building **and** RoomNr, **there is at most one** Room **such that** the location of **that** Room is in **that** Building **and that** RoomNr is of **that** Room.	**Bagi setiap** Bangunan **dan** NoBilik, **terdapat paling banyak satu buah** Bilik **di mana** Bilik **itu** terletak di Bangunan **itu** dan NoBilik **itu** ialah kepunyaan Bilik **itu**.	**对于每一个**‘建筑物**与**房号’，**其拥有最多一间**房**为此那一间**房坐落于**那一栋**建筑物**和那一个**房号是**那一间**房所

			属。
6 -ve	**It is impossible that for some** Building **and** RoomNr, **there is more than one** Room **such that** the location of **that** Room is in **that** Building and **that** RoomNr is of **that** Room.	**Ia adalah tidak mungkin bahawa bagi sesuatu** Bangunan **dan** NoBilik, **terdapat lebih daripada satu buah** Bilik **di mana** kedudukan Bilik **itu** terletak di Bangunan **itu** dan NoBilik **itu** ialah kepunyaan Bilik **itu**.	**对于每一个**‘建筑物**与**房号’，**其拥有多过一间**房**为此那一间**房坐落于**那一栋**建筑物**和那一个**房号是**那一间**房所属是**不可能的**。
7 -ve	**It is impossible that for some** Driver **and** CarPlateNo, **there is more than one** Car **such that that** Car is owned by **that** Driver and **that** CarPlateNo is of **that** Car.	**Ia adalah tidak mungkin bahawa bagi sesuatu** Pemandu **dan** PlatNomborKereta, **terdapat lebih daripada satu buah** Kereta **di mana** Kereta **itu** kepunyaan Pemandu **itu** dan PlatNomborKereta **itu** ialah kepunyaan Kereta **itu**.	**对于每一个**‘驾驶者**与**汽车号牌’，**其拥有多过一个**汽车**为此那一辆**汽车是**那一名**驾驶者所属**和那一个**汽车号牌是**那一辆**汽车所属是**不可能的**。

3.1.7 Simple Mandatory Role Constraints

3.1.7.1 Unary Predicates

Figure 3.9 depicts simple mandatory role constraints on a unary fact type, along with their English verbalizations. In case (a) the constraint is alethic, and in case (b) the constraint is deontic. Front text is denoted by *ft*.

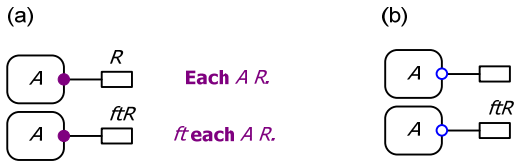


Figure 3.9 Simple mandatory role constraints on a unary fact type

Table 3.16 shows the logical forms of the positive verbalizations (ignoring front text) of the mandatory role constraints in Figure 3.9.

Table 3.16 Logical formalizations of the mandatory role constraints in Figure 3.9

Sign	Modality	English Logical Form	BM Logical Form	Mandarin Logical Form
+ve	Alethic	□∀*x:A.Rx*	□∀*x:A.Rx*	∀Δ*x:A.Rx*□
	Deontic	***O*** ∀*x:A.Rx*	***O***∀*x:A.Rx*	∀Δ*x:A.Rx****O***

These patterns verbalize as shown below.

Alethic:

Each *A R.*

Setiap *A R.*

每一Δ*A R*。

ft **each** *A R.*

ft **Setiap** *A R.*

ft **每一**Δ*A R*。

Deontic:

It is obligatory that each*A R.*

Ia adalah wajib bahwa setiap*A R.*

每一Δ*AR* **是强制性的**。

It is obligatory that *ft* **each***A R.*

Ia adalah wajib bahwa *ft* **setiap***A R.*

ft **每一**Δ*AR* **是强制性的**。

Table 3.17 Example verbalizations of simple mandatory role constraints on unaries

No.	English	BM	Mandarin
1	**Each** Square is rectangular in shape	**Setiap** empat segi ialah berbentuk empat tepat panjang .	**每一**个四方体的形状是长方形。
2	**It is obligatory that each** Doctor is licensed.	**Ia adalah wajib bahawa setiap** Doktor adalah diisukan license.	**每一**位医生是被批准有执照的是强制性的。
3	Once a week **each** Employee has a day off.	Sekali seminggu **setiap** pekerja mempunyai satu hari rehat.	每一星期一次**每一**位员工拥有一天假日。
4	**Each** Dog is a kind of animal.	**Setiap ekor** Anjing ialah sejenis binatang.	**每一**只狗是种动物。

3.1.7.2 Binary Predicates

Figure 3.10 depicts simple mandatory role constraints on a binary fact type, along with their English verbalizations.

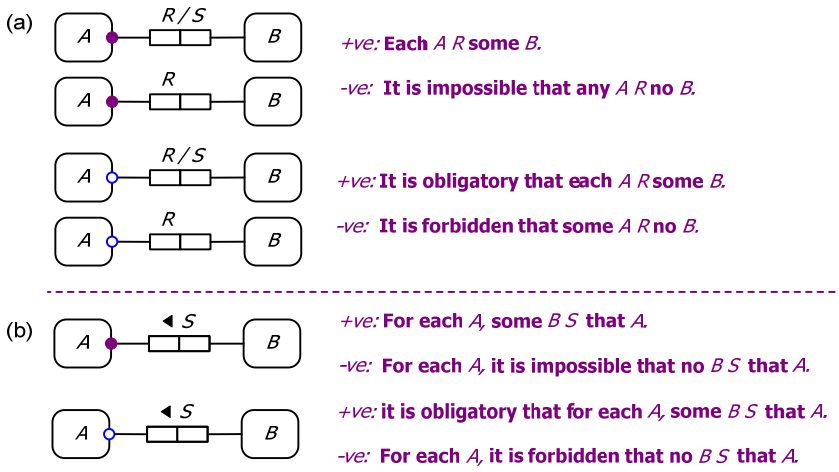


Figure 3.10 Simple mandatory role constraints on a binary fact type

Table 3.18 shows the logical formalizations of the simple mandatory role constraints in Figure 3.10.

Table 3.18 Logical formalizations of the mandatory role constraints in Figure 3.10

Case	Sign	Modality	Language	Logical Form
(a)	+ve	Alethic	English BM Mandarin	$\Box\forall x{:}A\ \exists y{:}B\ xRy$ $\Box\forall x{:}A\exists[\Delta]y{:}B\ xRy$ $\forall\Delta x{:}A\ \exists\Delta y{:}B\ xRy\Box$
		Deontic	English BM Mandarin	$\boldsymbol{O}\forall\ x{:}A\ \exists y{:}B\ xRy$ $\boldsymbol{O}\forall x{:}A\exists[\Delta]y{:}B\ xRy$ $\forall\Delta x{:}A\ \exists\Delta y{:}B\ xRy\boldsymbol{O}$
	-ve	Alethic	English BM Mandarin	$\sim\Diamond\exists x{:}A\sim\exists y{:}B\ xRy$ $\sim\Diamond\exists[\Delta]x{:}A\sim\exists[\Delta]y{:}B\ xRy$ $\exists\Delta x{:}A\sim\exists\Delta y{:}B\ xRy\ \sim\Diamond$
		Deontic	English BM Mandarin	$\boldsymbol{F}\exists x{:}A\sim\exists y{:}B\ xRy$ $\boldsymbol{F}\exists[\Delta]x{:}A\sim\exists[\Delta]y{:}B\ xRy$ $\exists\Delta x{:}A\sim\exists\Delta y{:}B\ xRy\boldsymbol{F}$

(b)	+ve	Alethic	English BM Mandarin	□∀*x:A* ∃*y:B ySx* □∀*x:A*∃[Δ]*y:B ySx* ∀Δ*x:A* ∃Δ*y:B ySx*□
		Deontic	English BM Mandarin	***O***∀*x:A*∃[Δ]*y:B ySx* ***O***∀*x:A*∃[Δ]*y:B ySx* ∀Δ*x:A* ∃Δ*y:B ySx* ***O***
	-ve	Alethic	English BM Mandarin	~◊∃*x:A*~∃*y:B ySx* ~◊∃[Δ]*x:A*~∃[Δ]*y:B ySx* ∃Δ*x:A*~∃Δ*y:B ySx*~◊
		Deontic	English BM Mandarin	***F***∃*x:A*~∃*y:B ySx* ***F***∃[Δ]*x:A*~∃[Δ]*y:B ySx* ∃Δ*x:A*~∃Δ*y:B ySx****F***

Table 3.19 Example verbalizations of simple mandatory role constraints on binaries

No.	English	BM	Mandarin
1	**Each** Person was born in **some** Year.	**Setiap** Orang dilahirkan dalam **sesuatu** Tahun.	**每一**个人出生于**某一**个年份。
2	**It is impossible** that **some** Person was born in **no** Year.	**Ia adalah tidak mungkin bahawa sese** Orang tidak dilahirkan **dalam mana-mana satu** Tahun.	**某一**个人出生于**无任何一**个年份**是不可能的**。
3	**It is obligatory that each** Immigrant has **some** Passport.	**Ia adalah wajib bahawa setiap** Immigrasi mempunyai **sesebuku** Passport.	**每一**位移民拥有**某一**本护照**是强制性的**。
4	**It is forbidden that some** Immigrant has **no** Passport.	**Ia adalah dilarang bahawa seseorang** Immigrasi mempunyai **tiada mana-mana satu buku** passport.	**某一**位移民拥有**无任何一**本护照**是被禁止的**。

3.1.7.3 *n*-ary Predicates: mandatory role starts a predicate reading

Figure 3.11 depicts a simple mandatory role constraint on an *n*-ary fact type ($n \geq 3$), with a predicate reading *R* starting at that role, where *ft* is front text (if present). The English verbalizations are also shown. The object types A_1 .. A_n may be entity types or value types, and may or may not be distinct.

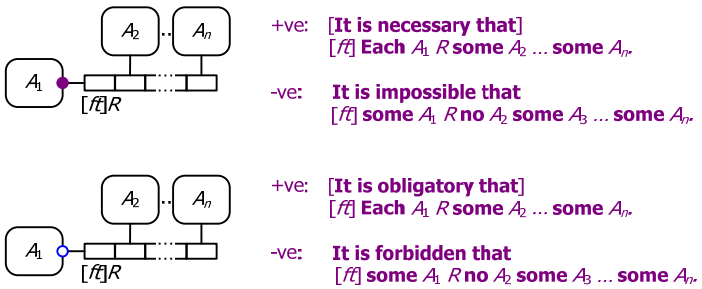


Figure 3.11 Leading mandatory role constraints on an *n*-ary fact type

Table 3.20 shows the logical forms of the verbalizations (ignoring front text) of the mandatory role constraints in Figure 3.11.

Table 3.20 Logical formalizations of the mandatory role constraints in Figure 3.11

Sign	Modality	English Logical Form	BM Logical Form	Mandarin Logical Form
+ve	Alethic	$\Box\forall x{:}A_1\exists x_2{:}A_2..\ x_n{:}A_n$ $Rx_1..x_n$	$\Box\forall x{:}A_1\exists[\Delta]x_2{:}A_2..\ x_n{:}A_n$ $Rx_1..x_n$	$\forall\Delta x{:}A_1\exists\Delta x_2{:}A_2..\ x_n{:}A_n$ $Rx_1..x_n\Box$
	Deontic	$\boldsymbol{O}\forall x{:}A_1\exists x_2{:}A_2..\ x_n{:}A_n$ $Rx_1..x_n$	$\boldsymbol{O}\forall x{:}A_1\exists[\Delta]x_2{:}A_2..\ x_n{:}A_n$ $Rx_1..x_n$	$\forall\Delta x{:}A_1\exists\Delta x_2{:}A_2..\ x_n{:}A_n$ $Rx_1..x_n\ \boldsymbol{O}$
-ve	Alethic	$\sim\Diamond\exists x{:}A_1\sim\exists x_2{:}A_2..\ x_n{:}A_n$ $Rx_1..x_n$	$\sim\Diamond\exists[\Delta]x{:}A_1\sim\exists[\Delta]x_2{:}A_2..$ $x_n{:}A_n\ Rx_1..x_n$	$\exists\Delta x{:}A_1\sim\exists\Delta x_2{:}A_2..\ x_n{:}A_n$ $Rx1..xn\sim\Diamond$
	Deontic	$\boldsymbol{F}\exists x{:}A_1\sim\exists x_2{:}A_2..\ x_n{:}A_n$ $Rx_1..x_n$	$\boldsymbol{F}\exists[\Delta]x{:}A_1\sim\exists[\Delta]x_2{:}A_2..\ x_n{:}A_n$ $Rx_1..x_n$	$\exists\Delta x{:}A_1\sim\exists\Delta x_2{:}A_2..\ x_n{:}A_n$ $Rx_1..x_n\boldsymbol{F}$

The alethic patterns verbalize as follows.

English:

+ve: [**It is necessary that**] [*ft*] **Each** A_1 *R* **some** A_2…**some** A_n.

-ve: **It is impossible that** [*ft*] **some**A_1 *R* **no** A_2 **some** A_3…**some** A_n.

BM:

+ve: [**Ia adalah perlu bahawa**][*ft*] **Setiap** A_1 *R* **sese**Δ**/sesuatu** A_2…**sese**Δ**/sesuatu** A_n.

-ve: **Ia adalah tidak mungkin bahawa** [*ft*] **sese**Δ**/sesuatu** A_1 **tidak** *R* **mana-mana satu**[Δ] A_2 **sese**Δ**/sesuatu** A_3…**sese**Δ**/sesuatu** A_n.

Mandarin:

+ve: [*ft*]**每一**Δ*A1R* **某一**Δ*A2*…**某一**Δ*An*[**是必要的**]。

-ve: [*ft*]**某一**Δ*A1* **没** *R* **任何一**Δ**某一**Δ *A3*…**某一**Δ*An* **的** *A2* **是不可能的**。

The deontic patterns verbalize as follows.

English:

+ve: **It is obligatory that** [*ft*] **Each** A_1 *R* **some** A_2…**some** A_n.

-ve: **It is forbidden that** [*ft*] **some**A_1 *R* **no** A_2 **some** A_3…**some** A_n.

BM:

+ve: **Ia adalah wajib bahawa**[*ft*] **Setiap** A_1 *R* **sese**Δ**/sesuatu** A_2…**sese**Δ**/sesuatu** A_n.

-ve: **Ia adalah dilarang bahawa** [*ft*] **sese**Δ**/sesuatu** A_1 *R* **tiada mana-mana satu**[Δ] A_2 **sese**Δ**/sesuatu** A_3…**sese**Δ**/sesuatu** A_n.

Mandarin:

+ve: [*ft*]**每一**Δ*A1R* **某一**Δ*A2*…**某一**Δ*An*[**是强制性的**]。

-ve: [*ft*]**某一**Δ*A1R* **无任何一**Δ**某一**Δ *A3*…**某一**Δ*An* **的** *A2* **被禁止的**。

Table 3.21 Example verbalizations of leading mandatory role constraints on *n*-aries

No.	English	BM	Mandarin
1	**Each** Programmer codes in **some** Programming Language at **some** SkillLevel.	**Setiap** Programmer memprogram dalam **sesebuah** BahasaProgramming pada **sesebuah** SkillTahap.	**每一**位软件编写员编写于**某一**种编写语言在于**某一**个技术程度。
2	**It is obligatory that each** Translator knows **some** ForeignLanguage at **some** SkillLevel.	**Ia adalah wajib bahawa setiap** AhliPenterjemah mengetahui **sesebuah** BahasaAsing pada **sesebuah** SkillTahap.	**每一**位翻译员知晓**某一**种外国语语言在于**某一**个技术程度**是强制性的**。
3	When hungry **each** Person eats **some** Food cooked by **some** Person.	Apabila lapar **setiap** Orang makan **sesejenis** Makanan yang dimasak oleh **sese**Orang.	当饥饿时**每一**个人享用**某一**样被**某一**个人烹饪的食物。

4	**It is impossible that some** Person eats **no** Food cooked by **some** Person.	**Ia adalah tidak mungkin bahawa sese**Orang makan **tiada mana-mana satu jenis** makanan yang dimasak oleh **sese** Orang.	**某一个**人享用**无任何一样**被**某一个**人烹饪的食物**是不可能的**。
5	**Each** Candidate in **some** CarTest obtained **some** Score.	**Setiap** Calon dalam **sesebuah** UjianKereta mendapat **sesebuah** Markah.	**每一位**考生在**某一项**考车测验里考获**某一个**评分。

3.1.7.4 *n*-ary Predicates: mandatory role does not start a predicate reading

Figure 3.12 depicts a mandatory on role i of an n-ary fact type where $1 \leq i \leq n$ and the role does not start a predicate reading. The English verblaizations are also shown.

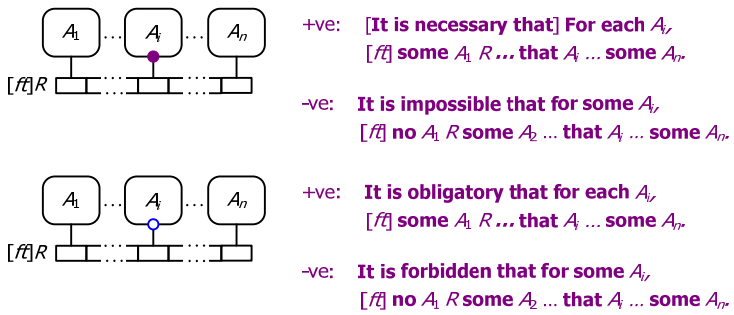


Figure 3.12 Mandatory role constraints on a role of an *n*-ary fact type, where the role does not start a predicate reading

Table 3.22 shows the logical formalizations.

Table 3.22 Logical formalizations of the mandatory role constraints in Figure 3.12

Sign	Modality	Language	Logical Form
+ve	Alethic	English BM Mandarin	$\Box\ \forall x{:}A_i\ \exists x_1{:}A_1..x_{i-1}{:}A_{i-1}\ x_{i+1}{:}A_{i+1}..x_n{:}A_n\ Rx_1..x_n$ $\Box\ \forall x{:}A_i\ \exists[\Delta]x_1{:}A_1..x_{i-1}{:}A_{i-1}\ x_{i+1}{:}A_{i+1}..x_n{:}A_n\ Rx_1..x_n$ $\forall\Delta x{:}A_i\exists\Delta\ x_1{:}A_1..x_{i-1}{:}A_{i-1}\ x_{i+1}{:}A_{i+1}..x_n{:}A_n\ Rx_1..x_n\ \Box$
	Deontic	English BM Mandarin	$\boldsymbol{O}\forall x{:}A_i\ \exists x_1{:}A_1..x_{i-1}{:}A_{i-1}\ x_{i+1}{:}A_{i+1}..x_n{:}A_n\ Rx_1..x_n\ \boldsymbol{O}\forall x{:}A_i$ $\boldsymbol{O}\forall x{:}A_i\ \exists[\Delta]x_1{:}A_1..x_{i-1}{:}A_{i-1}\ x_{i+1}{:}A_{i+1}..x_n{:}A_n\ Rx_1..x_n$ $\forall\Delta x{:}A_i\exists\Delta\ x_1{:}A_1..x_{i-1}{:}A_{i-1}\ x_{i+1}{:}A_{i+1}..x_n{:}A_n\ Rx_1..x_n\ \boldsymbol{O}$
-ve	Alethic	English BM Mandarin	$\sim\Diamond\exists x{:}A_i\ \sim\exists x_1{:}A_1..x_{i-1}{:}A_{i-1}\ x_{i+1}{:}A_{i+1}..x_n{:}A_n\ Rx_1..x_n$ $\sim\Diamond\exists[\Delta]x{:}A_i\sim\exists[\Delta]x_1{:}A_1..x_i{-}1{:}A_{i-1}\ x_{i+1}{:}A_{i+1}..\ x_n{:}A_n\ \ Rx_1..x_n$ $\exists\Delta x{:}A_i\sim\exists\Delta x_1{:}A_1..x_{i-1}{:}A_{i-1}\ x_{i+1}{:}A_{i+1}..\ x_n{:}A_n\ \ Rx_1..x_n\sim\Diamond$
	Deontic	English BM Mandarin	$\boldsymbol{F}\ \exists x{:}A_i\ \sim\exists x_1{:}A_1..x_{i-1}{:}A_{i-1}\ x_{i+1}{:}A_{i+1}..x_n{:}A_n\ Rx_1..x_n$ $\boldsymbol{F}\ \exists[\Delta]x{:}A_i\sim\exists[\Delta]x_1{:}A_1..x_i{-}1{:}A_{i-1}\ x_{i+1}{:}A_{i+1}..\ x_n{:}A_n\ \ Rx_1..x_n$ $\exists\Delta x{:}A_i\sim\exists\Delta x_1{:}A_1..x_{i-1}{:}A_{i-1}\ x_{i+1}{:}A_{i+1}..\ x_n{:}A_n\ \ Rx_1..x_n\ \boldsymbol{F}$

The alethic patterns verbalize as follows.

English:

+ve: [**It is necessary that**] **For Each** A_i, [*ft*] **some** A_1 R … **that** A_i…**some** A_n.

-ve: **It is impossible that for some** A_i, [*ft*] **no** A_1 R **some** A_2… that A_i…**some** A_n.

BM:

+ve: [**Ia adalah perlu bahawa**][*ft*] **Setiap** A_1 R **sese**Δ**/sesuatu** A_2…**sese**Δ**/sesuatu** A_n.

-ve: **Ia adalah tidak mungkin bahawa bagi sese**Δ**/sesuatu** A_i,[*ft*] **tiada mana-mana satu**[Δ] A_1 R **sese**Δ**/sesuatu** A_2…A_i **itu**…**sese**Δ**/sesuatu** A_n.

Mandarin:

+ve: **对于每一**ΔA_1, [*ft*]**某一**Δ$A_1$$R$…**那一**Δ$A_i$…**某一**Δ$A_n$[**是必要的**]。

-ve: **对于某一**ΔA_i,[*ft*]**没任何一**Δ$A_1$$R$ **某一**Δ A_2…**那一**ΔA_i…**某一**ΔA_n **是不可能的**。

The deontic patterns verbalize as follows.

English:

+ve: **It is obligatory that** for **each** A_i, [*ft*] **some** A_1 R … that A_i…**some** A_n

-ve: **It is forbidden that for some** A_i, [*ft*] **no** A_1 R **some** A_2… **that** A_i…**some** A_n.

BM:

+ve: **Ia adalah wajib bahawa Bagi Setiap** A_i, [*ft*] **sese**Δ**/sesuatu** A_1 R … A_i… itu **sese**Δ A_n.

-ve: **Ia adalah dilarang bahawa bagi sese**Δ**/sesuatu** A_i ,[*ft*] **tiada mana-mana satu**[Δ] A_1 *R* **sese** Δ**/sesuatu** A_2…A_i **itu**…**sese**Δ**/sesuatu** A_n.

Mandarin:

+ve: **对于每一**ΔA_1, [*ft*]**某一**ΔA_1*R*…**那一**ΔA_i…**某一**ΔA_n[**是强制性的**]。

-ve: **对于某一**ΔA_i,[*ft*]**无任何一**ΔA_1*R* **某一**Δ A_2…**那一**ΔA_i…**某一**ΔA_n **是被禁止的**。

Table 3.23 Example verbalizations of non-leading mandatory role constraints on binaries

No.	English	BM	Mandarin
1	**For each** Programmer, **some** ProgrammingLanguage is coded in by **that** Programmer at **some** SkillLevel.	**Bagi setiap** Programmer, **sese buah** BahasaProgramming diprogram oleh Programmer **itu** pada **sesuatu** TahapSkill.	**对于每一**位软件编写员，**某一种**编写语言被**那一**位软件编写员所编写于**某一**个技术程度。
2	**For each** Candidate, **some** CarTest is taken by **that** Candidate at **some** Time.	**Bagi setiap** Calon, **sesebuah** UjianKereta diambil oleh Calon **itu** pada **sesuatu** Masa.	**对于每一**位考生，**某一**项考车测验被**那一**位考生所拿于**某一**个时间。
3	**It is obligatory that for each** Translator, **some** ForeignLanguage is known by **that** Translator at **some** SkillLevel.	**Ia adalah wajib bahawa bagi setiap** AhliPenterjemah, **sesebuah** BahasaAsing dikenali oleh AhliPenterjemah itu pada **sesuatu** TahapSkill.	**对于某一**位翻译员，**某一种**外国语言被**那一**位翻译员所知晓于**某一**个技术程度**是强制性的**。
4	**For each** Person$_1$, when ready **some** Food cooked by **some** Person is eaten by **that** Person$_1$.	**Bagi setiap** Orang$_1$, apabila bersedia **sesejenis**Makanan yang dimasak oleh **sese**Orang telah dimakan oleh Orang$_1$**itu**.	**对于某一个**人 $_1$，当准备好**某一样**被**某一个**人烹饪的食物已被**那一个**人 $_1$吃了。
5	**It is impossible that for some** Programmer, **no** ProgrammingLanguage is coded in by **that** Programmer at **some** SkillLevel.	**Ia adalah tidak mungkin bagi sese**Orang Progarmmer, **tiada mana-mana satu buah** BahasaProgramming diprogram oleh Programmer **itu** pada **sesuatu** TahapSkill.	**对于某一位**软件编写员，**无任何一种**编写语言是被**那一位**软件编写员所编写于**某一个**技术程度**是不可能的**。

3.1.8 Disjunctive Mandatory Role Constraints

A disjunctive mandatory role constaint applies to a logical disjunction of two or more roles, and is also known as an inclusive-or constraint. Only positive verbalizations are used for this case because their negative versions are awkward.

3.1.8.1 Each constrained role starts a predicate reading with no front text

Figure 3.13 shows the English positive, alethic pattern used by the NORMA tool, as specified in [16]. The object types are not necessarily distinct. Each predicate R_i may have *ni* roles ($ni \geq 0$) following the role played by *A*. So the predicates may all be of different arity (unary upwards), and the object types may or may not be the same. Non-unaries are verbalized before before unaries. The *deontic* version prepends "it is obligatory that" to the positive form.

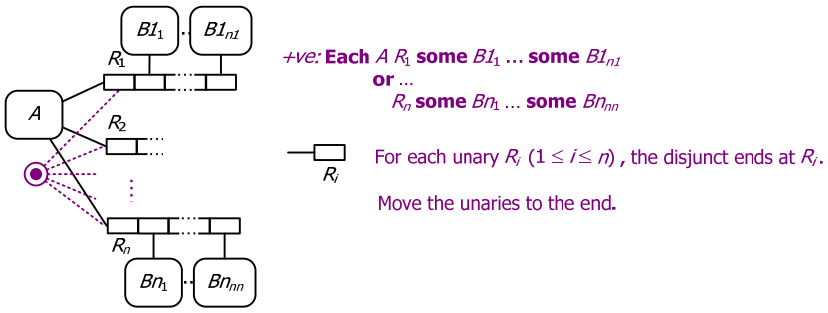


Figure 3.13 Disjunctive mandatory role constraints where each constrained role starts a predicate reading with no front text.

Table 3.24 shows the logical formalizations.

Table 3.24 Formalizations of the disjunctive mandatory role constraints in Figure 3.13

Sign	Modality	Language	Logical Form
+ve	Alethic	English	$\Box\ \forall x{:}A\ (\exists x_1{:}B1_1..x_{n1}{:}B1_{n1}\ Rxx_1..x_{n1} \vee \ldots \vee \exists x_1{:}Bn_1..x_{nn}{:}Bn_{nn}\ Rxx_1..x_{nn})$
		BM	$\Box\ \forall x{:}A\ (\exists[\Delta]x_1{:}B1_1..x_{n1}{:}B1_{n1}\ Rxx_1..x_{n1} \vee \ldots \vee \exists[\Delta]x_1{:}Bn_1..x_{nn}{:}Bn_{nn}\ Rxx_1..x_{nn})$
		Mandarin	$\forall\Delta x{:}A\ (\exists\Delta x_1{:}B1_1..x_{n1}{:}B1_{n1}\ Rxx_1..x_{n1} \vee \ldots \vee \exists\Delta x_1{:}Bn_1..x_{nn}{:}Bn_{nn}\ Rxx_1..x_{nn})\ \Box$
	Deontic	English	$\boldsymbol{O}\ \forall x{:}A\ (\exists x_1{:}B1_1..x_{n1}{:}B1_{n1}\ Rxx_1..x_{n1} \vee \ldots \vee \exists x_1{:}Bn_1..x_{nn}{:}Bn_{nn}\ Rxx_1..x_{nn})$
		BM	$\boldsymbol{O}\ \forall x{:}A\ (\exists[\Delta]x_1{:}B1_1..x_{n1}{:}B1_{n1}\ Rxx_1..x_{n1} \vee \ldots \vee \exists[\Delta]x_1{:}Bn_1..x_{nn}{:}Bn_{nn}\ Rxx_1..x_{nn})$
		Mandarin	$\forall\Delta x{:}A\ (\exists\Delta x_1{:}B1_1..x_{n1}{:}B1_{n1}\ Rxx_1..x_{n1} \vee \ldots \vee \exists\Delta x_1{:}Bn_1..x_{nn}{:}Bn_{nn}\ Rxx_1..x_{nn})\ \boldsymbol{O}$

The positive alethic patterns verbalize in English, BM and Mandarin respectively as follows.

Each *A* R_1 **some** $B1_1$…**some** $B1_{n1}$ **or**… R_n **some** Bn_1…**some** Bn_{nn}.

Setiap *A* R_1 **sese**Δ**/sesuatu** $B1_1$…**sese**Δ**/sesuatu** $B1_{n1}$ **atau**…R_n **sese**Δ**/sesuatu** Bn_1…**sese** Δ**/sesuatu** Bn_{nn}

每一Δ*A*R_1**某一**Δ$B1_1$…**某一**Δ$B1_{n1}$**或**…R_n**某一**ΔBn_1…**某一**ΔBn_{nn}

Table 3.25 Example verbalizations of disjunctive mandatory role constraints where each constrained role starts a predicate reading with no front text.

No.	English	BM	Mandarin
1	**Each** Partner became the husband of **some** Partner on **some** Date **or** became the wife of **some** Partner on **some** Date.	**Setiap** Pasangan menjadi suami bagi **seseorang** Pasangan pada **sesuatu** Tarikh **atau** menjadi isteri bagi **seseorang** Pasangan pada **sesuatu**Tarikh.	**每一**位侣伴在**某一**个日期成为**某一**位侣伴的丈夫**或**在**某一**个日期成为**某一**位侣伴的妻子。
2	**It is obligatory that each** Vehicle was purchased from **some** AutoRetailer **or** is rented.	**Ia adalah wajib bahawa setiap** Kenderaan dibeli oleh **seseorang** Pembeli **atau** disewa.	**某一**辆交通工具被**某一**位买家购买或被租用**是强制性的**。

3	**It is obligatory that each** Person was contracted until **some** Date **or** is a full time worker.	**Ia adalah wajib bahawa setiap** Orang dikontrakkan sehingga sesuatu Tarikh atau **seseorang** Pembeli **atau** ialah Pekerja Sepenuh Masa.	某一个人被签约至某一个日期或是全职员工**是强制性的**。

3.1.8.2 At least one constrained role has front text and/or does not start a predicate reading

Figure 3.14 shows the English positive, alethic pattern used by the NORMA tool, as specified in [16]. If *A* hosts more than one role in at least one of the R_i predicates, we subscript its instances where needed to distinguish them. Each predicate R_i may have *ni* roles ($ni \geq 0$) plus the constrained role played by *A*. If at least two of *A*'s constrained roles appear in the same predicate (e.g. $R_1 = R_2$) we distinguish them by subscripting unless preceded by "some" in the verbalization. Non-unaries are verbalized before unaries. The deontic version prepends "It is obligatory that" to the positive form.

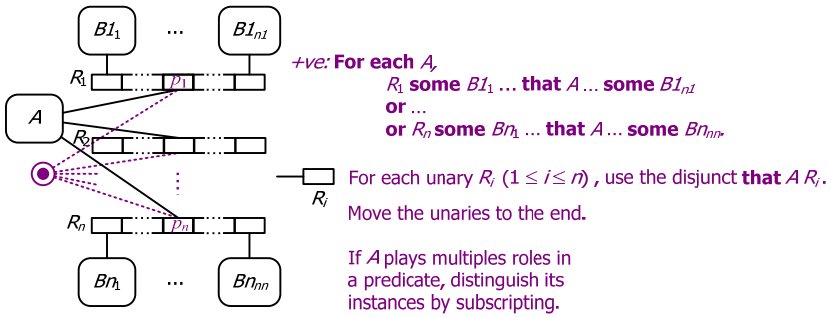


Figure 3.14 Disjunctive mandatory role constraints where at least one constrained role has front text and/or does not start a predicate reading.

Table 3.26 Formalizations of the disjunctive mandatory role constraints in Figure 3.14

Sign	Modality	Language	Logical Form
+ve	Alethic	English	$\Box \forall x{:}A\ (\exists x_1{:}B1_1..\ x_{n1}{:}B1_{n1}\ Rx_1..x_{p1-1}xx_{p1+1}..x_{n1} \vee \ldots \vee \exists x_1{:}Bn_1..x_{nn}{:}Bn_{nn}\ Rx_1..x_{pn-1}xx_{pn+1}..x_{nn})$
		BM	$\Box \forall x{:}A\ (\exists[\Delta]x_1{:}B1_1..\ x_{n1}{:}B1_{n1}\ Rx_1..x_{p1-1}xx_{p1+1}..x_{n1} \vee \ldots \vee \exists[\Delta]x_1{:}Bn_1..x_{nn}{:}Bn_{nn}\ Rx_1..x_{pn-1}xx_{pn+1}..x_{nn})$
		Mandarin	$\forall\Delta x{:}A\ (\exists\Delta x_1{:}B1_1..\ x_{n1}{:}B1_{n1}\ Rx_1..x_{p1-1}xx_{p1+1}..x_{n1} \vee \ldots \vee \exists\Delta x_1{:}Bn_1..x_{nn}{:}Bn_{nn}\ Rx_1..x_{pn-1}xx_{pn+1}..x_{nn})\ \Box$
	Deontic	English	$\boldsymbol{O}\ \forall x{:}A\ (\exists x_1{:}B1_1..\ x_{n1}{:}B1_{n1}\ Rx_1..x_{p1-1}xx_{p1+1}..x_{n1} \vee \ldots \vee \exists x_1{:}Bn_1..x_{nn}{:}Bn_{nn}\ Rx_1..x_{pn-1}xx_{pn+1}..x_{nn})$
		BM	$\boldsymbol{O}\ \forall x{:}A\ (\exists[\Delta]x_1{:}B1_1..\ x_{n1}{:}B1_{n1}\ Rx_1..x_{p1-1}xx_{p1+1}..x_{n1} \vee \ldots \vee \exists[\Delta]x_1{:}Bn_1..x_{nn}{:}Bn_{nn}\ Rx_1..x_{pn-1}xx_{pn+1}..x_{nn})$
		Mandarin	$\forall\Delta x{:}A\ (\exists\Delta x_1{:}B1_1..\ x_{n1}{:}B1_{n1}\ Rx_1..x_{p1-1}xx_{p1+1}..x_{n1} \vee \ldots \vee \exists\Delta x_1{:}Bn_1..x_{nn}{:}Bn_{nn}\ Rx_1..x_{pn-1}xx_{pn+1}..x_{nn})\ \boldsymbol{O}$

The positive alethic patterns verbalize in English, BM and Mandarin respectively as follows.

For each*A*, R_1 **some** $B1_1$…**that** *A*…**some** $B1_{n1}$ **or**… **or** R_n **some** Bn_1…**that** *A*…**some** Bn_{nn}

Bagi Setiap *A* R_1 **sese**Δ**/sesuatu** $B1_1$…*A* **itu**…**sese** Δ**/sesuatu** $B1_{n1}$ **atau**…**atau** R_n **sese**Δ**/sesuatu** Bn_1…*A* **itu**…**sese**Δ**/sesuatu** Bn_{nn}.

对于每一Δ*AR1* **某一**Δ*B11*…**那一**Δ*A*…**某一**Δ$B1_{n1}$ **或**…**或** R_n **某一**ΔBn_1…**那一**Δ*A*…**某一**ΔBn_{nn}

Table 3.27 Example verbalizations of disjunctive mandatory role constraints where at least one constrained role has front text and/or does not start a predicate reading.

No.	English	BM	Mandarin
1	**For Each** Partner1, on **some** Date **that** Partner1 became the husband of **some** Partner **or** on **some** Date **that** Partner1 became the wife of **some** Partner.	**Bagi Setiap** Pasangan1, pada **sesuatu** Tarikh Pasangan1 **itu** menjadi suami **bagi seseorang** pasangan **atau** pada **sesuatu** Tarikh Pasangan1 **itu** menjadi isteri bagi **seseorang** Pasangan.	**对于每一**位伴侣1，在**某一**个日期**那一**位伴侣1成为**某一**位伴侣的丈夫**或**在**某一**个日期**那一**位伴侣1成为的**某一**位的妻子。

2	**It is obligatory that for each** Vehicle, **some** AutoRetailer sold **that** Vehicle or **that** Vehicle is rented.	**Ia adalah wajib bahawa bagi setiap** Kenderaan, **seseorang** Pembeli menjual Kenderaan **itu atau Kenderaan itu** disewa.	**对于某一**辆交通工具，**某一**位买家卖**那一**辆交通工具**或那一**辆交通工具被租用**是强制性的**。

3.1.9 Value Constraints

3.1.9.1 Object Type Value Constraints

Enumeration:

Figure 3.15 shows the English positive, alethic verbalization patterns for an *enumerated list* of n values ($n \geq 1$) constraining either an entity type A with a reference mode r, or a value type A. In both cases, the constraint applies directly to the (implicit or explicit) value type. The values in the constant list are quoted if and only if the data type associated with the value type is a character string type. If the constraint is *deontic*, "permitted" is used instead of "possible". No negative form is supported.

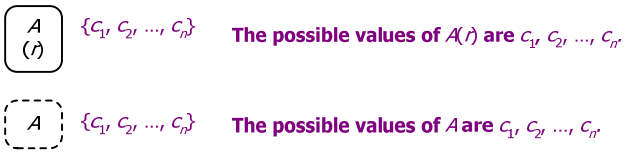


Figure 3.15 Object type value constraint: enumeration

The corresponding verbalizations in Bahasa Melayu and Mandarin are shown below.

BM: **Nilai yang mungkin bagi** *A(r)* **ialah** $c_1, c_2, \ldots, c_n$.
Nilai yang mungkin bagi *A* **ialah** $c_1, c_2, \ldots, c_n$.

Mandarin: *A(r)* **的可能数值是** $c_1, c_2, \ldots, c_n$。
A **的可能数值是** $c_1, c_2, \ldots, c_n$。

Table 3.28 Example verbalizations of object type value constraint: enumeration

English	Mandarin	BM
The possible/permitted values of Gender (Code) **are** 'M', 'F'.	性别(.代码)的可能/允许数值是'男', '女'。	**Nilai yang mungkin/dibenarkan bagi** Jantina (.Kod) **ialah** 'L', 'P'.

Bounded Discrete Range:

The English positive, alethic verbalization patterns are shown above for an *ordered range* of *discrete* values (typically integers or character strings) that is *bounded* at both ends. If the constraint is *deontic*, "permitted" is used instead of "possible". No negative form is supported.

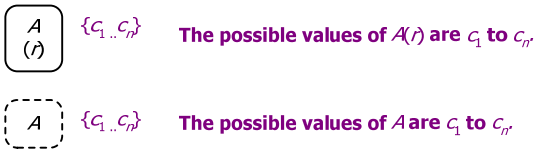


Figure 3.16 Object type value constraint: bounded discrete range

The corresponding verbalizations in Bahasa Melayu and Mandarin are shown below.

BM: **Nilai yang mungkin bagi** *A(r)* **ialah** c_1**, hingga** c_n**.**
Nilai yang mungkin bagi *A* **ialah** c_1**, hingga** c_n**.**

Mandarin: *A(r)* **的可能数值是** c_1**, 至** c_n**。**
A **的可能数值是** c_1**, 至** c_n**。**

Table 3.29 Example verbalizations of object type value constraint: bounded discrete range

English	Mandarin	BM
The possible/permitted values of Grade(Nr) **are** 1 **to** 7.	等级(.数目)的可能/允许数值是 1 至 7。	**Nilai yang Nilai yang mungkin/dibenarkan bagi** Pangkat(.Nr) **ialah** 1 **hingga** 7.

Semi-bounded Discrete Range:

Figure 3.17 shows the English positive, alethic verbalization patterns for an ordered range of discrete values (typically integers or character strings) that is *unbounded at one end*. If the constraint is *deontic*, "permitted" is used instead of "possible". No negative form is supported.

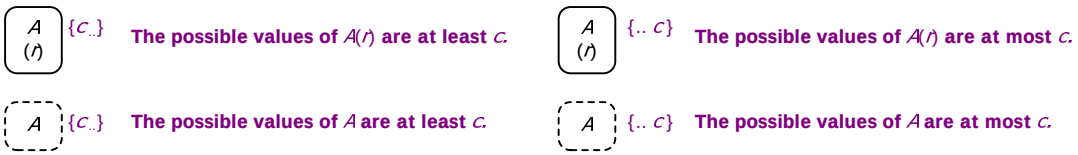


Figure 3.17 Object type value constraint: semi-bounded discrete range

The corresponding verbalizations in English, Bahasa Melayu and Mandarin are shown below.

English:
The possible values of *A(r)* are at least *c*.
The possible values of *A* are at least *c*.
The possible values of *A(r)* are at most *c*.
The possible values of *A* are at most *c*.

BM:
Nilai yang mungkin bagi *A(r)* ialah sekurang-kurangnya *c*.
Nilai yang mungkin bagi *A* ialah sekurang-kurangnya *c*.
Nilai yang mungkin bagi *A(r)* ialah paling banyak *c*.
Nilai yang mungkin bagi *A* ialah paling banyak *c*.

Mandarin:
***A(r)* 的可能数值是 至少 *c*。**
***A* 的可能数值是至少 *c*。**
***A(r)* 的可能数值是最多 *c*。**
***A* 的可能数值是最多 *c*。**

Table 3.30 Example verbalizations of object type value constraint: semi-bounded discrete range

English	Mandarin	BM
The permitted values of PassScore **are at least/at most** 50.	及格分数(.数目)的可能/允许数值 是至少/最多 50。	**Nilai yang mungkin/ dibenarkan bagi** MarkahLulus(.Nr) **ialah sekurang-kurangnya/paling banyak** 50.

Bounded Continuous Range:

Figure 3.18 shows the English positive, alethic verbalization patterns for a *continuous* range of numeric values (*real numbers*) that is *bounded at both ends in an inclusive and/or exclusive sense*. If the data type is real (e.g. float or decimal), ORM's graphical notation depicts ranges using a square bracket to signify inclusion of the end value, and a parenthesis to signify exclusion of the end value (e.g. "(0..10]" denotes a range of positive (hence excluding 0) real numbers up to and including 10). If the constraint is *deontic*, "permitted" is used instead of "possible". No negative form is supported.

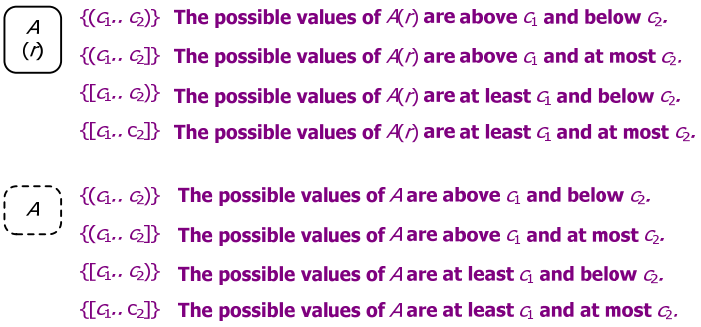


Figure 3.18 Object type value constraint: bounded continuous range

The corresponding verbalizations in English, Bahasa Melayu and Mandarin are shown below.

English: **The possible values of** *A(r)* **are above** c_1**, and below** c_n.
The possible values of *A(r)* **are above** c_1**, and at most** c_n.

The possible values of *A(r)* are at least c_1, and below c_n.

The possible values of *A(r)* are at least c_1, and at most c_n.

The possible values of *A* are above c_1, and below c_n.

The possible values of *A* are above c_1, and at most c_n.

The possible values of *A* are at least c_1, and below c_n.

The possible values of *A* are at least c_1, and at most c_n.

BM: **Nilai yang mungkin bagi *A(r)* ialah melebihi c_1, dan kurang daripada c_n.**

Nilai yang mungkin bagi *A(r)* ialah melebihi c_1, dan paling banyak c_n.

Nilai yang mungkin bagi *A(r)* ialah sekurang-kurangnya c_1, dan kurang daripada c_n.

Nilai yang mungkin bagi *A(r)* ialah sekurang-kurangnya c_1, dan paling banyak c_n.

Nilai yang mungkin bagi *A* ialah melebihi c_1, dan kurang daripada c_n.

Nilai yang mungkin bagi *A* ialah melebihi c_1, dan paling banyak c_n.

Nilai yang mungkin bagi *A* ialah sekurang-kurangnya c_1, dan kurang daripada c_n.

Nilai yang mungkin bagi ialah sekurang-kurangnya c_1, dan paling banyak c_n.

Mandarin: *A(r)* 的可能数值是高于 c_1, 以上同时低于 c_2。

A(r) 的可能数值是高于 c_1, 以上同时最多为 c_2。

A(r) 的可能数值是至少 c_1, 以上同时低于 c_2。

A(r) 的可能数值是至少 c_1, 以上同时最多为 c_2。

A 的可能数值是高于 c_1, 以上同时低于 c_2。

A 的可能数值是高于 c_1, 以上同时最多为 c_2。

A 的可能数值是至少 c_1, 以上同时低于 c_2。

A 的可能数值是至少 c_1, 以上同时最多为 c_2。

Table 3.31 Example verbalizations of object type value constraint: bounded continuous range

English	Mandarin	BM
The possible values of LuggageMass(kg) **are** **a) above** 0 **and at most** 100. **b) above** 0 **and below** 100. **c) at least** 0 **and below**100. **d) at least** 0 **and at most** 150.	行李质量(.公斤)的可能数值是 a) 高于 0/0 以上同时最多为 100。 b) 高于 0/0 以上同时低于 100 /100 以下。 c) 至少 0 同时低于 100/100 以下。 d) 至少 0 同时最多为 150。	**Nilai yang mungkin bagi** BeratBagasi(.Kg) **ialah** a) **melebihi** 0/0 **ke atas dan paling banyak** 100. b) **melebihi** 0/0 **ke atas dan kurang daripada** 100 /100 **di bawah**. c) **sekurang-kurangnya** 0 **dan kurang daripada** 100/100 **di bawah**. d) **sekurang-kurangnya** 0 **dan paling banyak** 150.

Semi-bounded Continuous Range:

Figure 3.19 shows the English positive, alethic verbalization patterns for a *continuous* range of numeric values (real numbers) that is *bounded at one end only* (in an inclusive or exclusive sense). If the constraint is *deontic*, "permitted" is used instead of "possible". No negative form is supported.

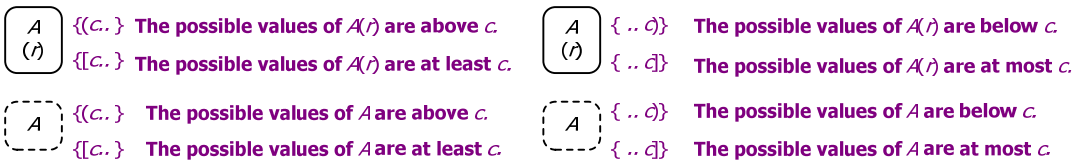


Figure 3.19 Object type value constraint: semi-bounded continuous range

The corresponding verbalizations in English, Bahasa Melayu and Mandarin are shown below.

English:

The possible values of *A(r)* **are above** *c*.
The possible values of *A(r)* **are at least** *c*
The possible values of *A* **are above** *c*.
The possible values of *A* **are at least** *c*
The possible values of *A(r)* **are below** *c*.
The possible values of *A(r)* **are at most c**.
The possible values of *A* **are below** *c*.
The possible values of *A* **are at most** *c*

BM:

Nilai yang mungkin bagi *A(r)* **ialah melebihi** *c*.
Nilai yang mungkin bagi *A(r)* **ialah sekurang-kurangnya** *c*.
Nilai yang mungkin bagi *A* **ialah melebihi** *c*.
Nilai yang mungkin bagi *A* **ialah sekurang-kurangnya** *c*.
Nilai yang mungkin bagi *A(r)* **ialah kurang daripada** *c*.
Nilai yang mungkin bagi *A(r)* **ialah paling banyak** *c*.
Nilai yang mungkin bagi *A* **ialah kurang daripada** *c*.
Nilai yang mungkin bagi *A* **ialah paling banyak** *c*.

Mandarin:

A(r) 的可能数值是高于 *c*。
A(r) 的可能数值是至少 *c*。
A 的可能数值是高于 *c*。
A 的可能数值是至少 *c*。

*A(r)*的可能数值是低于 *c*。

A(r) 的可能数值是最多为 *c*。

A 的可能数值是低于 *c*。

A 的可能数值是最多为 *c*。

Table 3.32 Example verbalizations of object type value constraint: semi-bounded continuous range

English	Mandarin	BM
The possible values of PassScore(.Nr) **are above** 50.	及格分数(.数目)的可能数值是高于 50/50 以上。	**Nilai yang mungkin bagi** MarkahLulus(.Nr) **ialah melebihi** 50/50 **ke atas.**

Combinations:

All of the previous patterns may be used in *combination* with instances of themselves or the others, using a *comma-delimited list* to separate the pattern instances. Table 3.33 shows some examples.

Table 3.33 Example verbalizations of object type value constraint combinations

English	Mandarin	BM
The possible values of ExtremeTemperature(Celsius) **are** -100 **to** -20, 40 **to** 100.	极端温度(.摄氏)的可能数值是-100 至-20, 40 至 100。	**Nilai yang mungkin bagi** SuhuEkstrem(.Celsius)**ialah** -100 **hingga** -20, 40 **hingga** 100。

3.1.9.2 Role Value Constraints

Named Role in Binary Fact Type:

Figure 3.20 shwos the English positive, alethic verbalization patterns for a value constraint on a *named role in a binary predicate*. The constrained role has the role name *r*, and is played by either an entity type *B* with a reference mode *ref*, or a value type *B*. Here *value-spec* is any specification of values allowed as a value constraint on the object type *B*, and hence is replaced by the relevant value pattern described in section 3.1. The values in the constant list are quoted if and only if the data type associated with *B*'s value type is a character string type. If the constraint is *deontic*, "permitted" is used instead of "possible". No negative form is supported.

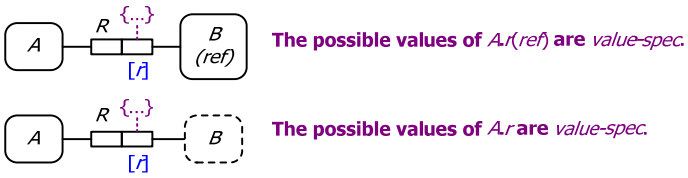


Figure 3.20 Role value constraint: named role in a binary fact type

The corresponding verbalizations in Bahasa Melayu and Mandarin are shown below.

BM: **Nilai yang mungkin bagi** *A(ref)* **ialah** *nilai-spec*.
Nilai yang mungkin bagi *A* **ialah** *nilai-spec*.

Mandarin: *A(ref)* **的可能数值是***数值规格*。
A **的可能数值是***数值规格*。

Table 3.34 Example verbalizations of named role value constraints

English	Mandarin	BM
The possible values of Person.height(cm) **are** [20..270].	人。身高(.厘米)**的可能数值是** [20…270]。	**Nilai yang mungkin bagi** Orang.Tinggi(.cm) **ialah** [20.270].

All other cases (non-binary fact type, or unnamed role in binary fact type):

All other cases are dealt with using the patterns given here, which ignore role names even if they are supplied. Figure 3.21 shows the English positive, alethic verbalization patterns for a value constraint on a *role in a unary or n-ary fact type* ($n > 2$) or on an *unnamed role in a binary fact type*. The constrained role is played by either an entity type *A* with a reference mode *ref*, or a value type *A*. The role may appear in any position. *Value-spec* is any relevant value pattern. *Fact-type* denotes the *primary reading* of the fact type. *If the same object type plays more than one role in the fact type, its instances are distinguished by subscripting* in the fact type reading; the order of subscript numbers is determined by the order of the object types in the primary reading for the fact type, regardless of how the fact type is laid out on a diagram. If the constraint is *deontic*, "permitted" is used instead of "possible". No negative form is supported.

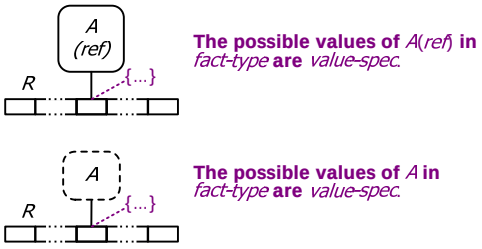


Figure 3.21 Role value constraint: non-binary fact type, or unnamed role in a binary fact type

The corresponding verbalizations in Bahasa Melayu and Mandarin are shown below.

BM: **Nilai yang mungkin bagi** *A(ref)* **untuk jenis-fakta ialah** *nilai-spec.*
Nilai yang mungkin bagi *A* **untuk jenis-fakta ialah** *nilai-spec.*

Mandarin: **对于事实类型的** *A(ref)* **的可能数值是***数值规格。*
对于事实类型的 *A* **的可能数值是***数值规格。*

Table 3.35 Example verbalizations of role value constraints (non-binary fact type or unnamed role)

English	Mandarin	BM
The possible values of Company$_1$(Name) **in** Company$_1$ supplied Product to Company$_2$ **are** 'AlphaRobotics', 'BetaCybernetics'.	**对于**公司 $_1$ 提供货物给公司 $_2$ 的公司 $_1$(.名字)**可能数值是** '阿尔法机械'，'贝塔网盟'。	**Nilai yang mungkin bagi** Syarikat$_1$(.Nama) untuk Syarikat$_1$ membekalkan produk untuk Syarikat$_2$ **ialah** 'AlphaRobotik', 'BetaCybernetics'.

3.1.10 Subset Constraints ⊆

3.1.10.1 Subset constraints between single roles

Unaries:

Figure 3.22 shows the English positive, alethic verbalization patterns for a subset constraint between two unary predicates. The constrained roles are played by either an entity type *A* with a reference mode *ref*, or a value type *A*. If the object type *A* is declared *personal*, "who" is used by default instead of "that" in case (a). If front-text is absent, either pattern (a) or (b) may be used. If front-text is present, pattern (c) must be used. If the constraint is *deontic*, the verbalization is prepended by "it is obligatory that". No negative form is supported (because it is often awkward to concisely specify negation in a natural way without modifying the predicate).

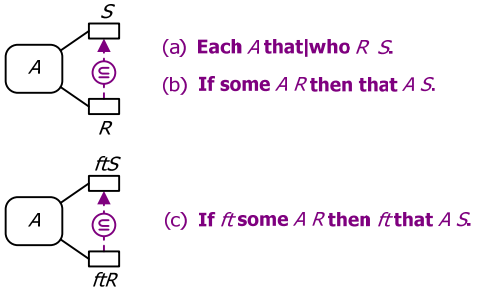


Figure 3.22 Subset constraint between unary fact types

The corresponding verbalizations in Bahasa Melayu and Mandarin are shown below.

BM:
(a) **Setiap** *A* **yang** *R S*.
(b) **Jika sese**Δ*AR*，**maka** *A* **itu** *S*.
(c) **Jika** *ft* **sese**Δ*AR*，**maka** *ft A* **itu** *S*.

Mandarin:
(a) 每一Δ*R* 的 *AS*。
(b) 如果某一Δ*AR*，那么那一Δ*A S*。
(c) 如果 *ft* 某一Δ*AR*，那么 *ft* 那一Δ *AS*。

Table 3.36 Example verbalizations of subset constraints between unaries

English	Mandarin	BM
Each Patient **who** smokes is cancer-prone. **If some** Patient smokes **then that** Patient is cancer-prone.	每一位吸烟的病人易得癌症。 如果某一位病人吸烟，那么那一位 人易得癌症。	**Setiap** Pesakit **yang** merokok mudah menjangkiti kanser. **Jika seseorang** Pesakit merokok **maka** Pesakit **itu** mudah menjangkiti kanser.

Note: In Chinese, a pronoun equivalent to the English pronoun 'who' is usually used to indicate that a person carries out a certain task/action. 'Who' can be easily translated to '谁' in Chinese. When 'that' is used to indicate a thing that is not a person, in Chinese we use '那一'. However, when used to structurally link clauses together, it doesn't work that way. As you can see above, 'Each Patient who smokes' becomes '每一位病人谁吸烟', this doesn't sound fine as a Chinese grammatical phrase.

There are a few options in this regard. Usually the first option is to omit the connector as you can simply add on a second clause as it is. For example, 'There's a policeman outside who wants to see you' can be translated as '外面有一位警察想见你'. The second option is to use '的'to convert the part of the sentence with "who" or "that" into a noun phrase. '的' is used to attribute qualities to things. You commonly see it used to link adjectives to nouns. It can also be used to convert actions into qualities, so that they can be used like adjectives in a sentence. Thus, 'Each patient who smokes' will be '每一位吸烟的病人'.

In BM, pronouns such as'who' are usually used to indicate the person type of noun that carries out a certain task/action. In such a situation, 'yang' is used in BM. As you can see above, 'Each Patient who smokes' becomes 'Setiap Pesakit yang merokok'. In BM, 'that' is rendered as 'itu' and it always placed right after the Noun itself.

Note also that when translating '**some'** in BM, '**sesuatu'** is used for unspecified kinds of object (mostly uncountable nouns), whereas '**Sese-<<noun_classifier>>**' is used for known kinds of object（mostly countable nouns).

All other single role cases where each constrained role starts a predicate reading:

Figure 3.23 shows the English positive, alethic verbalization patterns for a subset constraint between two roles, each of which starts a predicate reading. Predicate *R* has arity *m*+1 ($m \geq 0$) and S has arity *n*+1 ($n \geq 0$). The object types are not necessarily distinct. The constrained roles are played by either an entity type *A* with a reference mode *ref*, or a value type *A*. If the object type *A* is declared *personal*, "who" is used by default instead of "that" in case (a). If an object type plays more than one role in *R* or more than one role in *S*, its occurrences are distinguished by subscripting. If front-text is absent, either pattern (a) or (b) may be used. If front-text is present, pattern (c) must be used. If the constraint is *deontic*, the verbalization is prepended by "it is obligatory that". No negative form is supported.

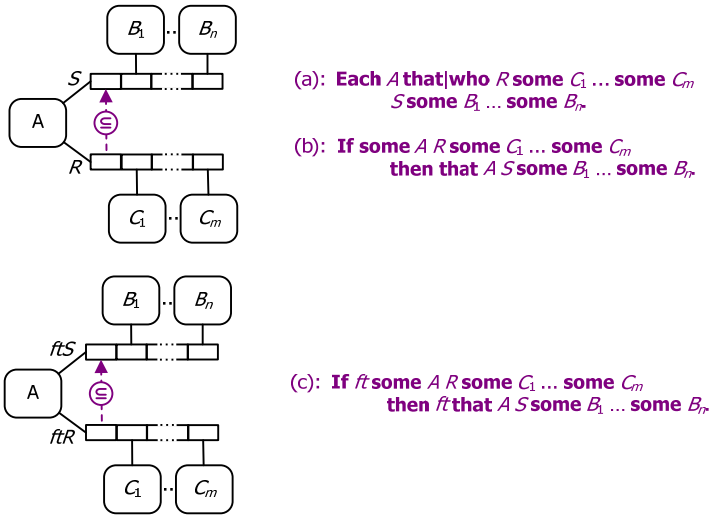


Figure 3.23 Other subset constraints between roles starting a predicate reading

The corresponding verbalizations in Bahasa Melayu and Mandarin are shown below.

BM: (a) **Setiap** *A* **yang** *R* **sese**Δ**/sesuatu** C_1...**sese**Δ **/sesuatu**C_m *S* **sese**Δ **/sesuatu** B_1... **sese**Δ **/sesuatu** B_n**.**

(b) **Jika sese∆ /sesuatu** *A R* **sese∆ /sesuatu** C_1**...sese∆ /sesuatu** C_m**maka** *A* **itu** *S* **sese∆ /sesuatu** B_1**...sese∆ /sesuatu** B_n**.**

(c) **Jika** *ft* **sese∆ /sesuatu** *A R* **sese∆ /sesuatu** C_1**...sese∆ /sesuatu** C_m**maka** *ft A* **itu** *S* **sese∆ /sesuatu** B_1**...sese∆ /sesuatu** B_n**.**

Mandarin:

(a) 每一∆*AR* 某一∆$C_{1...}$某一∆C_m， *S* 某一∆$B_{1...}$某一∆B_n。

(b) 如果某一∆*AR* 某一∆$C_{1...}$某一∆C_m，那么那一∆*A S* 某一∆$B_{1...}$某一∆B_n。

(c) 如果 *ft* 某一∆*AR* 某一∆$C_{1...}$某一∆C_m，那么 *ft* 那一∆*A S* 某一∆$B_{1...}$某一∆B_n。

Table 3.37 Example verbalizations of other subset constraints between roles starting a predicate reading

English	Mandarin	BM
Each Course **that** uses **some** Room in **some** HourSlot is approved. **If some** Course uses **some**Room in **some** HourSlot **then that** Course is approved.	每一门课程被批准使用某一间房间于 某一个时段。 如果某一门课程使用某一间房间，那么那一门课程被批准。	**Setiap** Kursus **yang** menggunakan **sese**buah Bilik dalam sesuatu Masa diluluskan. **Jika sese**buah Kursus menggunakan **sese**buah Bilik. **maka** Kursus **itu** diluluskan.

Other single role cases where a constrained role does not start a predicate reading:

Figure 3.24 displays the English verbalization for these cases. If either the source or the target role does *not* start a predicate reading, use pattern (a) above, where *R* and *S* are the primary predicate readings. The object types are not necessarily distinct, *R* has arity m+1 ($m \geq 0$), and *S* has arity n+1 ($n \geq 0$). The conditional's antecedent is the primary fact type reading for *R*, adjusted by inserting "some" before each *Ci* object type term playing the non-constrained roles, and inserting "that" before the constrained role played by *A*.

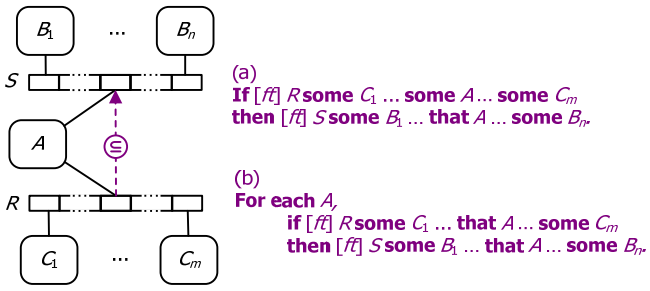


Figure 3.24 Single role subset constraints where some constrained role does not start a predicate reading

The conditional's consequent is the primary fact type reading for *S*, adjusted by inserting "some" before each *Bi* object type term playing the non-constrained roles, and inserting "that" before the constrained role played by *A*. If the primary reading for *R* or *S* has front text *ft*, include it after the "if" or "then" respectively. If an object type plays more than one role in *R* or more than one role in *S*, its occurrences are distinguished by subscripting. If the constraint is *deontic*, "it is obligatory that" is inserted before "if". No negative form is supported.

The corresponding verbalizations in Bahasa Melayu and Mandarin are shown below.

BM: (a) **Jika** *ft R* **sese**Δ **/sesuatu** $C_{1...}$ **sese**Δ **/sesuatu** *A* **…sese**Δ **/sesuatu** C_m
maka *ft S* **sese**Δ **/sesuatu** $B_{1...}$ *A* **itu…sese**Δ **/sesuatu** B_n.

(b) **Bagi setiap** *A* **, Jika** *ft R* **sese**Δ **/sesuatu** $C_{1...}$ *A* **itu …sese**Δ **/sesuatu** C_m
maka *ft S* **sese**Δ **/sesuatu** $B_{1...}$ *A* **itu… sese**Δ **/sesuatu** B_n.

Mandarin:

(a) 如果 *ft R* 某一Δ$C_{1...}$某一Δ*A*…某一ΔC_m 那么 *ft S* 某一Δ$B_{1...}$那一Δ*A*…某一ΔB_n。

(b) 对于每一Δ *A*,如果 *ft R* 某一Δ$C_{1...}$那一Δ*A*…某一ΔC_m 那么 *ft S* 某一Δ$B_{1...}$那一Δ*A*…某一ΔB_n。

Table 3.38 Example verbalizations of single role subset constraints where some constrained role does not start a predicate reading

English	Mandarin	BM
If some Room in **some** HourSlot is booked for **some** Course **then that** Course is approved.	如果某一间房间在某一个时段被预定给某一门课程，那么那一门课程被批准。	**Jika sese**buah Bilik **dalam sesuatu** Masa diluluskan untuk **sese**buah Kursus，**maka** Kursus **itu diluluskan**.

3.1.10.2 Subset constraints between role sequences from single predicates

Predicate readings available from a constrained role in both role sequences where the order of the constrained roles is the same in both predicates:

Figure 3.25 shows the English verbalization patterns for a subset constraint from a sequence of *n* roles ($n \geq 2$) in a single fact type *F1*, played by object types A_1 through A_n (not necessarily distinct), to a compatible role sequence in fact type *F*2. If there are predicate readings *R* and *S* for *F1* and *F2* respectively, starting from a role in each constrained role sequence, where the order of the constrained roles is the same in each predicate (this is the common case) then the verbalization patterns (a) and (b) shown may be used for the positive, alethic case. For each predicate *R* or *S*, if some A_i plays more than one of the constrained roles in that predicate, use subscripts to distinguish the different occurrences. If the constraint is *deontic*, "it is obligatory that" is prepended. No negative form is supported.

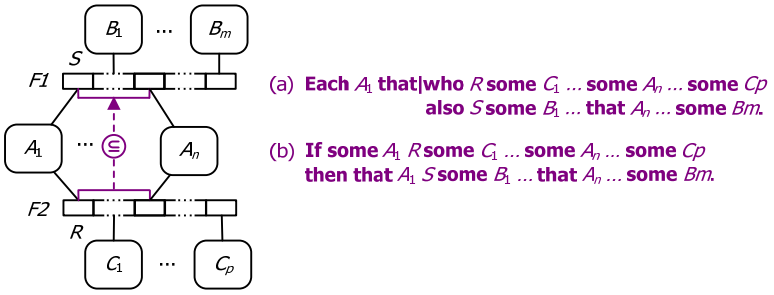


Figure 3.25 Subset constraints between role sequences where some constrained role does not start a predicate reading

The corresponding verbalizations in Bahasa Melayu and Mandarin are shown below.

BM: (a) Setiap A_1 yang *R* sese Δ/sesuatu C_1… sese Δ/sesuatu A_n … sese Δ/sesuatu C_p juga *S* sese Δ/sesuatu B_1 … *A* itu … sese Δ/sesuatu B_m.

(b) Jika sese Δ/sesuatu A_1 *R* sese Δ/sesuatu $C_{1...}$ sese Δ/sesuatu $A_{n...}$ sese Δ/sesuatu C_p maka A_1 itu *S* sese Δ/sesuatu $B_{1...}$ A_n itu… sese Δ/sesuatu B_m.

Mandarin: (a) 每一个 *R* 某一ΔC_1…某一ΔA_n 的 A_1…某一ΔC_p
同时 *S* 某一ΔB_1 …那一Δ*A* …某一ΔB_m。

(b) 如果某一Δ A_1 *R* 某一Δ C_1…某一ΔA_n…某一ΔC_p
那么那一Δ A_1*S* 某一ΔB_1…那一ΔA_n…某一ΔB_m。

Table 3.39 Example verbalizations of subset constraints between role sequences where some constrained role does not start a predicate reading

English	Mandarin	BM
Each Person$_1$ **who** assessed **some** Person$_2$ on **some** Date **also** had a higher rank than **that** Person$_2$ on **that** Date.	每一个评估某一个人$_2$于某一个日期的人$_1$同时拥有比人$_2$高等的等级于那一个日期。	**Setiap** Orang$_1$ **yang** menilaikan Orang$_2$ **pada sesuatu** Tarikh **juga** mempunyai pangkat yang lebih tinggi daripada Orang$_2$ pada Tarikh **itu**.
If some Person$_1$ assessed **some** Person$_2$ on **some** Date **then that** Person$_1$ had a higher rank than **that** Person$_2$ on **that** Date.	如果某一个人$_1$评估某一个人$_2$于某一个日期那么那一个人$_1$拥有比那一个人$_2$高等的等级于某一个日期。	**Jika sese** Orang$_1$ menilaikan **sese** Orang$_2$ pada **sesuatu** Tarikh **maka** Orang$_1$ mempunyai pangkat yang lebih tinggi daripada Orang$_2$ **itu** pada Tarikh **itu**.

The predicate readings may have different orders for even the constrained roles in the two predicates:

Figure 3.26 shows the most general case. The predicate readings for the two predicates may differ in any possible way regarding the order in which the roles are traversed by the reading. Prepend all the object type terms by "some" in both fact type readings, except use "that" instead for the object type terms playing the constrained roles in *F1*. If *R* has front text, include it after "if". If *S* has front text, include it after "then". If the constraint is *deontic*, "it is obligatory that" is prepended. No negative form is supported.

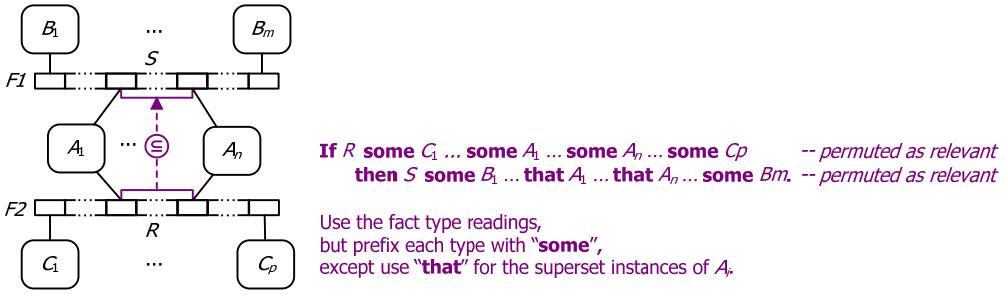


Figure 3.26 Subset constraints between role sequences where the predicate readings may have different orders for even the constrained roles in the two predicates

The corresponding verbalizations in Bahasa Melayu and Mandarin are shown below.

BM: **Jika** *R* **sese Δ/sesuatu** C_1 ... **sese Δ/sesuatu** A_1 ... **sese Δ/sesuatu** A_n... **sese Δ/sesuatu** C_p
maka *S* **sese Δ/sesuatu** B_1...A_1 **itu**... A_n **itu**... **sese Δ/sesuatu** B_m。

Mandarin: 如果 *R* 某一Δ C_1 ...某一ΔA_1 ...某一ΔA_n...某一ΔC_p
那么 *S* 某一ΔB_1...那一ΔA_1...那一Δ A_n...某一ΔB_m。

The first example below assumes the readings are: Student passed Exam; Exam was taken by Student (the reading Student sat Exam is not provided).

Table 3.40 Example verbalizations of subset constraints between role sequences where the predicate readings may have different orders for even the constrained roles in the two predicates

English	Mandarin	BM
If some Student passed **some** Exam **then that** Exam was taken by **that** Student.	如果某一名学生通过某一个测验那么那一个测验被那一名学生所报考。	**Jika seseorang** Pelajar lulus dalam **sesebuah** Ujian **maka** Ujian **itu** diambil oleh Pelajar **itu**.
If some Student on **some** Exam obtained **some** Score **then that** Exam was taken by **that** Student.	如果某一名学生在某一个测验获得某一个分数那么那一个测验被那一名学生所报考。	**Jika seseorang** Pelajar bagi **sesuatu** Ujian memperolehi **sesuatu** markah **maka** Ujian **itu** diambil oleh Pelajar **itu**.

3.1.11 Equality Constraints

3.1.11.1 Equality Constraints between 2 Single Roles

Unaries:

Figure 3.27 shows the English positive, alethic verbalization patterns for an equality constraint between two unary predicates. The constrained roles are played by either an entity type *A* with a reference mode *ref*, or a value type *A*. If the object type *A* is declared *personal*, "who" is used by default instead of "that" in case (a). If front-text is absent, either pattern (a) or (b) may be used. If front-test is present, pattern (c) must be used. If the constraint is *deontic*, the verbalization (a) is prepended by "It is obligatory that". No negative form is supported.

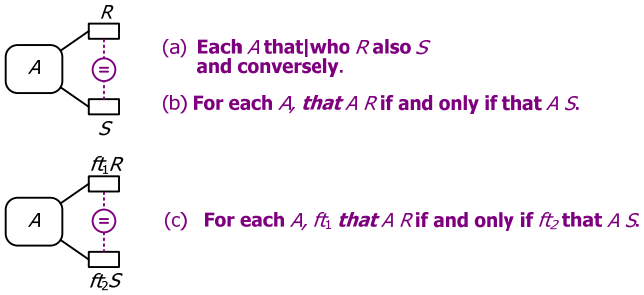


Figure 3.27 Equality constraints between two unary fact types

The corresponding verbalizations in Bahasa Melayu and Mandarin are shown below.

BM: (a) **Setiap** *A* **yang** *R* **juga** *S* **dan sebaliknya.**

(b) **Bagi setiap** *A*， *A* **itu** *R* **jika dan hanya jika** *A S*.

(c) **Bagi setiap** *A*，ft_1 *A* **itu** *R* **jika dan hanya jika** ft_2 *A S*.

Mandarin: (a) 每一Δ *R* 的 *A* 也 *S* 同时相互。

(b) 对于每一Δ *A*，那一Δ *A R* 除非那一Δ *A S*。

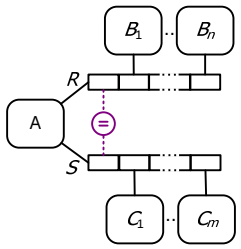
(c) 对于每一Δ *A*，ft_1 那一Δ *A R* 除非 ft_2 那一Δ *A S*。

Table 3.41 Example verbalizations of equality constraints between unaries

English	Mandarin	BM
For each Patient, daily **that** Patient has a blood pressure reading **if and only if** nightly **that** Patient has a pulse reading.	对于每一位病人，每一日那一位病人拥有血压读数除非每一夜那一位病人拥有脉冲读数。	**Bagi setiap** Pesakit， harian Pesakit **itu** memiliki Bacaan Tekanan Darah **jika dan hanya jika** setiap malam Pesakit **itu** memiliki Bacaan Nadi.

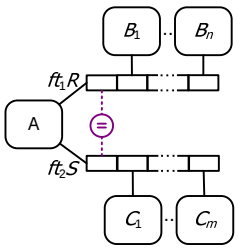
All other 2-role cases where each constrained role starts a predicate reading:

Figure 3.28 shows the positive, alethic verbalization patterns for other cases of an equality constraint between two roles, each of which starts a predicate reading. Predicate *R* has arity *n*+1 ($n \geq 0$) and S has arity *m*+1 ($m \geq 0$). The object types are not necessarily distinct.



(a): **Each** *A* **that|who** *R* **some** B_1 ... **some** B_n
also *S* **some** C_1 ... **some** C_m **and conversely**.

(b): **For each** *A*, **that** *A R* **some** B_1 ... **some** B_n
if and only if that *A S* **some** C_1 ... **some** C_m.



(c): **For each** *A*, ft_1 **that** *A R* **some** B_1 ... **some** B_n
if and only if ft_2 **that** *A S* **some** C_1 ... **some** C_m.

Figure 3.28 Other cases of equality constraints between two roles that start a predicate reading

The corresponding verbalizations in Bahasa Melayu and Mandarin are shown below.

BM: (a) **Setiap** *A* **yang** *R* **sese**Δ**/sesuatu** $B_{1...}$ **sese**Δ**/sesuatu** B_n
Juga *S* **sese**Δ**/sesuatu** $C_{1...}$ **sese**Δ**/sesuatu** C_m **dan sebaliknya.**

(b) **Bagi setiap** *A*， *A* **itu** *R* **sese**Δ**/sesuatu** $B_{1...}$ **sese**Δ**/sesuatu** B_n
Jika dan hanya jika *A* **itu** *S* **sese**Δ**/sesuatu** $C_{1...}$ **sese**Δ**/sesuatu** C_m.

(c) **Bagi setiap** *A*， *ft*$_1$ *A* **itu** *R* **sese**Δ**/sesuatu** *B*$_{1...}$ **sese**Δ**/sesuatu** *B*$_n$

Jika dan hanya jika *ft*$_2$ *A* **itu** *S* **sese**Δ**/sesuatu** *C*$_{1...}$ **sese**Δ**/sesuatu** *C*$_m$**.**

Mandarin: (a) 每一Δ *R* 某一Δ *B*$_{1...}$某一Δ *B*$_n$ 的 *A*

同时 *S* 某一Δ *C*$_{1...}$某一Δ *C*$_m$ 同时相互。

(b) 对于每一Δ *A*，那一Δ *A R* 某一Δ *B*$_{1...}$某一Δ *B*$_n$

除非那一Δ *A S* 某一Δ *C*$_{1...}$某一Δ *C*$_m$。

(c) 对于每一Δ *A*，*ft*$_1$ 那一Δ *A R* 某一Δ *B*$_{1...}$某一Δ *B*$_n$

除非 *ft*$_2$ 那一Δ *A S* 某一Δ *C*$_{1...}$某一Δ *C*$_m$。

Table 3.42 Example verbalizations of other equality constraints between 2 roles

English	Mandarin	BM
Each Patient **who** has **some** systolic BloodPressure **also** has **some** diastolic BloodPressure **and conversely**.	每一位拥有某一个收缩血压的病人同时拥有某一个舒张血压同时相互。	**Setiap** Pesakit **yang** mempunyai **sesuatu** sistolik tekananDarah **dan juga sesuatu** diastolic tekananDarah **dan sebaliknya**.
For each Patient, **that** Patient has **some** systolic BloodPressure **if and only if that** Patient has **some** diastolic BloodPressure.	对于每一位病人， 那一位病人拥有某一个收缩血压除非那一位病人拥有舒张血压。	**Bagi setiap** Pesakit, Pesakit **itu** mempunyai **sesuatu** sistolik tekananDarah **jika dan hanya jika** Pesakit mempunyai **sesuatu** diastolic tekananDarah.

All other 2-role cases where some constrained role does not start a predicate reading:

Figure 3.29 shows the English verbalization patterns where at least one constrained role does *not* start a predicate reading. Here *R* has arity *n*+1 ($n \geq 0$) and *S* has arity *m*+1 ($m \geq 0$). The biconditional's first operand is the primary fact type reading for *R*, adjusted by inserting "some" before each *Bi* object type term playing the non-constrained roles, and inserting "that" before the constrained role played by *A*. The biconditional's second operand is the primary fact type reading for *S*, adjusted by inserting "some" before each *Ci* object type term playing the non-constrained roles, and inserting "that" before the constrained role played by *A*. If the primary reading

for *R* or *S* has front text, the front text precedes the first "some" in the *R* or *S* reading respectively. If *R* or *S* is unary, use the shorter form shown for its reading (if front text exists, insert the front text before "that"). If an object type plays more than one role in *R* or more than one role in *S*, its occurrences are distinguished by subscripting. If the constraint is *deontic*, "it is obligatory that" is prepended. No negative form is supported.

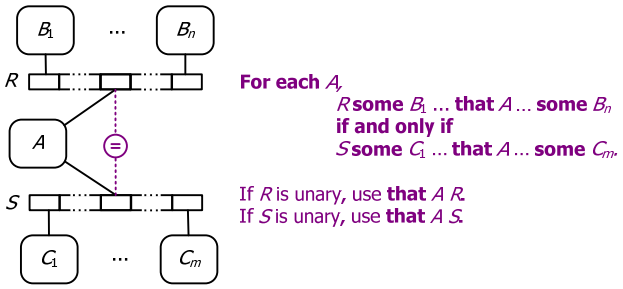


Figure 3.29 Other cases of equality constraints between two roles

The corresponding verbalizations in Bahasa Melayu and Mandarin are shown below.

BM: **Bagi setiap** *A*,

R **sese** Δ**/sesuatu** B_1 ... *A* **itu**... **sese** Δ**/sesuatu** B_n

Jika dan hanya jika *S* **sese** Δ**/sesuatu** C_1... *A* **itu**... **sese** Δ**/sesuatu** C_m.

Mandarin: **对于每一**Δ *A*，

R 某一Δ B_1 ... 那一Δ *A*...某一Δ B_n

除非 *S* 某一Δ C_1... 那一Δ *A*...某一Δ C_m。

Table 3.43 Example verbalizations of equality constraints between 2 roles where some role does not start a predicate reading

English	Mandarin	BM
For each Patient, **that** Patient has **some** systolic BloodPressure **if and only if some** Bloodpressure is the diastolic pressure of **that** Patient.	对于每一位病人，那一位病人拥有某一个收缩血压除非某一个血压是那一位病人的舒张压。	**Bagi setiap** Pesakit, Pesakit **itu** mempunyai **sesuatu** sistolik tekananDarah **jika dan hanya jika sesuatu** tekananDarah ialah diastolik tekanan bagi Pesakit **itu**.

3.1.11.2 Equality constraints over 3 or more roles

Figure 3.30 shows the English positive, alethic pattern. The object types are not necessarily distinct. If *A* plays more than one role in at least one of the R_i predicates, do *not* subscript its instances. Each predicate R_i may have *ni* roles ($ni \geq 0$) plus the role played by *A*. The deontic version prepends "It is obligatory that" to the positive form. If front text exists, start with it.

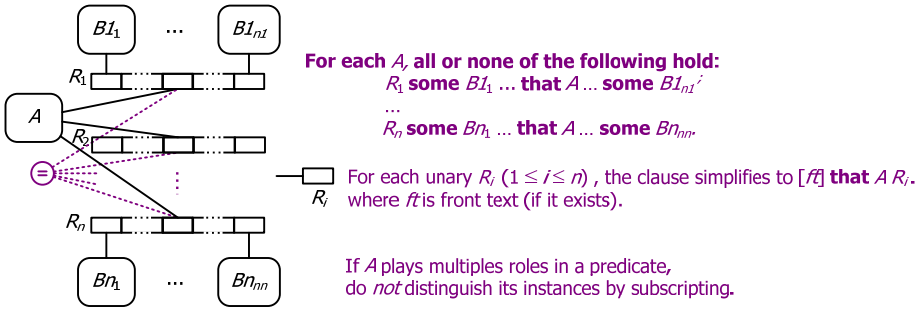


Figure 3.30 Equality constraints over three or more roles

The corresponding verbalizations in Bahasa Melayu and Mandarin are shown below.

BM: **Bagi setiap** *A*, **memiliki kesemua atau tiada daripada berikut:**

*R*₁ **sese**Δ **/sesuatu** *B1*₁... *A* **itu**... **sese**Δ **/sesuatu** *B1*ₙ₁

...

*R*ₙ **sese**Δ **/sesuatu** *Bn*₁... *A* **itu**... **sese**Δ **/sesuatu** *B1*ₙₙ

Mandarin: **对于每一**Δ *A*，**持有以下全部或无任何一个条件：**

*R*₁**某一**Δ *B1*₁...**那一**Δ ***A***...**某一**Δ *B1*ₙ₁

...

*R*ₙ**某一**Δ *Bn*₁...**那一**Δ *A*...**某一**Δ*B1*ₙₙ。

Table 3.44 Example verbalizations of equality constraints over more than 2 roles

English	Mandarin	BM
For each Person, **all or none of the following hold**: **that** Person gave first preference to **some** Person; **that** Person gave second preference to **some** Person; **some** Person was given third preference by **that** Person.	对于每一个人，持有以下全部或无任何一个条件： 那一个人给予某一个人第一选择； 那一个人给予某一个人第二选择； 某一个人被那一个人给予第三选择。	**Bagi setiap** Orang，**memiliki kesemua atau tiada daripada berikut**: Orang **itu** memberikan **sese**Orang Pilihan Pertama; Orang **itu** memberikan **sese**Orang Pilihan Kedua; **sese**orang diberikan Pilihan Ketiga oleh Orang **itu**.

2.1.11.3 Equality constraints between 2 Role Sequences from Single Predicates

Figure 3.31 shows the English positive alethic verbalization patterns for this case.

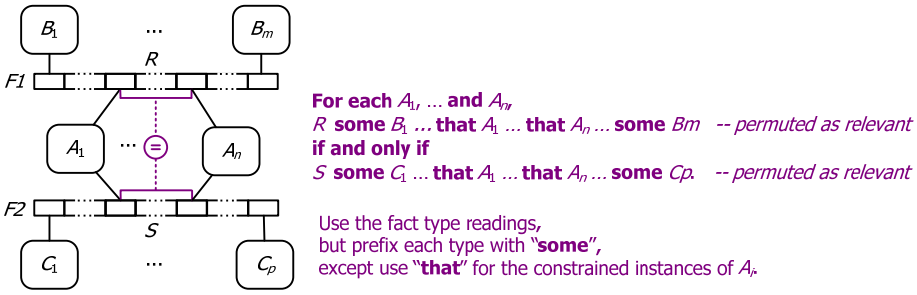


Figure 3.31 Equality constraints between 2 role sequences from single predicates

The corresponding verbalizations in Bahasa Melayu and Mandarin are below.

BM: **Bagi setiap** A_1 **dan**,… A_n ,

R **sese** Δ/**sesuatu** B_1 . . . A_1**itu**… A_n **itu**… **sese** Δ/**sesuatu** B_m

Jika dan hanya jika S **sese** Δ/**sesuatu** C_1... A_1 **itu** … A_n **itu**… **sese** Δ/**sesuatu** C_p.

Mandarin: 对于每一▲ 'A_1 与,… A_n ',

R 某一Δ B_1 . . . 那一Δ A_1…那一Δ A_n…某一Δ B_m

除非 S 某一Δ C_1...那一Δ A_1…那一Δ A_n…某一Δ C_p。

Table 3.45 Example verbalizations of equality constraints between 2 role sequencdes from single predicates

English	Mandarin	BM
For each Patient **and** Date, **that** Patient on **that** Date had **some** systolic BloodPressure **if and only if that** Patient on **that** Date had **some** diastolic BloodPressure.	对于每一个‘病人与日期’，那一位病人在那一个日期拥有某一个收缩血压除非那一位病人在那一个日期拥有某一个舒张血压。	**Bagi setiap** Pesakit **dan** Tarikh, Pesakit **itu** bagi Tarikh **itu** mempunyai **sesuatu** sistolik tekananDarah **jika dan hanya jika** Pesakit **itu** bagi Tarikh **itu** mempunyai **sesuatu** diastolic tekananDarah.

2.1.11.4 Equality Constraints between 3 or more Role Sequences from Single Predicates

Figure 3.32 shows the English positive alethic verbalization patterns for this general and rare case.

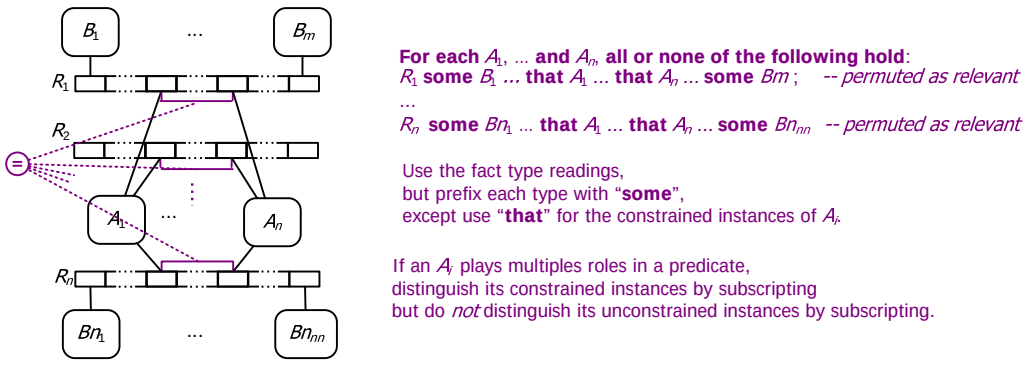


Figure 3.32 Equality constraints between 3 or more role sequences from single predicates

The corresponding verbalizations in Bahasa Melayu and Mandarin are shown below.

BM: **Bagi setiap** A_1 **dan**,… A_n，**memiliki kesemua atau tiada mana-mana satu daripada berikut:**

R_1 **sese** Δ**/sesuatu** $B_{1...}$ A_1 **itu**… A **itu**… **sese** Δ**/sesuatu** Bm

…

R_n **sese** Δ**/sesuatu** $Bn_{1...}$ A_1 **itu**… A **itu**… **sese** Δ**/sesuatu** Bn_{nn}.

Mandarin: 对于每一▲ 'A_1 与,… A_n'，持有以下全部或无任何一个条件：

R_1 某一Δ $B_{1..}$那一Δ A_1…那一Δ A…某一Δ Bm

…

R_n 某一Δ $Bn_{1..}$那一Δ A_1…那一Δ A…某一ΔBn_{nn}。

Table 3.46 Example verbalizations of equality constraints between 3 or more role sequences from single predicates

English	Mandarin	BM
For each Star **and** Date, **all or none of the following hold**: **that** Star on **that** Date was located at **some** x-Coordinate; **that** Star on **that** Date was located at **some** y-Coordinate; **that** Star on **that** Date was located at **some** z-Coordinate.	对于每一个‘星星与日期’，持有以下全部或无任何一个条件： 那一个星星在那一个日期坐落于某一个 x 座标 ； 那一个星星在那一个日期坐落于某一个 y 座标； 那一个星星在那一个日期坐落于某一个 z 座标。	**Bagi setiap** Bintang **dan**Tarikh, **memiliki kesemua atau tiada daripada berikut:** Bintang **itu** bagi Tarikh **itu** terletak di Koordinat-x; Bintang **itu** bagi Tarikh **itu** terletak di Koordinat-y; Bintang **itu** bagi Tarikh **itu** terletak di Koordinat-z.

Note: In Chinese, a noun classifier is used to classify every single type of noun (countable and uncountable nouns) that is being quantified. For countable nouns, the non classifier varies according to the noun type (e.g '尾' is used for any kind of fish). For uncountable nouns, '个' is generally used to represent the noun itself (e.g '个' for Name, Number, etc.). '个' may also be used by some types of countable nouns (e.g '个' for Person, Room, etc.). In Bahasa Melayu, a noun classifier is always used to classify a countable noun type, but is not used uncountable nouns (e.g time, name, etc.). In BM, 'orang' can act as a noun classifier as well as denoting an object type (viz. Person). In the case where 'orang' denotes an object type, no noun classifer is used.

3.1.12 Exclusion ⊗ and Exclusive-or ⊗ Constraints

3.1.12.1 Exclusion Constraints over 2 or more Single Roles

Exclusion constraint but no inclusive-or constraint over the same roles:

Figure 3.33 shows the English alethic verbalization pattern. Each predicate R_i may have n_i roles ($n_i \geq 0$) plus the constrained role played by A. No negative version is supported. The deontic version prepends "It is obligatory that" to the positive form. If front text exists, start with it.

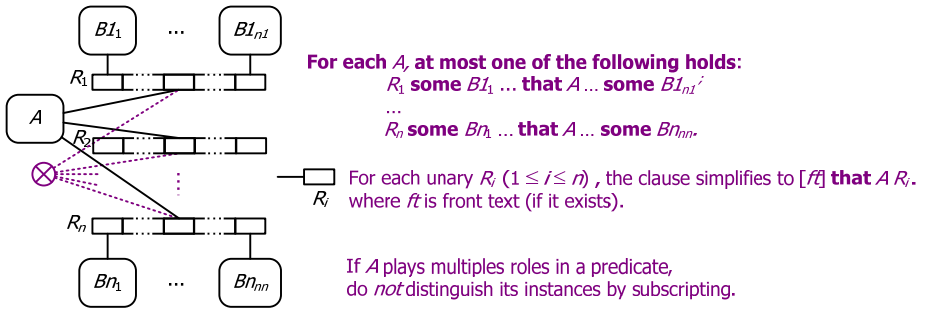


Figure 3.33 Exclusion constraints over single roles

The corresponding verbalizations in Bahasa Melayu and Mandarin are shown below.

BM: **Bagi setiap** *A*，**memiliki paling banyak satu daripada berikut**：

*R*₁ **sese**Δ **/sesuatu** *B1*₁… *A* **itu**… **sese**Δ **/sesuatu** *B1*ₙ₁

…

*R*ₙ **sese**Δ **/sesuatu** *Bn*₁… *A* **itu**… **sese**Δ **/sesuatu** *B1*ₙₙ

Mandarin: **对于每一**Δ *A*，**持有以下最多一个条件**：

*R*₁**某一**Δ *B1*₁…**那一**Δ ***A***…**某一**Δ *B1*ₙ₁

…

*R*ₙ**某一**Δ *Bn*₁…**那一**Δ *A*…**某一**Δ*B1*ₙₙ。

Table 3.47 Example verbalizations of exclusion constraints over single roles

English	Mandarin	BM
For each Application, **at most one of the following holds**: **that** Application was approved on **some**Date; **that** Application was rejected on **some** Date; **that** Application was withdrawn on **some**Date.	对于每一个申请，持有以下最多一个条件： 那一个申请在某一个日期被批准； 那一个申请在某一个日期被拒绝； 那一个申请在某一个日期被撤回。	**Bagi setiap** Permohonan，**memiliki paling banyak satu daripada berikut**： Permohonan **itu** diluluskan pada **sesuatu** Tarikh； Permohonan **itu** ditolak pada **sesuatu** Tarikh； Permohonan **itu** ditarik balik pada **sesuatu** Tarikh。

Exclusion plus Inclusive-or constraint over the same roles:

An exclusive-or (xor) constraint combines an exclusion constraint with an inclusive-or (disjunctive mandatory role) constraint. Figure 3.34 shows the English alethic verbalization pattern.

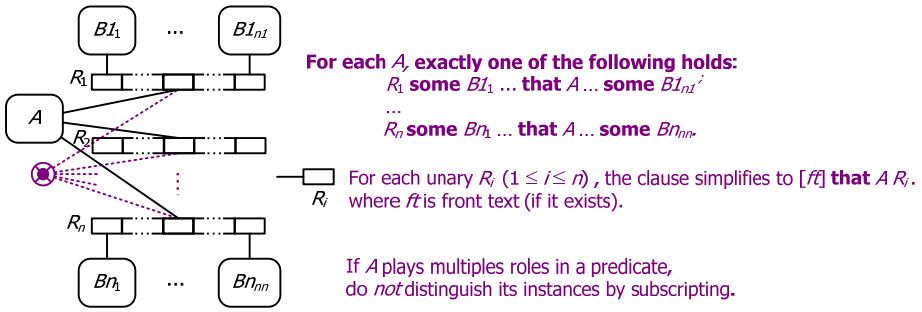


Figure 3.34 Exclusive-or constraints over single roles

The corresponding verbalizations in Bahasa Melayu and Mandarin are shown below.

BM: **Bagi setiap** *A*，**memiliki hanya satu daripada berikut**：

*R*₁ **sese**Δ **/sesuatu** *B1*₁… *A* **itu**… **sese**Δ **/sesuatu** *B1*ₙ₁

…

*R*ₙ **sese**Δ **/sesuatu** *Bn*₁… *A* **itu**… **sese**Δ **/sesuatu** *B1*ₙₙ

Mandarin: 对于每一Δ *A*，持有仅有一个条件：

R$_1$某一Δ *B1*$_{1...}$那一Δ ***A***…某一Δ *B1*$_{n1}$

…

R$_n$某一Δ *Bn*$_{1...}$那一Δ *A*…某一Δ*B1*$_{nn}$。

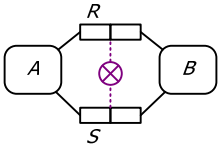
Table 3.48 Example verbalizations of exclusive-or constraints over single roles

English	Mandarin	BM
For each Employee, **exactly one of the following holds**: **that** Employeeis tenured; **that** Employeeis pre-tenured; **that** Employeeis contracted till **some**Date.	对于每一名雇员，持有仅有一个条件： 那一名雇员是终身职位； 那一名雇员是任期职位； 那一名雇员是签约至某一个日期。	**Bagi setiap** Pekerja，**memiliki hanya satu daripada berikut**： Pekerja **itu** ialah Pekerja Bertempoh； Pekerja **itu** ialah Pekerja Pra-Bertempoh； Pekerja **itu** ialah dikontrakkan sehingga **sesuatu** Tarikh.

3.1.12.2 Exclusion Constraints between 2 or more Role Sequences from Single Predicates

Exclusion constraint between 2 binary predicates with readings in the same direction between different object types, with no front text and no hyphen binding:.

Figure 3.35 shows the English alethic verbalization pattern.



No *A R*s and *S*s the same *B*.

Figure 3.35 Exclusion constraints between 2 binary predicates (simple case)

The corresponding verbalizations in Bahasa Melayu and Mandarin are shown below.

BM: **Tiada mana-mana satu** Δ *A'*$_S$ **dan** *S'*$_S$ *B* **yang sama.**

Mandarin: 无任何一Δ *A'*$_S$和 *S'*$_S$同一Δ*B*。

Table 0.49 Example verbalizations of exclusion constraints between 2 binaries

English	Mandarin	BM
No Person authored **and** reviewed **the same**Book.	无任何一个人能撰写和审查同一本书。	**Tiada mana-mana satu** Orang boleh mengarang **dan** mengkaji semula Buku **yang sama.**

All other cases:

Figure 3.36 shows the English alethic verbalization pattern.

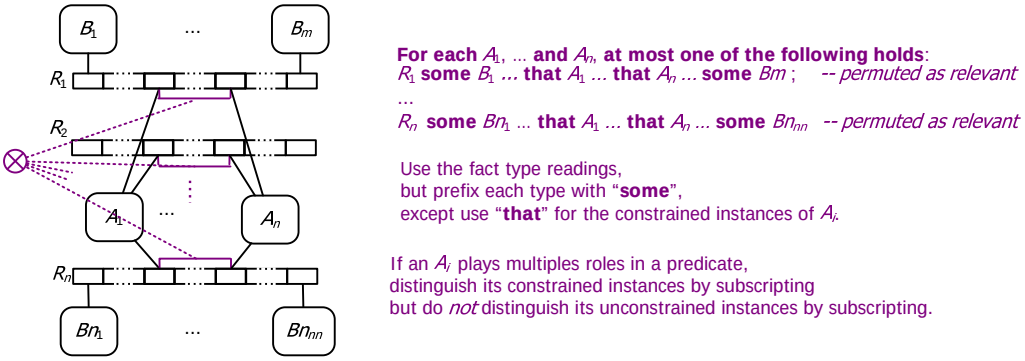


Figure 3.36 Exclusion constraints over role sequences (general case)

The corresponding verbalizations in Bahasa Melayu and Mandarin are shown below.

BM: **Bagi setiap** A_1,… ***dan*** A_n, **memiliki paling banyak satu daripada:**

R_1 **sese**Δ**/sesuatu** $B_{1...}$ A_1 **itu**… A **itu**… **sese**Δ**/sesuatu** Bm

…

R_n **sese**Δ**/sesuatu** $Bn_{1...}$ A **itu**… A **itu**… **sese**Δ**/sesuatu** Bn_{nn}.

Mandarin: **对于每一▲** 'A_1,… **与** A_n', **持有最多一个条件:**

R_1 **某一**Δ $B_{1...}$**那一**Δ A_1…**那一**Δ A…**某一**Δ Bm

…

R_n **某一**Δ $Bn_{1...}$**那一**Δ A_1 A…**那一**Δ A…**某一**ΔBn_{nn}。

Table 3.50 Example verbalizations of exclusion constraints over role sequences (general case)

English	Mandarin	BM
For each Competitor **and** Event, **at most one of the following holds**: **that** Competitor came first in **that** Event; **that** Competitor came second in **that** Event; **that** Competitor came third in **that** Event.	对于每一个‘竞争者与活动’，持有最多一个条件： 那一名竞争者第一个加入那一项活动； 那一名竞争者第二个加入那一项活动； 那一名竞争者第一个加入那一项活动。	**Bagi setiap** Pesaing **dan** Aktiviti, **memilki paling banyak satu daripada**: Pesaing **itu** adalah orang pertama menyertai Aktiviti **itu**; Pesaing **itu adalah** orang kedua menyertai Aktiviti **itu**; Pesaing **itu** adalah orang ketiga menyertai Aktiviti **itu**;

3.1.13 Frequency Constraints

2.1.13.1 Internal Frequency Constraints on Single Roles

Simple frequency constraint on one role of a binary with a predicate reading from that role:

Figure 3.37 shows the English verbalization pattern for a simple alethic frequency constraint on one role of a binary predicate *R*, where the role is played by the object type *A* (an entity type or value type), and *R* has a predicate reading starting at that role, possibly including front text. Let *fc* denote the simple frequency constraint specification, which is of the form *n* (exactly *n*), ≥*n* (at least *n*), ≤*n* (at most *n*), or *n..m* (at least *n* and at most *m*). If the object type *A* plays two or more roles in the same predicate, distinguish its instances by numbered subscripts. If the constraint is *deontic*, ‘It is obligatory that’ is prepended. No negative form is supported.

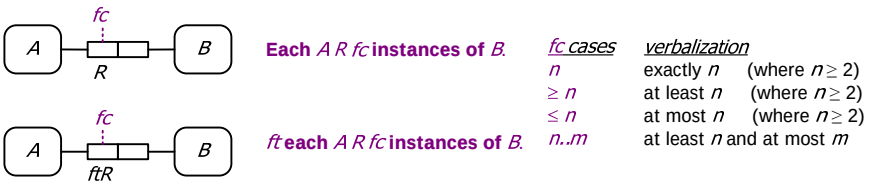


Figure 3.37 Frequency constraints on a role of a binary fact type

The corresponding verbalizations in Bahasa Melayu and Mandarin are shown below.

BM: **Setiap** *A R fc* Δ *B*。

fc case	*verbalization*
n	*hanya n (where n ≥ 2)*
≥ n	*sekurang-kurangnya n (where n ≥ 2)*
≤ n	*paling banyak n (where n ≥ 2)*
n…m	*sekurang-kurangnya ndan paling banyak m*

ft **Setiap** *A R fc* Δ *B*。

Mandarin: **每一**Δ *A R fc*Δ *B*。

fc case	*verbalization*
n	*仅有 n (where n ≥ 2)*
≥ n	*至少 n (where n ≥ 2)*
≤ n	*最多 n (where n ≥ 2)*
n…m	*至少 n 同时最多 m*

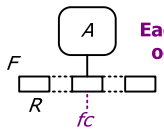
ft **每一**Δ *A R fc*Δ *B*。

Table 3.51 Example verbalizations of frequency constraints on a role of a binary

English	Mandarin	BM
Each Person is a child of **at most** 2 **instances of** Person.	**每一**个人是**最多两**个人的孩子.	**Setiap** Orang ialah anak bagi **paling banyak 2** Orang.

All other cases of a frequency constraint on a single role:

Figure 3.38 shows the English verbalization pattern for all other cases of an alethic frequency constraints on a single role of a binary or longer predicate *R*, where the role is played by the object type *A*. Here *F* denotes the primary fact type reading of the fact type (possibly including front text) and *fc* denotes the frequency constraint specification, or a finite list of these components (verbalized as a disjunction by placing "**times or**" between the components). If the object type *A* plays two or more roles in the same predicate, distinguish its instances by numbered subscripts. If the constraint is *deontic*, 'It is obligatory that' is prepended. No negative form is supported.



Each *A* in the population of "*F*" occurs there *fc* times.

fc cases	*verbalization*
n	exactly *n* (a positive integer)
≥ *n*	at least *n*
≤ *n*	at most *n*
n..m	at least *n* and at most *m*
fc1, fc2, ...	*fc1_verbalization* **times or** *fc2_verbalization* **times or ...**

Figure 3.38 Other frequency constraints on a single role

The corresponding verbalizations in Bahasa Melayu and Mandarin are shown below.

BM: **Setiap *A* dalam populasi untuk *"F"* wujud *fc* kali.**

fc case	*verbalization*
n	*hanya n (where n ≥ 2)*
≥ *n*	*sekurang-kurangnya n (where n ≥ 2)*
≤ *n*	*paling banyak n (where n ≥ 2)*
n…m	*sekurang-kurangnya ndan paling banyak m*
fc1,fc2,… fc1_verbalization kail atau fc 2 _verbalization kali atau …	

Mandarin: **每一Δ *A* 在 *"F"*的群体里出现 *fc* 次。**

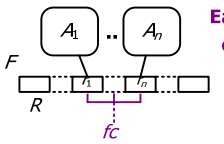
fc case	*verbalization*
n	*仅有 n (where n ≥ 2)*
≥ *n*	*至少 n (where n ≥ 2)*
≤ *n*	*最多 n (where n ≥ 2)*
n…m	*至少 n 同时最多 m*
fc1,fc2,… fc1_verbalization 次或 fc 2 _verbalization 次或…	

Table 3.52 Example verbalizations of other frequency constraints on a single role

English	Mandarin	BM
Each Paper **in the population of** "Expert reviews Paper" **occurs there at least** 2 **times.**	每一张论文在"专家评论论文"的群体里出现至少 2 次。	**Setiap** kertas **dalam populasi untuk**"Pakar mengkaji semula Kertas"**wujud sekurang-kurangnya 2 kali**.

3.1.13.2 Internal Frequency Constraints on Multiple Roles

This case is similar to that of the previous section, except that the frequency constraint spans n roles ($n > 1$) played respectively by object types A_1 .. A_n (not necessarily distinct). Figure 3.39 shows the English alethic verbalization pattern. The deontic pattern is obtained by prepending '**it is obligatory that**'. No negative form is supported.



Each A_1, .., A_n **combination in the population of** "*F*" **occurs there** *fc* **times.**

fc cases	*verbalization*
n	exactly *n* (a positive integer)
≥ *n*	at least *n*
≤ *n*	at most *n*
n..m	at least *n* and at most *m*
fc1, fc2, ...	*fc1_verbalization* **times or** *fc2_verbalization* **times or ...**

Figure 3.39 Internal frequency constraints on a multiple roles

The corresponding verbalizations in Bahasa Melayu and Mandarin are shown below.

BM: **Setiap gabungan bagi** A_1,..,A_n **dalam populasi untuk** *"F"* **wujud** *fc* **kali.**

fc case	*verbalization*
n	*hanya n (where n ≥ 2)*
≥ *n*	*sekurang-kurangnya n (where n ≥ 2)*
≤ *n*	*paling banyak n (where n ≥ 2)*
n…m	*sekurang-kurangnya ndan paling banyak m*
fc1,fc2,…	*fc1_verbalization kail atau fc 2 _verbalization kali atau …*

Mandarin: **每一▲** A_1,..,A_n **的组合在** *"F"* **的群体里出现** *fc* **次。**

fc case	*verbalization*
n	*仅有 n (where n ≥ 2)*
≥ *n*	*至少 n (where n ≥ 2)*
≤ *n*	*最多 n (where n ≥ 2)*
n…m	*至少 n 同时最多 m*
fc1,fc2,…	*fc1_verbalization 次或 fc 2 _verbalization 次或…*

Table 3.53 Example verbalizations of frequency constraints on multiple roles

English	Mandarin	BM
EachYear, City **combination in the population of** "In Year in City monitors of MonitorKind sold in Quantity" **occurs there exactly** 2 **times.**	每一个年份，城市的组合在"在年份在城市，"显示器种类的显示器被销售于数量"的群体里出现仅有 2 次。	**Setiap gabungan bagi** Tahun，Bandar **dalam populasi untuk**"Pada Tahun di Bandar"，monitor dengan JenisMonitor dijualkan dalam Kuantiti"**wujud hanya 2 kali**.

2.1.13.3 Simple External Frequency Constraints on Single Roles

Figure 3.40 shows the English alethic verbalization pattern. The primary readings of the fact types F_1 .. *Fn* are used. Front text makes no difference. Deontic readings are prepended by 'It is obligatory that'.

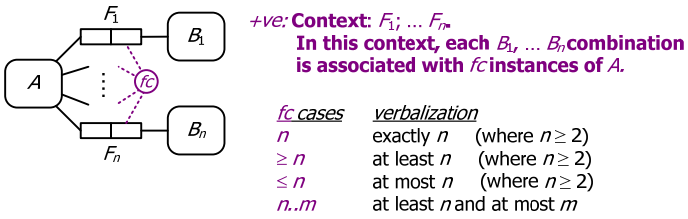


Figure 3.40 External frequency constraints on single roles

The corresponding verbalizations in Bahasa Melayu and Mandarin are shown below.

BM: **Konteks:** F_1,..,F_n.

Dalam Konteks ini，Setiap gabungan bagi B_1,..,B_n **ialah berkaitan dengan** *fc*Δ*A*。

fc case	*verbalization*
n	*hanya n (where n ≥ 2)*
≥ *n*	*sekurang-kurangnya n (where n ≥ 2)*
≤ *n*	*paling banyak n (where n ≥ 2)*
n…m	*sekurang-kurangnya ndan paling banyak m*

Mandarin: 范围：F_1,..,F_n.

在这范围内，每一▲ B_1,..,B_n 的组合关联到 $fc\Delta A$。

fc case	*verbalization*
n	*仅有 n (where n ≥ 2）*
≥ n	*至少 n (where n ≥ 2)*
≤ n	*最多 n (where n ≥ 2)*
n…m	*至少 n 同时最多 m*

Table 3.54 Example verbalizations of external frequency constraints on single roles

English	Mandarin	BM
Context: Enrollment is by Student; Enrollment is in Course. **In this context, each** Student, Course **combination is associated with at most** 2 **instances of** Enrollment.	**范围：**入学报名由学生，入学报名于课程. **在这范围内，每一**个学生，课程**的组合是关联到最多 2** 个入学报名。	**Konteks：** Pendaftaran oleh Pelajar，Pendaftaran bagi Kursus. **Dalam Konteks ini，Setiap gabungan bagi** Pelajar，Kursus **ialah berkaitan dengan paling banyak 2 buah** Pendaftaran.

3.1.14 Ring Constraints

3.1.14.1 Irreflexive Ring Constraints

Both constrained roles are in the same binary predicate and are played by the same object type:

Figure 3.41 shows the English positive alethic and negative alethic verbalization patterns for an irreflexive ring constraint between roles of the same binary predicate where the predicate reading has *no front text*. Pattern (a) is neater in most cases, but may sometimes be mildly ambiguous (e.g. It is obligatory that no Person reviews a paper by itself.), in which case the alternative (b) pattern may be preferred. Ideally the tool should provide both phrasings and let the user choose the best one. Currently, NORMA supports only pattern (b). If the constraint is *deontic*, the positive verbalization is prepended by "it is obligatory that" and the negative verbalization substitutes "forbidden" for "impossible".

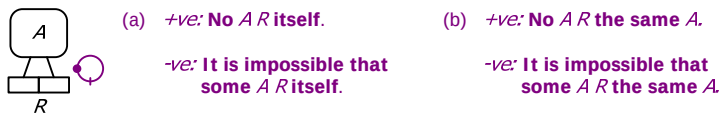


Figure 3.41 Irreflexive ring constraints on a homogeneous binary fact type

The corresponding verbalizations in Bahasa Melayu and Mandarin are shown below.

BM: (a) +ve: **Tiada mana-mana satu Δ *AR* dirinya.**

-ve: **Ia adalah tidak munghin bahawa seseΔ /Sesuatu *A R* dirinya.**

(b) +ve: **Tiada mana-mana satu Δ *A R A* yang sama.**

-ve: **Ia adalah tidak mungkin bahawa seseΔ /Sesuatu *A R A* yang sama.**

Mandarin: (a) +ve: 无任何一Δ*AR* 自己。

-ve: 某一Δ *AR* 自己是不可能的。

(b) +ve: 无任何一Δ*AR* 同一Δ*A*。

-ve: 某一Δ *AR* 同一Δ*A* 是不可能的。

Table 3.55 Example verbalizations of irreflexive ring constraints on homogeneous binaries

English	Mandarin	BM
No Person is a parent of **itself**. **No** Person is a parent of **the same** Person	无任何一个人是自己的父母。 无任何一个人是同一个人的父母。	**Tiada mana-mana satu** Orang ialah IbuBapa bagi **dirinya**. **Tiada mana-mana satu** Orang ialah IbuBapa bagi Orang **yang sama**.

In the very rare case where front-text is present, we use the negative pattern for (b) above, but place the front text just before the existential quantifier ("some").

Both constrained roles are in the same binary predicate but are hosted by different object types:

In this case, at least one of the object types *A* and *B* that host the constrained roles must be a subtype, and both *A* and *B* must have a common type *C*. The same patterns just considered may be used, simply by choosing a predicate reading *R* that starts at *A*. This is always attainable (e.g. for the reading Employee reports to Manager, *A* is Employee, whereas for the reading Manager manages Employee, *A* is Manager). Figure 3.42 shows the typical case with no front text. Since the patterns in English, BM and Mandarin are the same as those just considered, they are not repeated here.

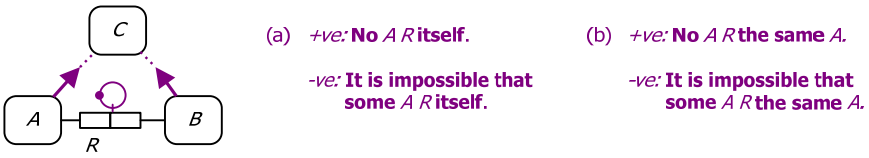


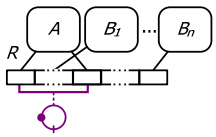
Figure 3.42 Irreflexive ring constraints on a heterogeneous binary fact type

Both constrained roles are in the same ternary or longer predicate, with a predicate reading starting at one of the constrained roles, with each role played by the same object type:

Figure 3.43 shows the English positive alethic and negative alethic verbalization patterns for an irreflexive ring constraint between two roles (not necessarily contiguous) played by *A* in the same *n*-ary predicate, where the predicate reading has *no front text*. It is possible that *A* plays more than 2 roles in the predicate (in which case, *A* is at least one of the B_i). If *A* plays exactly 2 roles, then pattern (a) is neater in most cases, but may sometimes be mildly ambiguous, in which case the alternative (b) pattern may be preferred.

If *A* plays 2 *non-contiguous roles* in the predicate, then pattern (b) must be used (because "itself" may well be ambiguous in this case, e.g. No Person reviewed some Book written by itself). If *A* plays *more than 2 roles* in the predicate, then pattern (b) must be used (because "itself" is ambiguous in this case), and the *A* object variables playing the constrained roles should be subscripted as A_1, with the other *A*'s subscripted sequentially higher (A_2, etc.).

If the constraint is *deontic*, the positive verbalization is prepended by "it is obligatory that" and the negative verbalization substitutes "forbidden" for "impossible". In the very rare case where front-text is present, we use the negative pattern for (b), but place the front text just before the existential quantifier ("some").



(a) *+ve:* **No** *A R* **itself** ... **some** B_1 ... **some** B_n.

-ve: **It is impossible that some** *A R* **itself** ... **some** B_1 ... **some** B_n.

(b) *+ve:* **No** *A R* **the same** *A* ... **some** B_1 ... **some** B_n.

-ve: **It is impossible that some** *A R* **the same** *A* ... **some** B_1 ... **some** B_n.

Figure 3.43 Irreflexive ring constraints on an *n*-ary fact type (basic case)

The corresponding verbalizations in Bahasa Melayu and Mandarin are shown below.

BM: (a) +ve: **Tiada mana-mana satu Δ *AR* dirinya…sese Δ /sesuatu** B_1 **… sese Δ /sesuatu** B_n**.**

-ve: **Ia adalah tidak munghin bahawa seseΔ /Sesuatu *A R* dirinya…sese Δ**

/sesuatu B_1 … sese Δ /sesuatu B_n.

(b) +ve: **Tiada mana-mana satu Δ *A R A* yang sama…sese Δ /sesuatu B_1 … sese Δ /sesuatu B_n.**

-ve: **Ia adalah tidak mungkin bahawa seseΔ /Sesuatu *A R A* yang ama…sese Δ /sesuatu B_1 … sese Δ /sesuatu B_n.**

Mandarin: (a) +ve: **无任何一Δ*AR* 自己。**

-ve: **某一Δ *AR* 自己是不可能的。**

(b) +ve: **无任何一Δ*AR* 同一Δ*A*。**

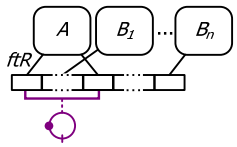
-ve: **某一Δ *AR* 同一Δ*A* 是不可能的。**

Table 3.56 Example verbalizations of irreflexive ring constraints on *n*-aries: basic

English	Mandarin	BM
No Part contains **itself** in **some** Quantity.	无任何一个部件包含自己于某一个数量。	Tiada mana-mana satu buah Bahagian mengandungi dirinya dalam sesuatu bilangan.
No Part contains **the same** Part in **some** Quantity.	无任何一个部件包含同一个部件于某一个数量。	Tiada mana-mana satu buah Bahagian mengandungi Bahagian yang sama dalam sesuatu bilangan.

Both constrained roles come from the same n-ary predicate, with no predicate reading starting at a constrained role, with each role hosted by the same object type:

Figure 3.44 shows the English alethic pattern for other cases, where the predicate may or may not contain front text (square brackets enclose the optional front text).



+ve: None supported

-ve: **It is impossible that *ft* some *A R* the same *A* … some B_1 … some B_n.**

Figure 3.44 Irreflexive ring constraints on an *n*-ary fact type (complex case)

The corresponding verbalizations in Bahasa Melayu and Mandarin are:

BM: **Ia adalah tidak mungkin bahawa *[ft] R* sese Δ/sesuatu B_1 … sese Δ/sesuatu *A*… *A* yang sama… sese Δ/sesuatu B_n**

Mandarin: ***[ft] R* 某一Δ B_1 …某一 *A*…同一Δ *A*…某一Δ B_n是不可能的。**

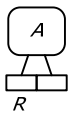
Table 3.57 Example verbalizations of irreflexive ring constraints on *n*-aries: complex

English	Mandarin	BM
It is impossible that on **some** Date **some** Part contained **the same** Part in **some** Quantity.	在某一个日期某一个部件包含同一个部件于某一个数量是不可能的。	Ia adalah tidak mungkin bahawa pada sesuatu Tarikh sesebuah bahagian mengandungi Bahagian yang sama dalam sesuatu Bilangan .

3.1.14.2 Asymmetric Ring Constraints

Both constrained roles are in the same binary predicate and are played by the same object type:

Figure 3.45 shows the English alethic verbalization pattern. If front text exists, it is placed before the first object type term in the pattern.



+ve: **If** A_1 R A_2
then it is impossible that A_2 R A_1.

-ve: None supported

Figure 3.45 Asymmetric ring constraints on a homogeneous binary fact type

The corresponding verbalizations in Bahasa Melayu and Mandarin are:

BM: **Jika** A_1 R A_2,
maka ia adalah tidak mungkin bahawa A_2 R A_1.

Mandarin: 如果 A_1 R A_2，
那么 A_2 R A_1 是不可能的。

Table 3.58 Example verbalizations of asymmetric ring constraints on homogenous binaries

English	Mandarin	BM
If $Part_1$ contains $Part_2$ **then it is impossible that** $Part_2$ contains $Part_1$.	如果部件 $_1$ 包含部件 $_2$，那么部件 $_2$ 包含部件 $_1$ 是不可能。	**Jika** $Bahagian_1$ mengandungi $Bahagian_2$，**maka ia adalah tidak mungkin** $Bahagian_2$ **bagi** mengandungi $Bahagian_1$.

Both constrained roles are in the same binary predicate and are played by different object types:

Figure 3.46 shows the English alethic verbalization pattern. If front text exists, it is placed before the first object type term in the pattern.

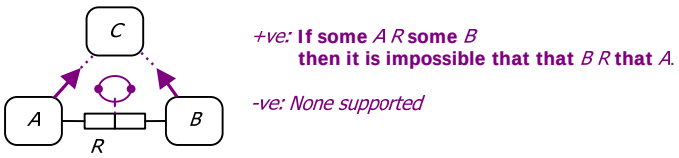


Figure 3.46 Asymmetric ring constraints on a heterogeneous binary fact type

The corresponding verbalizations in Bahasa Melayu and Mandarin are:

BM: **Jika** *A R* **ese** Δ**/sesuatu** *B*,
maka ia adalah tidak mungkin bahawa *B* **itu** *R A* **itu.**

Mandarin: 如果 *A R* 某一Δ*B*,
那么那一Δ*B R* 那一Δ*A* 是不可能的。

Table 3.59 Example verbalizations of asymmetric ring constraints on heterogenous binaries

English	Mandarin	BM
If some Employee reports to **some** Manager **then it is impossible that that** Manager reports to **that** Employee.	如果某一位员工呈报给某一位经理那么那一位经理呈报给那一位员工是不可能的。	Jika seseorang Pekerja melaporkan kepada seseorang Pengurus maka ia adalah tidak mungkin bahawa Pengurus itu melaporkan kepada Pekerja itu.

Both constrained roles come from the same ternary or longer predicate, with both roles played by the same object type:

Figure 3.47 shows the English alethic verbalization pattern. If front text exists, it is placed before the first existential quantifier.

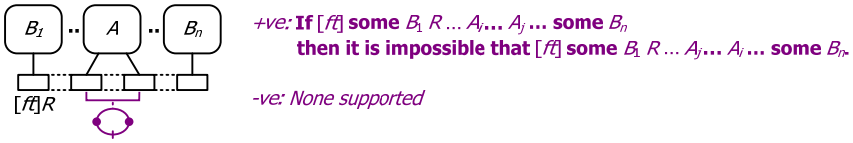


Figure 3.47 Asymmetric ring constraints on an *n*-ary fact type: basic case

The corresponding verbalizations in Bahasa Melayu and Mandarin are:

BM: **Jika** *[ft]***sese** Δ**/sesuatu** $B_1 R...A_i...A_j...$ **sese** Δ**/sesuatu** B_n**,**
maka ia adalah tidak mungkin bahawa *[ft]* **sese** Δ**/sesuatu** $B_1 R...A_i...A_j...$
sese Δ**/sesuatu** B_n

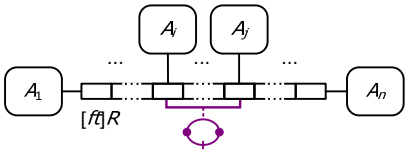
Mandarin: **如果***[ft]***某一**Δ B_1 R...A_i...A_j...**某一**Δ B_n**,**
那么*[ft]* **某一**Δ B_1 R...A_i...A_j...**某一**Δ B_n**是不可能的**。

Table 3.60 Example verbalizations of asymmetric ring constraints on *n*-aries : basic case

English	Mandarin	BM
If $Part_1$ contains $Part_2$ in some Quantity then it is impossible that $Part_2$ contains $Part_1$ in some Quantity.	如果部件 $_1$ 包含部件 $_2$ 于某一个数量那么那一个部件 $_2$ 包含部件 $_1$ 于某一个数量是不可能的。	Jika sesebuah $Bahagian_1$ mengandungi $Bahagian_2$ dalam sesuatu bilangan maka ia adalah tidak mungkin bahawa $Bahagian_2$ mengandungi $Bahagian_1$ dalam sesuatu bilangan.

Both constrained roles come from the same ternary or longer predicate, but are played by different object types.

Figure 3.48 shows the English alethic verbalization pattern. If front text exists, it is placed before the first existential quantifier in each clause.



+ve: **If** [*ft*] **some** A_1 *R* ... **some** A_i **... some** A_j **... some** A_n
then it is impossible that [*ft*] **some** A_1 *R* ... **that** A_j **... that** A_i **... some** A_n.

-ve: None supported

Figure 3.48 Asymmetric ring constraints on an *n*-ary fact type: complex case

The corresponding verbalizations in Bahasa Melayu and Mandarin are:

BM: **Jika** *[ft]***sese Δ/sesuatu** A_1 *R*... **sese Δ/sesuatu** A_i... **sese Δ/sesuatu** A_j... **sese Δ/sesuatu** A_n**, maka ia adalah tidak mungkin bahawa** *[ft]* **sese Δ/sesuatu** A_1 *R*...A_j **itu**...A_i **itu** ... **sese Δ/sesuatu** B_n.

Mandarin: **如果***[ft]***某一Δ** A_1 *R*...**某一Δ** A_i...**某一Δ** A_j...**某一Δ** A_n**,**
那么*[ft]* **某一Δ** A_1 *R*...**那一Δ** A_j...**那一Δ**A_i...**某一Δ** A_n**是不可能的**。

Table 3.61 Example verbalizations of asymmetric ring constraints on *n*-aries : complex case

English	Mandarin	BM
If on **some** Date **some** Man buried **some** Woman **then it is impossible that** on **some** Date **that** Woman buried **that** Man.	如果在某一个日期某一个男人埋葬某一个女人那么在某一个日期那一个女人埋葬那一个男人是不可能的。	Jika pada sesuatu Tarikh seseorang Lelaki mengebumikan seseorang Wanita maka ia adalah tidak mungkin bahawa pada sesuatu Tarikh Wanita itu mengebumikan Lelaki itu.

3.1.14.3 Antisymmetric Ring Constraints

Both constrained roles are in the same binary predicate and are played by the same object type:

Figure 3.49 shows the English alethic verbalization pattern. If front text exists, it is placed before the first object type.

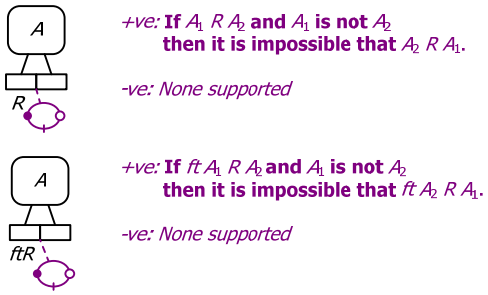


Figure 3.49 Antisymmetric ring constraints on a homogeneuous binary fact type

The corresponding verbalizations in Bahasa Melayu and Mandarin are:

BM: **Jika** A_1 *R* A_2 **dan** A_1 **bukan** A_2
maka ia adalah tidak mungkin bahawa A_2 **itu** *R* A_1**.**

Jika *ft* A_1 *R* A_2 **dan** A_1**bukan** A_2
maka ia adalah tidak mungkin bahawa *ft* A_2 **itu** *R* A_1**.**

Mandarin: 如果 A_1 *R* A_2 同时 A_1 不是 A_2
那么那一ΔA_2 *R* A_1 是不可能的。

如果 *ft* A_1 *R* A_2 同时 A_1 不是 A_2
那么 *ft* 那一Δ A_2 *R*A_1 是不可能的。

Table 3.62 Example verbalizations of antisymmetric ring constraints on a homogeneous binary

English	Mandarin	BM
If Topic$_1$ subsumes Topic$_2$ **and** Topic$_2$ **is not** Topic$_1$ **then it is impossible that** Topic$_2$ subsumes Topic$_1$.	如果题目 $_1$涵括题目 $_2$同时题目 $_2$不是题目 $_1$那么题目 $_2$涵括题目 $_1$是不可能的。	Jika Topik$_1$ meliputi Topik$_2$ dan Topik$_2$ bukan Topik$_1$ maka ia adalah tidak mungkin bahawa Topik$_2$ itu meliputi Topik$_1$.

Both constrained roles are in the same binary predicate but are played by different object types:

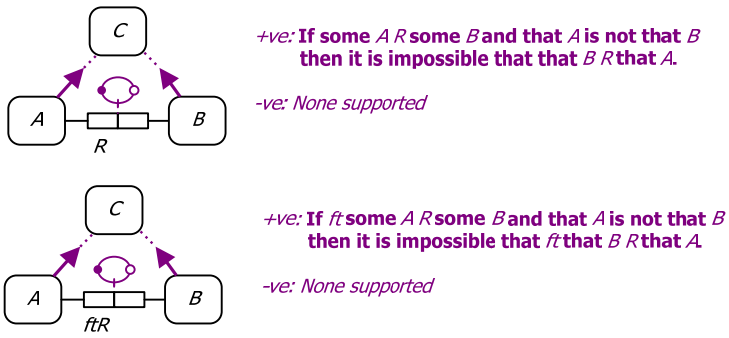


Figure 3.50 Antisymmetric ring constraints on a heterogeneuous binary fact type

Figure 3.50 shows the English alethic verbalization pattern. The corresponding verbalizations in Bahasa Melayu and Mandarin are:

BM: **Jika** A_1 *R* A_2 **dan** A_1 **bukan** A_2
maka ia adalah tidak mungkin bahawa A_2 **itu** *R* A_1**.**

Jiaka *ft* A_1 *R* A_2 **dan** A_1 **bukan** A_2
maka ia adalah tidak mungkin bahawa *ft* A_2 **itu** *R* A_1.

Mandarin: 如果某一Δ*A R* 某一Δ*B* 同时那一Δ*A* 不是那一Δ*B*
那么那一Δ *B R* 那一Δ *A* 是不可能的。

如果 *ft* 某一Δ*A R* 某一Δ*B* 同时那一Δ*A* 不是那一Δ*B*
那么 *ft* 那一Δ *B R* 那一Δ *A* 是不可能的。

Table 3.63 Example verbalizations of antisymmetric ring constraints on a heterogeneous binary

English	Mandarin	BM
If some Team includes **some** Group **and that**Team **is not that** Group **then it is impossible that that** Group includes **that**Team.	如果某一个团队包括某一个组队同时那一个团队不是那一个组队，那么那一个组队包括那一个团队是不可能。	**Jika sese**buah Pasukan mengandungi **ses**buah kumpulan **dan** Pasukan **itu bukan** Kumpulan **itu**，**maka ia adalah tidak mungkin bagi** Kumpulan **itu** mengandungi Pasukan **itu**.

Both constrained roles come from the same ternary or longer predicate, with both roles played by the same object type:

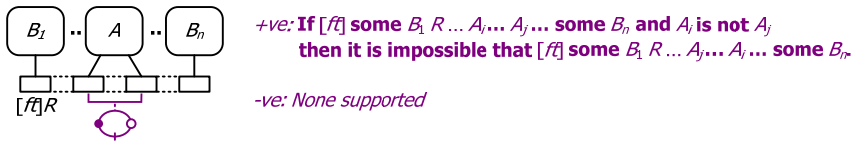


Figure 3.51 Antisymmetric ring constraints on an *n*-ary fact type: basic case

Figure 3.51 shows the English alethic verbalization pattern. The corresponding verbalizations in Bahasa Melayu and Mandarin are:

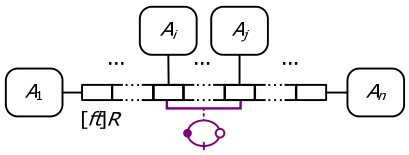
BM: **Jika [*ft*] sese Δ/sesuatu $B_1 R \ldots A_i \ldots A_j \ldots$ sese Δ/sesuatu B_n dan A_i bukan A_j maka ia adalah tidak mungkin bahawa [*ft*]sese Δ/sesuatu $B_1 \ldots A_i \ldots A_j \ldots$ sese Δ/sesuatu B_n.**

Mandarin: **如果[*ft*]某一ΔB_1 $R \ldots A_i \ldots A_j \ldots$某一Δ$B_n$ 同时 A_i 不是 A_j 那么[*ft*] 某一Δ $B1 \ldots A_i \ldots A_j \ldots$某一Δ$B_n$ 是不可能的。**

Table 3.64 Example verbalizations of antisymmetric ring constraints on an *n*-ary fact type: basic case

English	Mandarin	BM
If Set_1 contains Set_2 in **some** Quantity **and** Set_1 **is not** Set_2 **then it is impossible that** Set_2 contains Set_1 in **some** Quantity.	如果部件 $_1$ 包含部件 $_2$ 于某一个数量同时部件 $_1$ 不是部件 $_2$ 那么部件 $_2$ 包含部件 $_1$ 于某一个数量是不可能的。	Jika Set1 mengandungi Set2 dalam sesuatu bilangan dan Set1 bukan Set2 maka ia adalah tidak mungkin bahawa Set2 mengandungi Set1 dalam sesuatu bilangan.

Both constrained roles come from the same ternary or longer predicate, but are played by different object types:



+ve: **If** [*ft*] **some** A_1 R … **some** A_i **… some** A_j **… some** A_n **and that** A_i **is not that** A_j
then it is impossible that [*ft*] **some** A_1 R … **that** A_j **… that** A_i **… some** A_n.

-ve: None supported

Figure 3.52 Antisymmetric ring constraints on an *n*-ary fact type: complex case

Figure 3.52 shows the English alethic verbalization pattern. The corresponding verbalizations in Bahasa Melayu and Mandarin respectively are:

Jika[*ft*]sese Δ/sesuatu A_1 R… **sese Δ/sesuatu** A_i… **sese Δ/sesuatu** A_j… **sese Δ/sesuatu** A_n …**dan**A_i **itu** ΔA_j **itu**
maka ia adalah tidak mungkin bahawa [*ft*] sese Δ/sesuatu A_1 R …A_j **itu** … A_j **itu** … **sese Δ/sesuatu** A_n**.**

如果[*ft*]某一ΔA_1 R…**某一Δ** A_i…**某一Δ** A_j…**某一Δ** A_n …**同时那一Δ**A_i**不是那一Δ**A_j
那么[*ft*] 某一Δ A_1 R …**那一Δ**A_j…**那一Δ** A_j…**某一Δ** A_n**是不可能的。**

Table 3.65 Example verbalizations of antisymmetric ring constraints on an *n*-ary fact type: complex case

English	Mandarin	BM
If on **some** Date **some** Man buried **some** Soldier **and that** Man **is not that** Soldier **then it is impossible that** on **some** Date **that** Soldier buried **that** Man.	如果某一个日期某一个男人埋葬某一位士兵同时那一个男人不是那一位士兵那么在某一个日期那一位士兵埋葬那一个男人是不可能的。	Jika pada sesuatu Tarikh seseorang Lelaki mengebumikan seseorang Askar dan Lelaki itu bukan Askar maka ia adalah tidak mungkin bahawa pada sesuatu Tarikh Askar itu mengebumikan Lelakit itu.

3.1.14.4 Intransitive Ring Constraints

Both constrained roles are in the same binary predicate and are played by the same object type:

Figure 3.53 Intransitive ring constraints on a homogeneous binary fact type

Figure 3.53 shows the English alethic verbalization pattern. The corresponding verbalizations in Bahasa Melayu and Mandarin respectively are:

Jika A_1 *R* A_2 **dan** A_2 *R* A_3
maka ia adalah tidak mungkin bahawa A_1 *R* A_3.

Jika *ft* A_1 *R* A_2 **dan** A_2 *R* A_3
maka *ft* **ia adalah tidak mungkin bahawa** A_1 *R* A_3.

如果 A_1 *R* A_2 同时 A_2 *R* A_3
那么 A_1 *R* A_3 是不可能的。

如果 *ft*A_1 *R* A_2 同时 A_2 *R* A_3
那么 *ft*A_1 *R* A_3 是不可能的。

Table 3.66 Example verbalizations of intransitive ring constraints on a homogeneous binary fact type

English	Mandarin	BM
If $Person_1$ is the father of $Person_2$ **and** $Person_2$ is the father of $Person_3$ **then it is impossible that** $Person_1$ is the father of $Person_3$.	如果人 $_1$ 是人 $_2$ 的父亲同时人 $_2$ 是人 $_3$ 的父亲，那么人 $_1$ 是人 $_3$ 的父亲是不可能。	**Jika** $Orang_1$ ialah ayah bagi $Orang_2$ **dan** $Orang_2$ ialah ayah bagi $Orang_3$，**maka ia adalah tidak mungkin** $Orang_1$ ialah ayah bagi **bagi** $Orang_3$.

Both constrained roles are in the same binary predicate but are played by different object types:

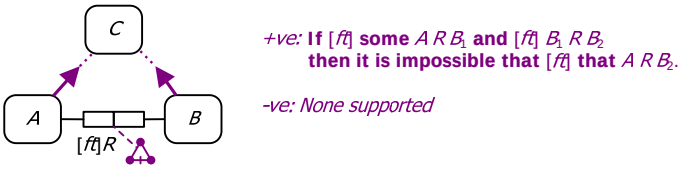


Figure 3.54 Intransitive ring constraints on a heterogeneous binary fact type

Figure 3.54 shows the English alethic verbalization pattern. The corresponding verbalizations in Bahasa Melayu and Mandarin respectively are:

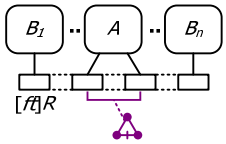
Jika[*ft*] sese Δ/sesuatu *A R* B_1 **dan [*ft*]**B_1 *R* B_2
maka ia adalah tidak mungkin bahawa [*ft*] *A* **itu** *R* B_2.

如果[*ft*] 某一Δ *A R* B_1 **同时[*ft*]**B_1 *R* B_2
那么[*ft*]那一Δ *A R* B_2**是不可能的**。

Table 3.67 Example verbalizations of intransitive ring constraints on a heterogeneous binary fact type

English	Mandarin	BM
If out of respect **some** Man buried $Soldier_1$ **and** out of respect $Soldier_1$ buried $Soldier_2$ **then it is impossible that** out of respect **that** Man buried $Soldier_2$.	如果处于尊重某一个男人埋葬士兵 $_1$ 同时处于尊重士兵 $_1$ 埋葬士兵 $_2$ 那么处于尊重那一个男人埋葬士兵 $_2$ 是不可能的。	Jika disebabkan penghormatan seseorang Lelaki mengebumikan $Askar_1$ dan disebabkan penghormatan $Askar_1$ mengebumikan $Askar_2$ maka ia adalah tidak mungkin bahawa disebabkan penghormatan Lelaki itu mengebumikan $Askar_2$.

Both constrained roles come from the same ternary or longer predicate, with both roles played by the same object type:



+ve: **If** [*ft*] **some** B_1 *R* … A_i **…** A_j **… some** B_n **and** [*ft*] **some** B_1 *R* … A_j **…** A_k **… some** B_n
then it is impossible that [*ft*] **some** B_1 *R* … A_i **…** A_k **… some** B_n.

-ve: None supported

Figure 3.55 Intransitive ring constraints on an *n*-ary fact type: basic case

Figure 3.55 shows the English alethic verbalization pattern. The corresponding verbalizations in Bahasa Melayu and Mandarin respectively are:

Jika[ft] sese Δ/sesuatu B_1 *R*… A_i… A_j… **sese Δ/sesuatu** B_n **dan[***ft***] sese Δ/sesuatu** B_1 *R*… A_i… A_j… **sese Δ/sesuatu** B_n
maka ia adalah tidak mungkin bahawa [*ft***] sese Δ/sesuatu** B_1 *R*… A_i… A_j… **sese Δ/sesuatu** B_n.

如果[*ft***] 某一Δ** B_1 *R*… A_i… A_j…**某一Δ** B_n **同时[***ft***] 某一Δ** B_1 *R*… A_i… A_j…**某一Δ** B_n
那么[*ft***] 某一Δ** B_1 *R*… A_i… A_j…**某一Δ** B_n **是不可能的**。

Table 3.68 Example verbalizations of intransitive ring constraints on an *n*-ary fact type: basic case

English	Mandarin	BM
If on **some** Date Person$_1$ buried Person$_2$ **and** on **some** Date Person$_2$ buried Person$_3$ **then it is impossible that** on **some** Date Person$_1$ buried Person$_3$.	如果在某一个日期人 $_1$ 埋葬人 $_2$ 同时在某一个日期人 $_2$ 埋葬人 $_3$ 那么在某一个日期人 $_1$ 埋葬人 $_3$ 是不可能的。	Jika pada sesuatu Tarikh Orang$_1$ mengebumikan Orang$_2$ dan pada sesuatu Tarikh Orang$_2$ mengebumikan Orang$_3$ maka ia adalah tidak mungkin bahawa pada sesuatu Tarikh Orang$_1$ mengebumikan Orang$_3$.

Both constrained roles come from the same ternary or longer predicate, but are played by different object type:

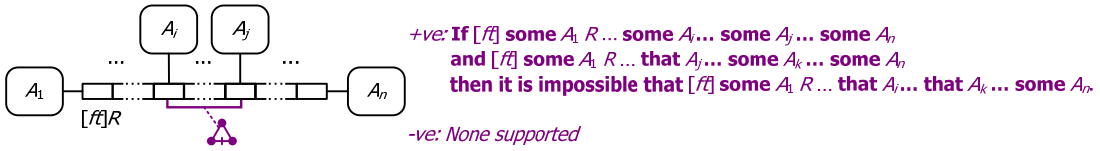


Figure 3.56 Intransitive ring constraints on an *n*-ary fact type: complex case

Figure 3.56 shows the English alethic verbalization pattern. The corresponding verbalizations in Bahasa Melayu and Mandarin respectively are:

> **Jika [*ft*] sese Δ/sesuatu A_1 *R*… sese Δ/sesuatu A_i… sese Δ/sesuatu A_j… sese Δ/sesuatu A_n dan [*ft*] sese Δ/sesuatu A_1 *R*… A_j itu… sese Δ/sesuatu A_k… sese Δ/sesuatu A_n maka ia adalah tidak mungkin bahawa [*ft*] sese Δ/sesuatu A_1 *R*… A_i itu … A_k itu … sese Δ/sesuatu A_n。**
>
> 如果[*ft*] 某一Δ A_1 *R*…某一ΔA_i…某一ΔA_j…某一Δ A_n 同时[*ft*] 某一Δ A_1 *R*…那一ΔA_j…某一ΔA_k…某一Δ A_n 那么[*ft*] 某一Δ A_1 *R*…那一ΔA_i…那一ΔA_k…某一Δ A_n 是不可能的。

Table 3.69 Example verbalizations of intransitive ring constraints on an *n*-ary fact type: complex case

English	Mandarin	BM
If on **some** Date **some** Man buried **some** $Soldier_1$ **and** on **some** Date **that** $Soldier_1$ buried **some** $Soldier_2$ **then it is impossible that** on **some** Date **that** Man buried **that** $Soldier_2$.	如果在某一个日期某一个男人埋葬某一个士兵 $_1$ 同时在某一个日期那一位士兵 $_1$ 埋葬某一个士兵 $_2$ 那么在某一个日期那一个男人埋葬那一位士兵 $_2$ 是不可能的。	Jika pada sesuatu Tarikh seseorang Lelaki mengebumikan seseorang $Askar_1$ dan pada sesuatu Tarikh $Askar_1$ itu mengebumikan seseorang $Askar_2$ maka ia adalah tidak mungkin bahawa pada sesuatu Tarikh Lelaki itu mengebumikan $Askar_2$ itu.

3.1.14.5 Strongly Intransitive Ring Constraints

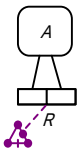


Figure 3.57 Strongly intransitive ring constraints on a homogeneous binary fact type

(Eng) If A_1 R some A_2 then it is no true that A_1 is indirectly related to A_2 by repeatedly applying this fact type.

(BM) Jika A_1 R sese∆/sesuatu A_2 maka ia adalah tidak benar bahawa A_1 adalah secara tidak langsung berkaitan dengan A_2 dengan berulangan menuruti jenis fakta ini.

(Mandarin) 如果 A_1 R 某一∆ A_2 那么那一∆ A_1 反复地不间断关联到 A_2 是不正确的。

English	Mandarin	BM
If Student$_1$ is class monitor for **some** Student$_2$ **then it is not true that** Student$_1$ **is indirectly related to** Student$_2$ **by repeatedly applying this fact type**.	如果学生 $_1$ 是某一位学生 $_2$ 的班长 那么那一位学生反复地不间断关联到学生 $_2$ 是不正确的。	Jika Pelajar$_1$ adalah kelas monitor bagi seseorang Pelajar$_2$ Maka ia adalah tidak benar bahawa Pelajar$_1$ adalah secara tidak langsung berkaitan dengan Pelajar$_2$ dengan berulangan menuruti jenis fakta ini.

3.1.14.6 Acycic Ring Constraints

Both constrained roles are in the same binary predicate and are played by the same object type:

Figure 3.58 Acyclic ring constraints on a homogeneous binary fact type

Figure 3.58 shows the English alethic verbalization pattern. The corresponding verbalizations in Bahasa Melayu and Mandarin respectively are:

Tiada mana-mana satu Δ *A* **boleh kitar semula kepada dirinya melalui salah satu daripada lebih penyusuran dengan** *A R A.*

Tiada mana-mana satu Δ *A* **boleh kitar semula kepada dirinya melalui salah satu daripada lebih penyusuran dengan** *ft A R A.*

无任何一Δ*A* 可以通过更多的 *A R A* 穿越循环回到自己。

无任何一Δ*A* 可以通过更多的 *ft A R A* 穿越循环回到自己。

Table 3.70 Example verbalizations of acyclic ring constraints on a homogeneous binary fact type

English	Mandarin	BM
No Person **may cycle back to itself via one of more traversals through** Person **is a parent of** Person.	无任何一个人可以通过更多的‘人是人的父母’穿越循环回到自己。	**Tiada mana-mana satu** Orang **boleh kitar semula kepada dirinya melalui salah satu daripada lebih penyusuran dengan** Orang **ialah** IbuBapa **bagi** Orang.

Both constrained roles come from the same n-ary predicate, with both roles played by the same object type, and with no other roles played by that object type:

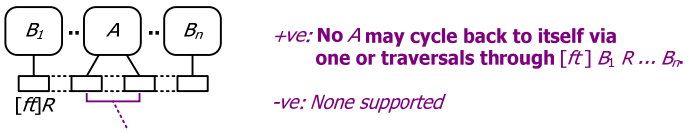


Figure 3.59 Acyclic ring constraints on an *n*-ary fact type: basic case

Figure 3.59 shows the English alethic verbalization pattern. The corresponding verbalizations in Bahasa Melayu and Mandarin respectively are:

> **Tiada mana-mana satu** Δ *A* **boleh kitar semula kepada dirinya melalui salah satu daripada lebih penyusuran dengan [*ft*]** B_1 *R*… B_n.
>
> 无任何一Δ*A* 可以通过更多的**[*ft*]** B_1 *R*… B_n 穿越循环回到自己。

Table 3.71 Example verbalizations of acyclic ring constraints on an *n*-ary fact type: basic case

English	Mandarin	BM
No Part **may cycle back to itself via one of more traversals through** Part contains Part in Quantity.	无任何一个部件可以**可以通过更多的**‘部件包含部件于数量’ **穿越循环回到自己。**	**Tiada mana-mana satu** Orang **boleh kitar semula kepada dirinya melalui salah satu daripada lebih penyusuran dengan** Bahagian mengandungi Bahagian dalam Kuantiti.

Both constrained roles come from the same n-ary predicate, with both roles played by the same object type, and with at least one other role played by that object type:

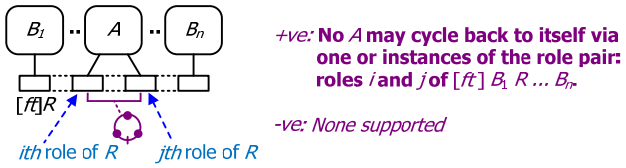


Figure 3.60 Acyclic ring constraints on an *n*-ary fact type: complex case

Figure 3.60 shows the English alethic verbalization pattern. The corresponding verbalizations in Bahasa Melayu and Mandarin respectively are:

> **Tiada mana-mana satu** Δ *A* **boleh kitar semula kepada dirinya melalui salah satu daripada lebih penyusuran dengan pasangan peranan: peranan i dan peranan j bagi [*ft*]** B_1 *R*… B_n.
>
> 无任何一Δ*A* 可以通过更多的角色配对：角色 **i** 和 **j** 的 **[*ft*]** B_1 *R*… B_n 穿越循环回到自己。

Table 3.72 Example verbalizations of acyclic ring constraints on an *n*-ary fact type: complex case

English	Mandarin	BM
No Person **may cycle back to itself via one of more instances of the role pair:roles** 1 **and** 3 **of** Person introduced Person to Person on Date.	无任何一个人**可以通过更多的角色配对：角色 1 和 3 的人介绍人给人于日期来穿越循环回到自己。**	**Tiada mana-mana satu** Orang **boleh kitar semula kepada dirinya melalui salah satu daripada lebih penyusuran dengan** pasangan peranan: peranan 1 dan 3 bagi Orang memperkenalkan Orang kepada Orang pada Tarikh.

3.1.14.7 Reflexive Ring Constraints

In this thesis we restrict our coverage of reflexive ring constraints to the most common case (locally reflexive constraints on homogeneous binaries). For discussion of other kinds of reflexive constraints see [17]. A ring predicate *R* is locally reflexive if and only if $\forall x \forall y(xRy \rightarrow xRx)$. Figure 3.61 shows the English alethic verbalization pattern.

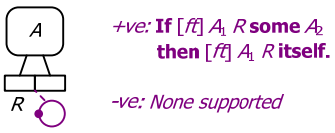


Figure 3.61 Reflexive ring constraints on a homogeneous binary.

The corresponding verbalizations in Bahasa Melayu and Mandarin respectively are:

The corresponding verbalizations in Bahasa Melayu and Mandarin are:

BM: **Jika** *[ft]* A_1 *R* **sese**Δ **/sesuatu** A_2 **,**
maka *[ft]*A_1 *R* **sendiri.**

Mandarin: 如果*[ft]*A 1 R 某一Δ A 2 ,
那么*[ft]*A 2 R 自己。

Table 3.73 Example verbalizations of reflexive ring constraints

English	Mandarin	BM
If Person$_1$ knows **some** Person$_2$ **then** Person$_1$ knows **itself**.	如果人 1 知晓人 2 那么人 1 知晓自己。	Jika Orang$_1$ mengetahui Orang$_2$ maka Orang$_1$ mengetahui sendiri.

3.1.14.8 Symmetric Ring Constraints

Both constrained roles are in the same binary predicate and are played by the same object type:

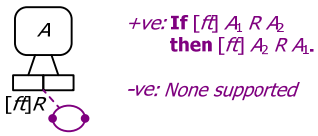


Figure 3.62 Symmetric ring constraints on a homogeneous binary.

Figure 3.62 shows the English alethic verbalization pattern. The corresponding verbalizations in Bahasa Melayu and Mandarin respectively are:

Jika *ft* A_1 *R* A_2
maka *ft* A_2 *R* A_1。

如果 *ft* A_1 *R* A_2
那么 *ft* A_2 *R* A_1。

Table 3.74 Example verbalizations of symmetric ring constraints on homogeneous binaries

English	Mandarin	BM
If $Country_1$ borders $Country_2$ **then** $Country_2$ borders $Country_1$.	如果国家 $_1$ 边界于国家 $_2$ 那么国家 $_2$ 边界于国家 $_1$。	Jika $Negara_1$ bersempedan dengan $Negara_2$ maka $Negara_2$ bersempedan dengan $Negara_1$.

Both constrained roles are in the same binary predicate but are played by different object types:

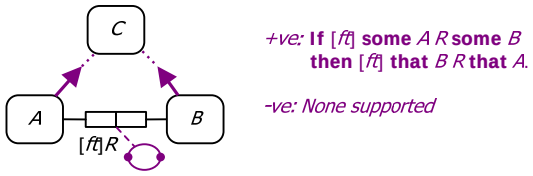


Figure 3.63 Symmetric ring constraints on a heterogeneous binary.

Figure 3.63 shows the English alethic verbalization pattern. The corresponding verbalizations in Bahasa Melayu and Mandarin respectively are:

Jika *ft* **sese**Δ **/sesuatu** *A R* **sese**Δ **/sesuatu** *B*
maka *ft B* **itu** *R A* **itu**.

如果 *ft* 某一Δ*A R* 某一Δ *B*
那么 *ft* 那一Δ *BR* 那一Δ *A*。

Table 3.75 Example verbalizations of symmetric ring constraints on heterogeneous binaries

English	Mandarin	BM
If some Man is married to **some** Woman **then that**Woman is married to **that** Man.	如果某一个男人与某一个女人结婚， 那么那一个女人与那一个男人结婚。	**Jika sese**orang Lelaki berkahwin dengan **sese**orang Wanita，**maka** Wanita **itu** berkahwin dengan Lelaki **itu**.

Both constrained roles come from the same ternary or longer predicate, with both roles played by the same object type:

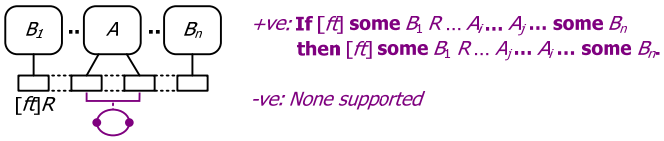


Figure 3.64 Symmetric ring constraints on an *n*-ary fact type: basic case

Figure 3.64 shows the English alethic verbalization pattern. The corresponding verbalizations in Bahasa Melayu and Mandarin respectively are:

Jika *ft* **sese** Δ**/sesuatu** B_1 *R*…A_i…A_j **sese** Δ**/sesuatu** B_n
maka *ft* **sese** Δ**/sesuatu** B_1 *R*…A_j…A_i **sese** Δ**/sesuatu** Δ B_n.

如果 *ft* **某一**ΔB_1 *R*…A_i…A_j **某一**Δ B_n
那么 *ft* **某一**ΔB_1 *R*…A_j…A_i **某一**Δ B_n。

Table 3.76 Example verbalizations of symmetric ring constraints on an *n*-ary fact type: basic case

English	Mandarin	BM
If $Country_1$ borders $Country_2$ under **some** Treaty **then** $Country_2$ borders $Country_1$ under **some** Treaty.	如果国家 $_1$ 边界于国家 $_2$ 在某一个条约下 那么国家 $_2$ 边界于国家 $_1$ 在某一个条约下。	Jika $Negara_1$ bersempedan dengan $Negara_2$ di bawah sesebuah Perjanjian maka $Negara_2$ bersempedan dengan $Negara_1$ di bawah sesebuah Perjanjian.

Both constrained roles come from the same ternary or longer predicate, but are played by different object types:

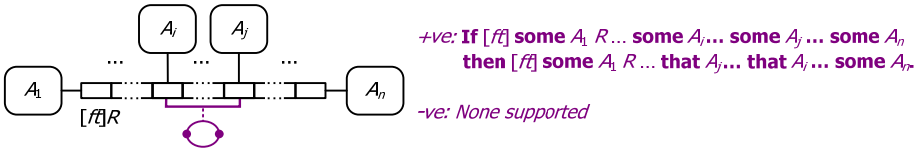


Figure 3.65 Symmetric ring constraints on an *n*-ary fact type: complex case

Figure 3.65 shows the English alethic verbalization pattern. The corresponding verbalizations in Bahasa Melayu and Mandarin respectively are:

Jika *ft* **sese** Δ**/sesuatu** A_1 *R*… **sese** Δ**/sesuatu** A_i… **sese** Δ**/sesuatu** A_j **sese** Δ**/sesuatu** A_n
maka *ft* **sese** Δ**/sesuatu** A_1 *R*… **sese** Δ**/sesuatu** A_j… **sese** Δ**/sesuatu** A_i **sese** Δ**/sesuatu** Δ A_n.

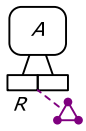
如果 *ft* 某一ΔA_1 *R*…某一ΔA_i…某一ΔA_j 某一Δ A_n
那么 *ft* 某一ΔA_1 *R*…某一ΔA_j…某一ΔA_i 某一Δ A_n。

Table 3.77 Example verbalizations of symmetric ring constraints on an *n*-ary fact type: complex case

English	Mandarin	BM
If on **some** Date **some** Man married **some** Woman **then** on **some** Date **that** Woman married **that** Man.	如果在某一个日期某一个男人与某一个女人结婚那么在某一个日期那一个女人与那一 个男人结婚。	Jika pada sesuatu Tarikh seseorang Lelaki berkahwin dengan seseorang Wanita maka pada sesuatu Tarikh Wanita itu berkahwin dengan Lelaki itu.

3.1.14.9 Transitive Ring Constraints

In this thesis we restrict our coverage of transitive ring constraints to the most common case (transitive constraints on homogeneous binaries). A ring predicate *R* is *transitive* if and only if $\forall x,y,z[(xRy \ \& \ yRz) \rightarrow xRz]$. Figure 3.66 shows the English alethic verbalization pattern.



+ve: **If** [*ft*] A_1 *R* A_2 **and** A_2 *R* A_3
then [*ft*] A_1 *R* A_3.

-ve: None supported

Figure 3.66 Transitive ring constraints on a homogeneous binary fact type

The corresponding verbalizations in Bahasa Melayu and Mandarin respectively are:

Jika *[ft]* A_1 *R* A_2 **dan** A_2 *R* A_3
maka *[ft]* A_1 *R* A_3.

如果*[ft]* A_1 *R* A_2 **同时** A_2 *R* A_3
那么*[ft]* A_1 *R* A_3。

Table 3.78 Example verbalizations of transitive ring constraints on a homogeneous binary fact type

English	Mandarin	BM
If $Person_1$ is an ancestor of $Person_2$ **and** $Person_2$ is an ancestor of $Person_3$ **then** $Person_1$ is an ancestor of $Person_3$.	**如果**人 $_1$ 是人 $_2$ 的祖先**同时**人 $_2$ 是人 $_3$ 的祖先，**那么**人 $_1$ 是人 $_3$ 的祖先。	**Jika**$Orang_1$ ialah moyang bagi $Orang_2$ **maka** $Orang_2$ ialah moyang bagi $Orang_3$，**maka** $Orang_1$ ialah moyang bagi $Orang_3$

3.1.14.10 Combinations of Ring Constraints

Various combinations of ring constraints may be declared [17]. Each ring constraint combination is verbalized simply by verbalizing each individual ring constraint in the combination. As the patterns for each of these have already been covered, we do not repeat them here. However, in this section we illustrate the two combinations depicted in Figure 3.67: (a) asymmetric and intransitive; (b) acyclic and intransitive.

Figure 3.67 Two of the many possible ring constraint combinations

Table 3.79 Example verbalizations of the ring constraint combinations in Figure 3.67

Case	English	Mandarin	BM
(a) Asymmetric and intransitive	**If** Person$_1$ is the father of Person$_2$ **then it is impossible that** Person$_2$ is the father of Person$_1$. **If** Person$_1$ is the father of Person$_2$ **and** Person$_2$ is the father of Person$_3$ **then it is impossible that** Person$_1$ is the father of Person$_3$.	如果人$_1$是人$_2$的父亲那么人$_2$是人$_1$的父亲是不可能的。 如果人$_1$是人$_2$的父亲同时人$_2$是人$_3$的父亲那么人$_1$是人$_3$是不可能的。	Jika Orang$_1$ ialah ayah bagi Orang$_2$ maka ia adalah tidak mungkin bahawa Orang$_2$ ialah ayah bagi Orang$_1$. Jika Orang$_1$ ialah ayah bagi Orang$_2$ dan Orang$_2$ ialah ayah bagi Orang$_3$ aka ia adalah tidak mungkin bahawa Orang$_1$ ialah ayah bagi Orang$_3$.
(b) Acyclic and intransitive	**No** Person **may cycle back to itself via one of more traversals through** Person is a parent of Person. **If** Person$_1$ is the father of Person$_2$ **and** Person$_2$ is the father of Person$_3$**then it is impossible that** Person$_1$ is the father of Person$_3$.	无任何一个人可以通过更多的‘人是人的父母’穿越循环回到自己。 如果人$_1$是人$_2$的父亲同时人$_2$是人$_3$的父亲那么人$_1$是人$_3$是不可能的.	**Tiada mana-mana satu** Orang **boleh kitar semula kepada dirinya melalui salah satu daripada lebih penyusuran dengan** Orang ialah IbuBapa bagi Orang. Jika Orang$_1$ ialah ayah bagi Orang$_2$ dan Orang$_2$ ialah ayah bagi Orang$_3$ aka ia adalah tidak mungkin bahawa Orang$_1$ ialah ayah bagi Orang$_3$.

3.1.15 Subtyping

Subtyping connections:

Figure 3.68 shows the English alethic verbalization pattern for a subtyping connection. As an alternative, "some" may be used instead of "an instance of".

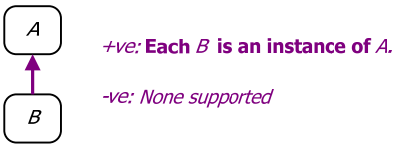


Figure 3.68 Subtyping connection

The corresponding verbalizations in Bahasa Melayu and Mandarin respectively are:

Setiap *B* **adalah sejenis** *A*

每一Δ*B* 是 *A* 的类型。

Table 3.80 Example verbalizations of a subtyping connection

English	Mandarin	BM
Each Dog **is an instance of** Animal**.**	每一只狗是动物的类型。	Setiap anjing adalah sejenis haiwan.

Exclusion constraints and inclusive-or constraints may be applied to a set of two or more subtyping connections with a common supertype. If both inclusive-or and exclusion constraints apply to the same set of subtyping connections, this is displayed as a single exclusive-or constraint.

Exclusion constraint:

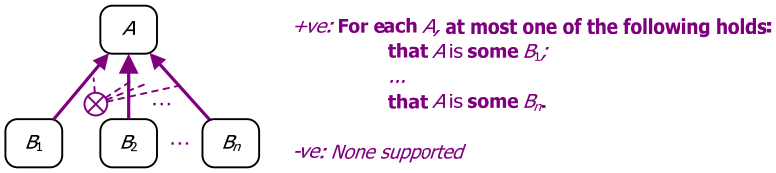


Figure 3.69 Exclusion constraint over subtypes

Figure 3.69 shows the English alethic verbalization pattern. The corresponding verbalizations in Bahasa Melayu and Mandarin respectively are:

Setiap *A*, **memiliki paling banyak satu daripada berikut:**

A **itu adalah sese** Δ**/sesuatu** B_1;

...

A **itu adalah sese** Δ**/sesuatu** B_n.

每一Δ *A*, **持有以下最多一个条件:**

那一Δ *A* **是某一**Δ B_1;

...

那一Δ *A* **是某一**Δ B_n。

Table 3.81 Example verbalizations of subtype exclusion constraints

English	Mandarin	BM
For each Person **at most one of the following holds:** **that** Person **is some** SinglePerson; **that** Person **is some** MarriedPerson; **that** Person **is some** WidowedPerson; **that** Person **is some** DivorcedPerson.	对于每一个人持有以下最多一项条件: 那一个人是某一个单身人士; 那一个人是某一个已婚人士; 那一个人是某一个丧偶人士; 那一个人是某一个离异人士。	Bagi setiap Orang memiliki paling banyak satu daripada berikut: Orang itu ialah seseOrangBujang; Orang itu ialah seseOrangBerkahwin; Orang itu ialah seseOrangJanda; Orang itu ialah seseOrangBercerai.

Inclusive-or constraint:

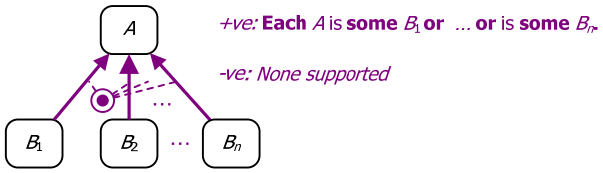


Figure 3.70 Inclusive-or constraint for subtyping

Figure 3.70 shows the English alethic verbalization pattern. The corresponding verbalizations in Bahasa Melayu and Mandarin respectively are:

Setiap *A* **adalah sese** Δ**/sesuatu** B_1 **atau … atau adalah sese** Δ**/sesuatu** B_n **.**

每一Δ *A* **是某一**Δ B_1 **或…或是某一**Δ B_n。

Table 3.82 Example verbalizations of inclusive-or constraints for subtyping

English	Mandarin	BM
Each Person is **some** SinglePerson**or** is **some** MarriedPerson **or** is **some** WidowedPerson **or** is **some** DivorcedPerson.	每一个人是某一个单身人士 或 那一个人是某一个已婚人士 或 那一个人是某一个丧偶人士 或 那一个人是某一个离异人士。	Orang itu ialah seseOrangBujang atau ialah seseOrangBerkahwin atau ialah seseOrangJanda atau ialah seseOrangBercerai;

Exclusive-or constraint:

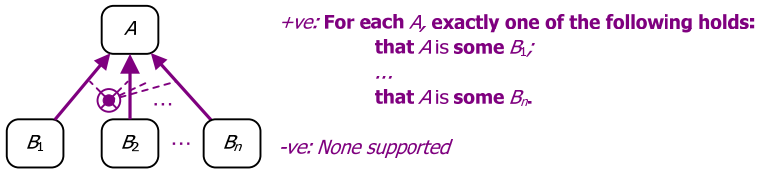


Figure 3.71 Exclusive-or constraint for subtyping

Figure 3.71 shows the English alethic verbalization pattern. The corresponding verbalizations in Bahasa Melayu and Mandarin respectively are:

Setiap *A*, memiliki hanay satu daripada berikut:

***A* itu adalah sese Δ/sesuatu B_1;**

...

***A* itu adalah sese Δ/sesuatu B_n.**

每一Δ *A*, 持有以下仅有一个条件:

那一Δ *A* 是某一Δ B_1;

...

那一Δ *A* 是某一Δ B_n。

Table 3.83 Example verbalizations of exclusive-or constraints for subtyping

English	Mandarin	BM
For each Person **exactly one of the following holds:** **that** Person is **some** SinglePerson; **that** Person is **some** MarriedPerson; **that** Person is **some** WidowedPerson; **that** Person is **some** DivorcedPerson.**some** DivorcedPerson.	对于每一个人持有以下仅有一项条件: 那一个人是某一个单身人士; 那一个人是某一个已婚人士; 那一个人是某一个丧偶人士; 那一个人是某一个离异人士;	Bagi setiap Orang memiliki hanya satu daripada berikut: Orang itu ialah seseOrangBujang; Orang itu ialah seseOrangBerkahwin; Orang itu ialah seseOrangJanda; Orang itu ialah seseOrangBercerai;

3.1.16 Value-Comparison Constraints

A *value-comparison constraint* compares one value with another using one of six comparison operators: <, ≤, >, ≥, =, ≠. The verbalizations of these operators are shown in Table 3.84.

Table 3.84 Verbalizations of value comparison operators

Operator	*English Verbalization*	*BM Verbalization*	*Mandarin Verbalization*
<	**is less than**	**adalah kurang daripada**	是少于
≤	**is less than or equal to**	**adalah kurang daripada atau sama dengan**	是少于或相同于
>	**is greater than**	**adalah lebih besar daripada**	是大于
≥	**is greater than or equal to**	**adalah lebih besar daripada atau sama dengan**	是大于或相同于
=	**equals**	**sama dengan**	相同于
≠	**is not equal to**	**adalah tidak sama dengan**	不是相同于

Case with two binary predicate readings from the join object ypye:

Figure 3.72 shows the English verbalization pattern for the ≤ operator for the most common cases. For the other operators, use the operator verbalization shown in Table 3.84. Deontic modal operators and explicit alethic modality operators are handled in the usual way.

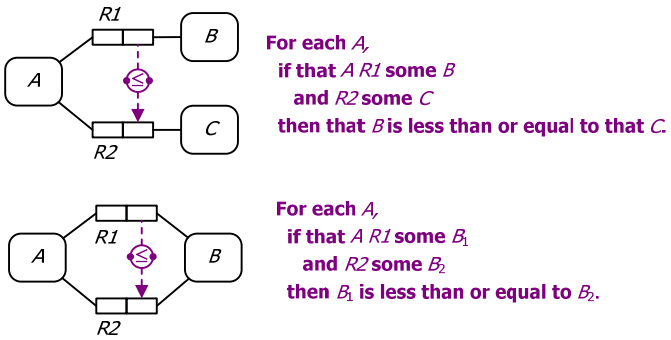


Figure 3.72 Value-comparison constraints for the most common cases

The corresponding verbalizations in Bahasa Melayu and Mandarin respectively are:

Bagi setiap *A R1* **sese∆/sesuatu** *B* **dan** *R2* **sese∆/sesuatu** *C B* **itu adalah kurang daripada atau sama dengan** *C* **itu.**

对于每一∆ *R1* **某一∆** *B* **和** *R2* **某一∆** *C* **的** *A*
那一∆ *B* **是少于或相同于那一∆***C*。

Table 3.85 Example verbalizations of value comparison constraints: common cases

English	BM	Mandarin
For each Employee, **if that** Employee was born on **some** $Date_1$ **and** was hired on **some** $Date_2$ **then** $Date_1$ **is less than** $Date_2$.	对于每一位职员，如果那一位职员出生于某一个日期 $_1$ 同时被雇佣于某一个日期 $_2$ 那么日期 $_1$ 是小于日期 $_2$。	Bagi setiap Pekerja, jika Pekerja itu dilahirkan pada sesuatu $Tarikh_1$ dan diupah pada $Tarikh_2$ maka $Tarikh_1$ adalah kurang daripada $Tarikh_2$.
For each Project, **if that** Project started on **some** $Date_1$ **and** ended on **some** $Date_2$, **then** $Date_2$ **is greater than or equal to** $Date_1$.	对于每一项计划案， 如果那一项计划案开始与某一个日期 $_1$ 同时结束与某一个日期 $_2$， 那么日期 $_2$ 大于或相于日期 $_1$。	Bagi setiap Peojek, jika Projek itu bermula pada sesuatu $Tarikh_1$, dan berakhir pada sesuatu $Tarikh_2$, maka $Tarikh_2$ adalah besar daripada atau sama dengan $Tarikh_1$.
For each OlympicGames, **if that** OlympicGames had venue decided in **some** Year **and** was held in **some** LeapYear, **then that** Year **is less than or equal to that** LeapYear.	对于每一场奥运会， 如果那一场奥运会的地点决定于某一个年份同时举办与某一个闰年，那么那一个年份是小于或相于那一个闰年。	Bagi setiap SukanOlimpik, jika tempat bagi SukanOlimpik ditentukan pada sesuatu Tahun dan diadakan pada sesuatu TahunLompat, maka Tahun itu adalah kurang daripada atau sama dengan TahunLompat itu.
For each Person, **if that** Person has **some** Height **and** has **some** Width, **that that** Height **is greater than that** Width.	对于每一个人， 如果那一个人拥有某一个高度同时拥有某一个宽度， 那么那一个高度是大于那一个宽度。	Bagi setiap Orang, jika Orang itu mempunyai sesuatu Ketinggian dan sesuatu Kelebahan, maka Ketinggian itu adalah lebih besar daripada Kelebahan itu.
For each Person, **if that** Person has **some** first-GivenName **and** has **some** FamilyName **then that** first-GivenName **is not equal to that** FamilyName.	对于某一个人， 如果那一个人拥有某一个给予姓名同时拥有某一个家族姓名那么那一个给予姓名不是相于那一个家族姓名。	Bagi setiap Orang, jika Orang itu mempunyai sesuatu NamePertama dan mempunyai sesuatu NameKeluarga maka NamePertama adalah tidak sama dengan NamaKeluarga itu.

Value-comparison constraint between roles of a ring fact type:

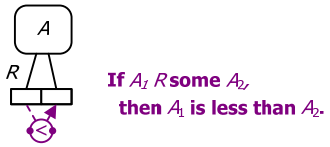


Figure 3.73 Value-comparison constraints on a ring fact type

Figure 3.73 shows the English verbalization pattern. The corresponding verbalizations in Bahasa Melayu and Mandarin respectively are:

Jika A_1 *R* sese∆/sesuatu A_2,
maka A_1 adalah kurang daripada A_2.

如果 A_1 某一∆RA_2,
那么 A_1 是少于 A_2。

Table 3.86 Example verbalizations of value comparison constraints on a ring fact type

English	BM	Mandarin
If EnglishWord$_1$ is a pre-synonym of **some** EnglishWord$_2$, **then** EnglishWord$_1$ **is less than** EnglishWord$_2$.	**Jika** PerkataanEnglish1 ialah pra-sinonim bagi PerkataanEnglish2, **maka** PerkataanEnglish1 **adalah kurang daripada** PerkataanEnglish2.	**如果**英文单词 1 是英文单词 2 的预同义词，**那么**英文单词 1 **是少于**英文单词 2。

All Other Cases:

For all other cases, at least one predicate has no reading from the join object type and/or at least one predicate reading has front text or at least one predicate is *n*-ary (*n* > 2). In this case, the general pattern for binaries is "For each *A*" followed by an if-then clause in the direction of the arrow, where each condition is phrased using the usual pattern for inverse readings and/or front text. Figure 3.74 shows the English verbalization pattern for a representative case.

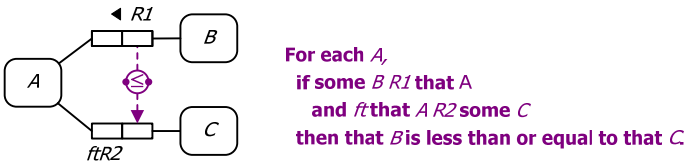


Figure 3.74 Value-comparison constraint for a complex case

For this case, the corresponding verbalizations in Bahasa Melayu and Mandarin respectively are:

Bagi setiap *A*,
jika sese∆/sesuatu *B* R_1 *A* itu
dan *ft A* itu R2 sese∆/sesuatu *C*,
maka *B* itu adalah kurang daripada atau sama denga *C* itu.

对于某一∆ *A*，
如果某一∆*B* R_1 那一∆*C*
和 *ft* 那一∆ *A* R_1 某一∆ *C*，
那么那一∆*B* 是少于那一∆ *C*。

Table 3.87 Example verbalizations of value comparison constraints: other cases

English	BM	Mandarin
For each Employee, **if** $Date_1$ is the birthdate of **that** Employee **and** after being interviewed **that** Employee was hired on $Date_2$,	**Bagi setiap** Pekerja, **jika** Tarikh1 ialah tarikhkelahiran bagi Pekerja **itu** **dan** setelah ditemu duga Pekerja **itu** diupah pada $Tarikh_2$,	**对于某一**位职员， **如果**日期$_1$是**那一**位职员的出生日期 **和**被受访后**那一**位职员被

then Date$_1$ **is less than** Date$_2$.	**maka** Tarikh$_1$ **adalah kurang daripada** Tarikh$_2$.	雇用于日期$_2$， **那么**日期$_1$**是少于**日期$_2$。
For each Student and Course, **if that** Student enrolled in **that** Course on **some** Date$_1$ **and** passed **that** Course on **some** Date$_2$ **then** Date$_1$ **is less than or equal to** Date$_2$.	**Bagi setiap pasangan dengan** Pelajar **dan** Kursus, **jika** Pelajar **itu** mendaftar dalam Kursus **itu** pada **sesuatu** Tarikh$_1$ **dan** lulus Kursus **itu** pada **sesuatu** Tarikh$_2$ **maka** Tarikh1 **adalah kurang daripada atau sama dengan** Tarikh$_2$.	**对于某一个**学生**和**课程**的组合，** **如果那一位**学生报名**那一门**课程于**某一个**日期$_1$ **和**通过**那一门**课程于**某一个**日期$_2$ **那么**日期$_1$**是少于或相同于**日期$_2$。

3.1.17 Cardinality Constraints

An *object cardinality constraint* restricts the cardinality (number of members) of the population of an object type for each state of the database. A *role cardinality constraint* restricts the cardinality of the population a specified role for each state of the database. Simple cardinality constraints are depicted graphically by placing "# θ *n*" next to the constrained object type or role, where θ is one of three numeric comparators (≤, =, ≥) and *n* is a positive integer (1, 2, 3, …). Complex cardinality constraints specify a set of cardinalities or a min..max range of cardinalities.

Table 3.88 Verbalizations of comparison operators in cardinality constraints

θ *n*	*English Verbalization*	*BM Verbalization*	*Mandarin Verbalization*
≤ *n*	**at most** *n*	Paling banyak n	最多 n
= *n*	**exactly** *n*	Hanya n	仅有 n
≥ *n*	**at least** *n*	Sekurang-kurangnya n	至少 n

3.1.17.1 Object Cardinality Constraints

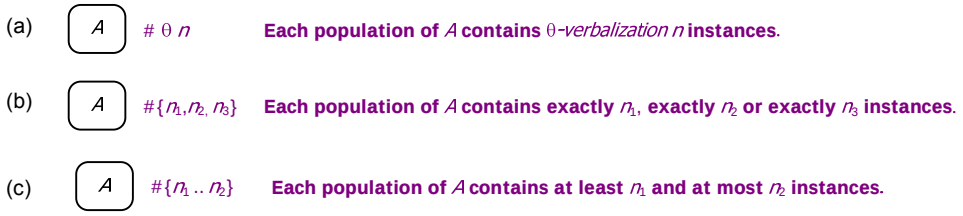


Figure 0.75 Object cardinality constraints

Figure 0.75 shows the English verbalization patterns. The corresponding verbalizations in Bahasa Melayu and Mandarin respectively are:

(a) Setiap populasi untuk *A* mengandungi θ *n*Δ.
每一个 *A* 的群体包含θ *n*Δ。

(b) Setiap populasi untuk *A* mengandungi hanya n_1, hanya n_2 atau hanya n_3 Δ.
每一个 A 的群体包含仅有 n_1, 仅有 n_2 或仅有 n_3 Δ。

(c) Setiap populasi untuk *A* mengandungi sekurang-kurangnya n_1 dan paling banyak n_2 Δ.

每一个 A 的群体包含至少 n_1 和最多 n_2 Δ。

Table 3.89 Example verbalizations of object cardinality constraints: case (a)

English	BM	Mandarin
Each population of Senator **contains at most** 100 **instances**.	**Setiap populasi untuk** Senator **mengandungi paling banyak** 100 **orang**.	每一个参议员的群体包含最多 1 0 0 位。
Each population of JuryMember **contains exactly** 12 **instances**.	**Setiap populasi untuk** AhliJuri **mengandungi hanya** 12 **orang**.	每一个评审团成员的群体包含 1 2 位。
Each population of ConsultingLawyer **contains at most** 1 **instance**.	**Setiap populasi untuk** PenguamRundingan **mengandungi paling banyak** 1 **orang**.	每一个咨询律师的群体包含最多 1 位。
It is necessary that each population of Senator **contains at most** 100 **instances**.	**Ia adalah perlu bahawa setiap populasi untuk** Senator **mengandungi paling banyak** 100 **orang**.	每一个参议员的群体包含最多 1 0 0 位是必要的。
It is obligatory that each population of ConsultingLawyer **contains at most** 1 **instance**.	**Ia adalah wajib bawa setiap populasi untuk** PenguamRundingan **mengandungi paling banyak** 1 **orang**.	每一个咨询律师的群体包含最多 1 位是强制性的。

Table 3.90 Example verbalizations of object cardinality constraints: case (b)

English	BM	Mandarin
Each population of CoRuler **contains exactly** 0 **or exactly** 2 **instances**.	**Setiap populasi untuk** RakanPemerintah **mengandungi hanya** 0 **atau hanya** 2 **orang**.	每一个联合统治者的群体包含仅有 0 或仅有 2 位。

Table 3.91 Example verbalizations of object cardinality constraints: case (c)

English	BM	Mandarin
Each population of Martian **contains at most** 3, **exactly** 7, **or at least** 9 **and at most** 11 **instances**.	**Setiap populasi untuk** PenghuniMarikh **mengandungi paling banyak** 3, **hanya** 7, **atau sekurang-kurangnya** 9 **dan paling banyak** 11 **orang**.	每一个火星人的群体包含最多 3，仅有 7，或至少 9 和最多 1 1 位。

3.1.17.2 Role Cardinality Constraints

Role cardinality constraints are restricted to unary fact types. For a simple cardinality, Figure 3.76 shows the English verbalization patterns (*ft* is front text). The verbalization of θ *n* is given in Table 3.88. Modalities are catered for in the usual way.

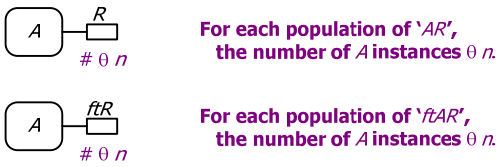


Figure 3.76 Role cardinality constraints: simple case

The corresponding verbalizations in Bahasa Melayu and Mandarin respectively are:

Bagi setiap populasi untuk '*A R*',
jumlah untuk *A* adalah θ *n*Δ.

Bagi setiap populasi untuk '*ft A R*',
jumlah untuk *A* adalah θ *n*Δ.

每一个'*A R*',的群体,
***A* 的数目是**θ *n* Δ。

每一个'*ft A R*',的群体,
***A* 的数目是**θ *n* Δ。

Table 3.92 Example verbalizations of role cardinality constraints

English	BM	Mandarin
For each population of 'Employee is chief executive officer', **the number of** Employee **instances is at most** 1.	**Bagi setiap populasi untuk** 'Pekerja ialah ketua pegawai eksekutif', **Jumlah untuk** Pekerja **adalah paling banyak** 1 **orang**.	**对于每一个**'职员是首席执行官' **的群体,** 职员**的数目是最多** 1 **位**。
it is obligatory that for each population of 'occasionally	**Ia adalah wajib bahawa bagi setiap populasi untuk** kadang-kadang 'Pekerja	**对于每一个**偶然的'职员是审核官' **的群体,**

Employee is a reviewer', **the number of** Employee **instances is at most** 3.	ialah penyemak', **Jumlah untuk** Pekerja **adalah paling banyak** 3 **orang**.	职员**的数目是最多** 3 位**是强制性的**。
For each population of 'Politician is co-ruler', **the number of** Politician **instances is** 1**or** 2.	**Bagi setiap populasi untuk** 'AhliPolitik ialah RakanPemerintah', **Jumlah untuk** AhliPolitik **adalah** 1 **atau** 2 **orang**.	**对于每一个**'政治家是联合统治者'**的群体，**政治家**的数目是** 1 **或** 2 位。
For each population of **'**Consultant is hired', **the number** Consultant **instances is at least** 2 **and at most** 5.	**Bagi setiap populasi untuk** ' perunding diupah', **Jumlah untuk** Perunding **adalah sekurang-kurangnya** 2 **dan paling banyak** 5 **orang**.	**对于每一个**'顾问受聘'**的群体，**顾问**的数目是至少** 2 **和最多** 5 位。
For each population of **'**Martian landed on our planet', **the number** Martian **instances is at most** 3, **exactly** 7, **or at least** 9 **and at most** 11.	**Bagi setiap populasi untuk** ' PenghuniMarikh mendarat di planet kita', **Jumlah untuk** PenghuniMarikh **adalah paling banyak** 3, **hanya** 7, **atau sekurang-kurangnya** 9 **dan paling banyak** 11 **orang**.	**对于每一个**'火星人登陆我们的星球'**的群体，**火星人**的数目是最多** 3，**仅有** 7，**或至少** 9 **和最多** 1 1 位。

3.2 Structure Analysis

3.2.1 Noun Classifiers

In this chapter, we compare the major differences between English, Bahasa Melayu, and Mandarin grammar that impact verbalization generation, including noun classifiers, singular/plural forms, pronouns (e.g. ‘who’and ‘that’) and indefinite articles (e.g. ‘a’ and ‘an’). In this section, we provide some examples of types of noun classifiers and their usage type, and demonstrate a graphical interface to implement Noun Classifiers in Mandarin and BM verbalization. We also further examine an ORM structure for noun classifiers to expose the structure of the possible relations between the connected properties.

Noun classifiers are called ‘PenjodohBilangan’ in BM, and ‘量词’ in Mandarin. Basically, a noun classifier is used before a word or name that indicates an amount or number. In English, noun classifiers are used only for special/specific conditions, such as uncountable quantities or a group of objects, unlike in BM and Mandarin where they are used widely. For Mandarin and BM, noun classifiers play a crucial role in making a phrase of word sound right. So we typically use noun classifiers in every Mandarin and BM noun phrase.

Table 0. illustrates a binary fact type with its simple mandatory role constraint verbalized in English, Bahasa Melayu and Mandarin.

Table 0.93 Examples of noun classifiers in mandatory role constraint verbalizations

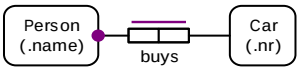
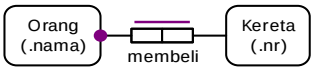
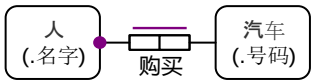
English Verbalization	BM Verbalization	Mandarin Verbalization
 **Each** Person buys **some** Car.	 **Setiap** Orang **membeli sekurang-kurangnya satu*buah*** Kereta.	 **每一*位*人**购买**某一*辆*汽**车。

Table 0. shows examples of how noun classifiers are used for various kinds of things in Bahasa Melayu.

Table 0.94 Examples of noun classifiers in Bahasa Melayu [34]

Noun Classifier	Usage	Examples
Batang	For long and thin things	rivers, candles, poles, pens, staples, teeth, sticks, chalk, bridges, needles
Bentuk	For small things that have curves	hooks, earrings, rings
Biji	For small fruits, vegetables or things	pumpkins, carrots, cabbages, eggplants, balls, cups, marbles, balloons, rice cakes, stones, eggs, glass
Bilah	For sharp things	machetes, knives, scissors, sickles
Buah	For large objects and solids, which have no specific shape	cars, land, houses, books, boxes, radios, towns, villages, pianos, computers, cameras
Buku	For objects that are lumpy	soap, bread, yarn
Ekor	For all kinds of animal	mosquitoes, ants, horses, alligators, foxes, elephants, rabbits, butterflies
Helai	For objects that are thin and wide	leaves, clothes, paper, cloth, towels, hair, grass
Orang	For humans	teachers, nurses, doctors, shopkeepers, retailers, postmen

Table 0. shows examples of how noun classifiers are used for various kinds of things in Mandarin.

Table 0.95 Examples of noun classifiers in Mandarin [33]

Noun Classifier	Usage	Examples
根、条、只、瓶、…	For long and thin things	pencils, trees, bottles, etc.
张、块、床、枚、…	For flat things	handkerchiefs, paper, linen, stamps, bedding, etc.
只、尾、条、…	For small animals	insects, dogs, cats,fish, birds, rabbits, etc.
本、册	For kinds of book objects	books, notebooks, photo albums, etc.
套、付、对、…	Sets of things	machines, utensils, instruments, etc.
辆、台、架、…	For transportation or electronic objects	Machinery, vehicles, etc.
个、位、条、…	For humans	teachers, nurses, doctors, shopkeepers, retail, postmen

As shown, in Mandarin the way of expressing the noun phrase is very complicated compared to BM and English. There are multiple classifiers to express one single noun phrase in Mandarin. The following table illustrates the difference between BM, Mandarin and English in this regard.

Table 0.96 Comparing noun classifiers for two single nouns in 3 languages

English Verbalization	BM Verbalization	Mandarin Verbalization
Each Fishmonger sells ***more than one*** Fish**.**	**Setiap *orang*** PenjualIkan menjual **lebih daripada satu *ekor*** Ikan.	**每一*位***鱼卖家出售**多过一*条***鱼。 (for specific animal) or **每一*位***鱼卖家出售**多过一*尾***鱼。 (for specific animal) or **每一*位***鱼卖家出售**多过一*只***鱼。 (for all kinds of animal)

3.2.1.1 Noun Classifiers in Bahasa Melayu

In BM, nouns are classified as either specific or general [43]. In ORM verbalization, the general kind of noun is mostly used, so we will look more deeply into it for better understanding to aid our later implementation. The general kind of noun is further classified into general concrete kind of noun and general abstract kind of noun.

A *general concrete noun* denotes a physical object that can be detected by human senses such as sight or touch. To be precise, it is usually used for a solid object that can't change its shape or behavior in action over time, unlike a general abstract kind of noun. Typical examples include orang (people), dinding (wall), kayu (wood), kucing (cat) and udara (air). Although air cannot be seen, it falls into the category of matter. Typically, a general concrete noun always requires a noun classifier, except in rare cases such as the noun 'udara'.

A *general abstract noun* denotes something that is usually non-physical, and hence cannot be detected by human senses, for example. pendapat (opinion), cadangan (suggestion), iklim (climate), keindahan (beauty), ajaran (teaching), and hasrat (desire). Abtract nouns also incude some things that can be visible, for example senyuman (smile), bayangan (shadow).

Typically, a general abstract noun does not requie a noun classifier. However, there are two exceptions. The noun classifier 'patah' is used for 'perkataan' (word), and the noun classifer 'buah' is used for for 'cerita' (story) , 'puisi' (poem), 'negara' (country), 'masyarakat' (society) , 'parti politik' (politic party), 'persatuan' (club) etc. For example: sepatah perkataan (one line of words), dua buah cerita (two stories), tiga buah persatuan (three clubs) [41].

Nouns that do not require noun classifiers:

Nouns that do not require noun classiers may be divided into the following six groups [41]:

i) Abstract nouns that show a feeling, characteristic, size, colour etc. Example words include hidup (living), hairan (surprise), karenah (behaviour), misteri (mystery), ketakutan (scare), cemburu (jealous), hari (day), umur

(age), masa (time), tarikh (date), harga (price), nilai (value), tempoh (period), naluri (instinct). For such nouns there is no need to include a noun classifier with the quantifier. For example: satu nama (one name), satu bank account nombor (one bank account number).

ii) **Measurement nouns (distance and capacity)**, such as dua kilometre (two kilometre), tiga kilogram (three kilogram).

iii) **Temporal nous**, such as dua pagi (two mornings), tiga malam (three nights).

iv) **Nouns that relate to God or astronomical objects** such as bulan (moon), matahari (sun) and bumi (earth). For example: satu bulan (one moon), satu bumi (one earth). However, the noun for star uses 'biji' as a noun classifier, and a planet other than Earth uses 'buah' as a noun classifier.

v) **Part of body which is singular**, such as satu mulut (one mouth), satu hidung (one nose).

vi) **The noun 'orang'.** This is because the word 'orang' can function as a noun classifier, such as dua orang (two people). That is why there is no such phrase as dua orang orang (two people people). In short, '**orang**' can be used as a noun phrase as well as a classifier in BM. In case the noun phrase/entity itself is **orang**, there is no need to place any classifier after the quantifier.

Examples of wrong ways using noun classifiers [43]:

1. **Each** Participant is required to comply with the rules set.

   (a) **Setiap** **orang** Peserta dikehendaki mematuhi peraturan yang telah ditetapkan. (×)

   (b) **Setiap** Peserta dikehendaki mematuhi peraturan yang telah ditetapkan. (✓)

   Explanation: Sometimes a noun classifier cannot be used along with a quantifier. (rule vi)

2. Mr. Ferry has provided a thoughtful Comment.

   (a) Encik Ikmal telah memberikan **sebuah**Pendapat yang bernas. (×)

   (b) Encik Ikmal telah memberikan **satu** Pendapat yang bernas. (✓)

   Explanation: Noun classifiers are not used to quantify abstract nouns. (rule i)

3. Ayu has picked **a** beautiful Flower and pinned on her hair.

(a) Ayu memetik **sejambak (for a bunch of flowers) bunga** lalu disematkan di rambutnya. (✗)

(b) Ayu memetik **sekuntum (for a single flower) bunga** lalu disematkan di rambutnya. (✓)

Explanation: A noun classifier is needed based on the suitability of the paragraph context.

3.2.1.2 Ideas for Implementing Noun Classifiers in Mandarin and BM Verbalization as an Extension to ORM

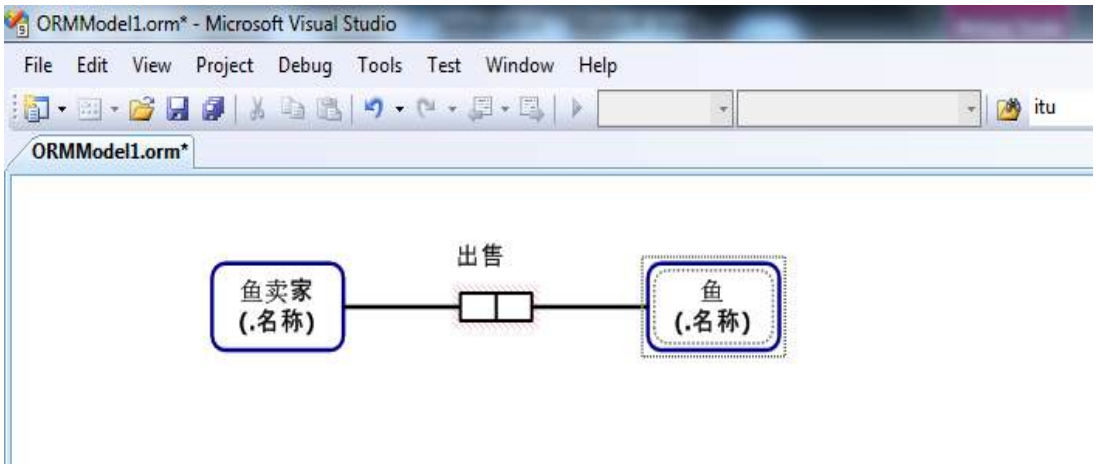


Figure 0.77 Binary fact type for Fishmonger sells Fish.

Figure 0. shows the Mandarin version of the fact type Fishmonger(.name) sells Fish (.name). Clicking the entity type Fish(.name) allows you to access its properties box. In the Fish entity type properties box shown in Figure 0. under the list name 'Noun Classifier', click  at the bottom right corner of the right column, and the selected box is displayed with a list of options to configure the noun classifier for the selected entity type as shown in Figure 0..

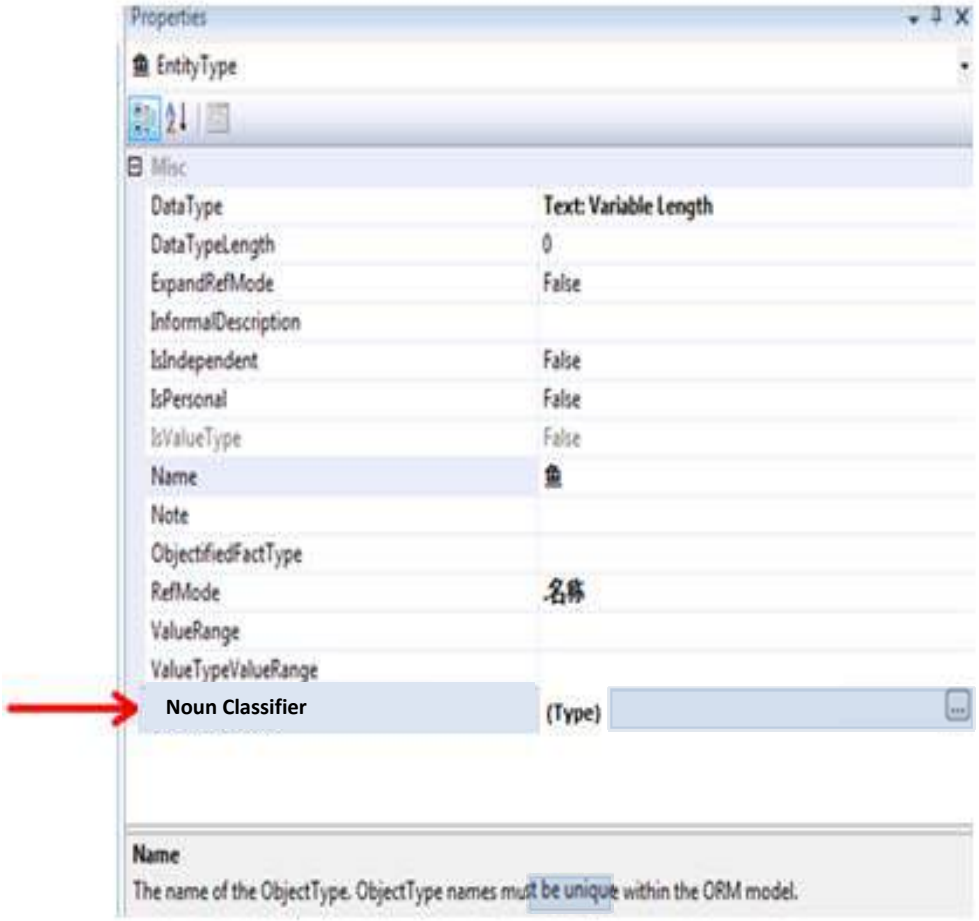


Figure 0.78 Properties Window for the Fish entity type

Figure 0.79 Noun Classifer Properties Window for the Fish entity type

The default noun phrase is based on the entity type that is typed in. As for type of noun phrase, a list of possible types of noun phrase is displayed (e.g. long and thin object, flat object, animal, book object, a group of objects, machine or vehicle). Based on what the user picks from this list box, a list of possibities for the 'Suggested Classifier' is shown based on the type/subtype of noun phrase chosen by the user.

Verbalization Result:

每一*位*鱼卖家出售多过一*尾*鱼。

3.2.1.3 ORM Metamodel for Use of Noun Classifiers

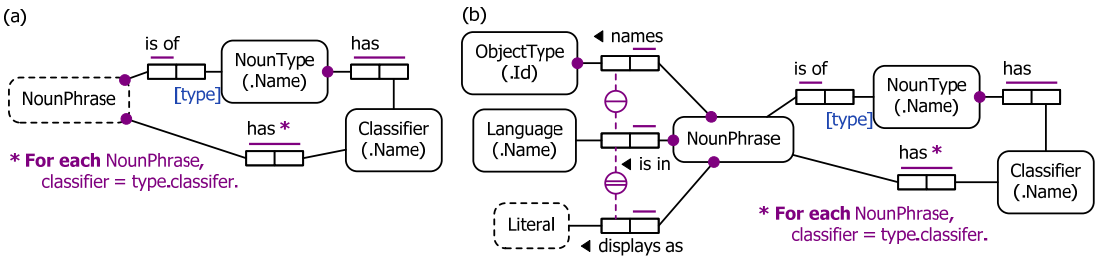


Figure 0.80 ORM metamodel fragment for use of noun classifiers

In a given ORM model, each object type has a distinct name. This name may be a simple noun but in general is a noun phrase (e.g. "Student", "Postgraduate Student", "Doctor or Dentist"). Using "NounType" for usage category, and "Classifier" for noun classifier, the situation for Mandarin may be modeled as shown in Figure 0.(a). The fact type NounType has Classifier may be prepopulated with known data.

In general however, the fact type NounPhrase is of NounType needs to be populated by the user. One reason for this to avoid massive databases (e.g. consider the number of noun phrases allowed for a give language). Another pragmatic reason is that the modelers often invent their own noun phrases to name various object types in their model, even though the terms they choose are not recognized in a standard language dictionary (e.g. "PostgradStudent", "PGstudent", or "Postgrad").

So the user interface for Bahasa Melayu or Mandarin needs to accept the noun type choice for a given object type from the user. Using the data in the prepopulated fact type for NounType has Classifier, the system can derive the allowed classifer(s), using the derivation rule shown in Figure 0.. For Bahasa Melayu, where the NounType has Classifier is *n*:1, that derived classifier will be the only possibility. For Mandarin however, it will often be the case that more than one classifier is derived, so the user needs to be presented with the list of possible classifiers from which to choose his/her preferred one. As we illustrate later, these requirements are met by our prototype tool.

The metamodel fragment in Figure 0.(b) provides one way to view the situation if, instead of using a separate model for each language, one wishes to use a single model with multiple display options based on the language choice.

3.2.2 'Who' and 'That'

In this section, we identify the differences between 'who' and 'that' (and their equivalent terms in BM and Mandarin) by discussing the different roles they play and specifying a mechanism to apply them in generating BM and Mandarin statements correctly. The words 'who' and 'that' (and their equivalent terms in other languages) may be used in many ways. For our purposes, they can act as relative pronouns, conjunctions or adjectives, based on how they are being used.

The choice of 'who' and 'that' is determined by whether they refer to people or other things. Formally, 'who' is reserved for people and 'that' is for other things. In Mandarin, 'who' and 'that' can often mean '谁' and '那一Δ', and as for BM, they can mean 'siapa'and 'itu'; but it won't always work this way when you physically connect them with clauses. In Mandarin, there are a few ways to consider in order to use them in a grammatically correct manner.

3.2.2.1 How to apply terms for 'who' and 'that' in Mandarin

Table 0.97 Methods for applying equivalent terms for 'who' and 'that' in Mandarin

Method	Usage Explanation
[1a] No connector is used for 'that' in a modal operator such as 'It is possible that' or 'It is necessary that'. [1b] Change the order of phrases when 'that' is used as a relative pronoun or when 'that' is used as a subordinating conjunction	[1a] In this case, we don't need to apply any kind of connector. We simply add on the clause as it is and it still works correctly. [1b] Change the order of some part of the sentence. Provide the details after 'that' in the 1st chunk, then add on the rest of the sentence in the next chunk. The intended meaning is still there even though the order of the statement is changed; 'that' here is '的'. E.g. It is possible that some Person visited more than one Country and that for some Country; more than one Person visited that Country. 某一个人到访多过一个国家和对于某一个国家，多过一个人到访那一个国家 [是可能的]. As shown, 'that' in the phrase 'and that' needs no connecter in Chinese, while the modal operator 'It is possible that' is moved to the end of the sentence. The 'that' in the phrase 'visited that country' is a demonstrative adjective to indicate which Country, so [3] is applied.

[2] Use '的'for 'who 'and 'that' used as relative pronouns	Make use of '的' to transform some portion of the sentence with 'who' and 'that' into a noun phase. '的' is applied to indicate a quality of a thing; usually it serves between an adjective and a noun (e.g. 美丽的花→beautiful flower). But it also can transform actions into qualities which in turn can act as an adjective in a sentence. (e.g. 每一个吸烟**的**人→ each person who smokes). As you can see, the phrase 'who smokes' has been qualified by using '的':"吸烟的" which can then easily plug into sentences like an adjective. Thus, it always comes behind the action that defines the noun, which is unlike English and BM. E.g. 每一辆被登记**的**车子 → Each car that is registered
[3] Use '**那一Δ**' for 'that' as a demonstrative adjective	'that' as a demonstrative adjective is used to identify which thing we want to talk about. In this case, 'that' usually means '那一Δ'. E.g. For each Country, at most one Region includes that Country →对于每一个国家,最多一个地区包含在**那一个**国家。

3.2.2.2 How to apply terms for 'who' and 'that' in BM

Table 0.98 Methods for applying equivalent terms for 'who' and 'that' in BM

Method	Usage Explanation
[1] Use '**yang**'→ 'that'and 'who' as relative pronouns	Make use of 'yang' to join the noun and the action clause together. E.g. Setiap orang **yang** merokok → each person who smokes Setiap kerete **yang** didaftarkan → each car that is registered
[2] Use '**bahawa**'→ 'that' as a conjunction	Make use of 'bahawa' used to join part of sentences.E.g. It is possible that some Person visited more than one Country and that for some Country; more than one Person visited that Country. **Iaadalahmungkin bahawa sese**Orang mengunjungi**lebihdaripadasatu**buahNegara dan bahawa**sese**buah Negara, **lebihdaripadasatu** Orang mengunjungi Negara **itu**. As you can see, 'that' in these phrases 'It is possible that' and 'and that' is used with 'yang' in BM, as for 'that' in the end phrase 'visited that country' is a demonstrative adjective that indicates which Country, so [3] is applied.
[3] Use '**itu**' for 'that' as a demonstrative adjective	'That' as a demonstrative adjective is used to identify which thing we want to talk about. In this case, 'that' usually means '**itu**'.E.g.

	For each Country, at most one Region includes that Country Bagi setiap negara, paling banyak satu buah daerah tergandung dalam Negara **itu**.

3.2.3 Other Grammar Standards

In this section, we discuss the key differences between singular and plural forms, and use of the indefinite articles 'a' & 'an' in English, Mandarin and Bahasa Melayu, as well as some minor grammatical features such as verb remuneration in BM. We also look into the different ways to apply the quantifiers 'exactly one', 'at most one' and 'more than one' in both active and passive sentences in Mandarin.

3.2.3.1 Singular/Plural Forms in English, BM and Mandarin

In both Mandarin and BM, most nouns and verbs have the same form regardless of whether they refer to one or multiple entities. The key point is that a 'number' is inferred from the context and is always placed ahead of a noun to distinguish whether a noun is singular or plural (e.g.一辆车 [satu buah kereta], 两辆车 [duabuah kereta]).

As you can see, '是' in Chinese and 'ialah' in BM are used to indicate both 'is' and 'are' in English. The same goes for 'has' and 'have' in English—Chinese uses '有' to represent either one of them, while BM uses 'ada'. Chinese grammar tends to emphasize the right usage of a classifier to define the type of noun; thus a noun classifier will always be used to classify the noun when applying a Mandarin equivalent to English articles such as 'a' and 'an'. Sometimes it is optional to omit the article for 'a' or 'an' in Chinese (e.g. 'she is **a** girl' which can mean '她是个女生' as well as '她是女生').

In English verbalization, there are many ways to express the plural form of nouns, as shown in item [3] below, in order to avoid changes made to the object

type's name. For example, '**is an instance of'** is used before a type term to indicate a member of a plural population, 'is an instance of Car' is used instead of saying 'one or many cars'.

Some key differences between between English, Mandarin and BM in verbalizing singular and plural are now summarized:

[1] Most nouns and verbs follow certain rules in how their plurals are formed. In particular, most English nouns are made plural by adding an "s" at the end:

- "car" (**汽**车)[kereta] ⇒ "cars"(**汽**车) [kereta]
- "subject"(**科目**)[subjek] ⇒ "subjects"(**科目**) [subjek]

[2] Nouns that end in certain letters (-s, -sh, -ch, -x, or -z) are made plural by adding "es" at the end:

- "gas" (气体)[gas]⇒ "gases" (气体)[gas]
- "class" (课堂)[kelas]⇒ "classes" (课堂)[kelas]

[3] A few nouns have plurals formed in other ways; several common examples of these nouns are provided below.

- "person"(人)[orang]⇒ "people" (人)[orang]
- "mouse"(老鼠)[tikus]⇒ "mice" (老鼠)[tikus]

[4] English verbs in the present tense follow the opposite pattern from nouns, as verbs typically have an "s" or "es" ending in the singular form but not in the plural form:

- "runs" (跑)[lari]⇒ "run" (跑)[lari]
- "washes"(洗)[cuci]⇒ "wash" (洗)[cuci]

[5] The verb "to be" and a few other common verbs form plurals irregularly:

- "is" (是)[ialah]⇒ "are" (是)[ialah]
- "am" (是)[ialah]⇒ "are" (是)[ialah]
- "was" (是)[ialah]⇒ "were" (是)[ialah]
- "has"(有)[ada]⇒ "have" (有)[ada]
- "does" (是/有)[ialah/ada]⇒ "do"(是/有)[ialah/ada]

For example:

- Each **expert** **reviews** at most 5 **papers**.

  每一位专家审查最多 5 篇论文。

  Setiap **Pakar**menyemak paling banyak 5 helai **Kertas**.

- Each **paper**that**is** **reviewed** by at least 2 **experts**.

  每一篇论文被最少两位专家审查。

  Setiap **Kertas** **disemak** oleh sekurang-kurangya dua orang **Pakar**.

[6] With the exception of "was" and "were", most past and future tense verbs do not have separate singular and plural forms. Verbs should agree in number with their subject (the noun or pronoun that performs the action).

For example:

- One researcher conducts an experiment.

  (一名研究者进行一项实验.)

  [satu orang penyelidikmenjalani satu eksperimen.]

- Two researchers conduct an experiment.

  (两名研究者进行一项实验.)

  [dua orang penyelidikmenjalani satu eksperimen.]

3.2.3.2 Tips on Using 'a' and 'an' in English, Mandarin and BM

In English, use of the indefinite articles 'a' and 'an' often depends on the pronunciation of the associated word. Usually, the sound of a word's first letter determines which article to use. If the word starts with a vowel sound (a, e, i, o, u), then '*an' should typically be used*, whereas if the word begins with a consonant sound, '*a' will typically be used*. In contrast, Mandarin and BM are not influenced by the pronunciation of the following word. Hence; in order to convey the equivalent of the articles 'a' and 'an' in Mandarin and BM, we simply use '一'and 'satu' in every case. For example, 'an apple' is translated as '一粒苹果' and 'satu biji epal'.

3.2.3.3 Verb Remuneration in BM [Tips for user input]

In BM, verbs almost all the time are added with remuneration, good prefix, suffix, clamp or insertion.This feature doesn't appear in English or Mandarin.

Prefix:

a) **me(n) -**

The remuneration prefix "me (n) -" indicates that the word is a verb that brings the meaning to the root word of an active sentence, whether transitive or intransitive. The prefix "me (n) -" changes form according to the first phoneme in the root word which includes, mem-, men-, meng-, meny-, me- and menge-.

E.g.

(i) SeseOrangmendaftar sesebuah Kursus.

Some Person enrolled in some Course.

某一个人报名于某一个课程。

(ii) SeseOrangmembaca sesebuah Buku.

Some Person read some Book.

某一个人阅读某一本书。

b) ber -

The remuneration prefix "ber-" shows that the word is a verb that brings the meaning to the root of an active sentence (where the subject performs the action described by the verb), whether transitive or intransitive

E.g.

(i) Setiap Persegi ialah berbentuk segi empat tepat.

Each Square is rectangular in shape.

每一个方形的形状是长方形。

c) per- and memper-

The remuneration prefix "per-" and "memper-" just use the transitive verb. The remuneration "per-" is often used in a sentence instruction, while a normal sentence completes it to become "memper-". Its uses are as follows:

E.g.

(i) Bagi setiap Orang dan Kursus, Orang itu dalam Kursus itumemperoleh paling banyak satu Gred.

For each Person and Course, that Person in that Course obtained at most one Grade.

对于每一个 ‘人与课程’，那一个人在那一个课程里获得最多一个等级。

d) di- and diper-

The remuneration prefix "di-" is the successor of the remuneration "me (n) -" in passive sentences (where the subject is the receiver of the action desctribed by the verb). The difference is that the first phoneme root word will not change. Similarly, the prefix "me (n)", the prefix "memper-" passive sentences translate into a "diper-".

E.g.

(i) Setiap Buku disemak oleh paling banyak satu Orang.

Each Book is reviewed by at most one Person.

每一本书被最多一个人审查。

Clamp - Remuneration terminal is the use of prefixes and suffixes to a root word to form a verb:

a) me(n)-…-kan

b) me(n)-…-i

c) memper-…-kan

Clamp "memper-…-kan" brings a causative meaning that is used only for transitive verbs, but is associated with the remuneration "ber- ".

d) memper-…-i

Similarly with clamp 'memper-…-kan', clamp 'memper-…-I' brings the causative meaning that is use only for transitive verbs.

e) di-…-kan & di-…-i

e.g.

(i) Setiap Orang dilahirkan di paling banyak satu buah Negara

Each Person was born in at most one Country.

每一个人出生于最多一个国家。

(ii) Setiap Kereta dibaiki oleh sekurang-kurangnya satu orang Jurutera.

Each Car was repaired by some Engineer.

每一辆车被某一个机械师修理。

f) diper-…-kan and diper-..-I

3.2.3.4 Exactly one/ at most one/ more than one for Active vs Passive Sentences in Mandarin

In this section, we discuss options to verbalize the 'exactly one', 'at most one' and 'more than one' quantifiers in Mandarin in both active and passive sentences.

a) Each Person has exactly one name. (Active Sentence)

每一个人只有/仅有一个名字。

→Similarly for 'at most one' and 'more than one'

b) Each Department **is headed by exactly one** Person. (Passive Sentence)

每一个部门只有/仅有被一个人所领导。

c) Each Department **is headed by at most one** Person.

每一个部门**最多被一**个人所领导。

d) Each Department **is headed by more than one** Person.

每一个部门**被多过一**个人所领导。

As you can see, for passive sentences that involve 'at most one' and 'exactly one', the pattern to structure the use of these quantifiers is **<at most/exactly>by<one><verb>.** In contrast, for passive sentences that involve 'more than one', the pattern to structure use of this quantifier is **by<more than one><verb>.**

3.2.3.5 The Preposition 'of' in English and Mandarin

Example:

Person$_1$ is a parent of **Person$_2$** (English)

人 $_1$ 是人 $_2$ 的父母。 (Mandarin)

The predicate reading for 'is a parent of' in Mandarin is '是…的父母', which is composed of two parts. The first part is for 'is' (是) and the second part is for 'a parent of' (的父母) with a placeholder for the object term Person$_2$ (人 $_2$) between them. In Mandarin, the preposition 'of' (的) is always placed just before its object. Thus, 'parent **of**' is rendered in Mandarin as **的**父母'.

4 Results and Discussion - Implementation

In this chapter, we describe how we have implemented the proposals discussed previously. We first discuss the prototype that we developed to automatically verbalize ORM models in Bahasa Melyau and Mandarin, catering for noun classifier choices and various other features. For this we include a representative sample of the C# code that implements the core verbalization functions. Finally, we discuss our extensions to the verbalization framework provided by the NORMA tool for ORM by simulating examples of possible output verbalization via NORMA software for Bahasa Melayu and Mandarin as well as snippets for noun classifier plugin identification.

4.1 C # Windows Application Prototype System (GUI)

In this section, we demonstrate our prototype system to model ORM verbalization covering enhanced features such as a language picker and verbalization entry browser for Bahasa Melayu and Mandarin, noun classifier pickup window properties (general and advanced search for noun classifiers with NounType/Classifier Usage Description) and a speech narrator feature to vocalize the verbalization output as well as as our ORM Tools Little Helper (an automated guide for using the system).

4.1.1 How to Insert Noun Classifiers into Chinese Verbalization

To better appreciate how our prototype system works in practice, we have prepared several videos that may be viewed by accessing the relevant URLs. For the following demonstration, please view the ***Chinese verbalization v3.avi*** video clip. Figure 4.1 shows the opening screen of our prototype.

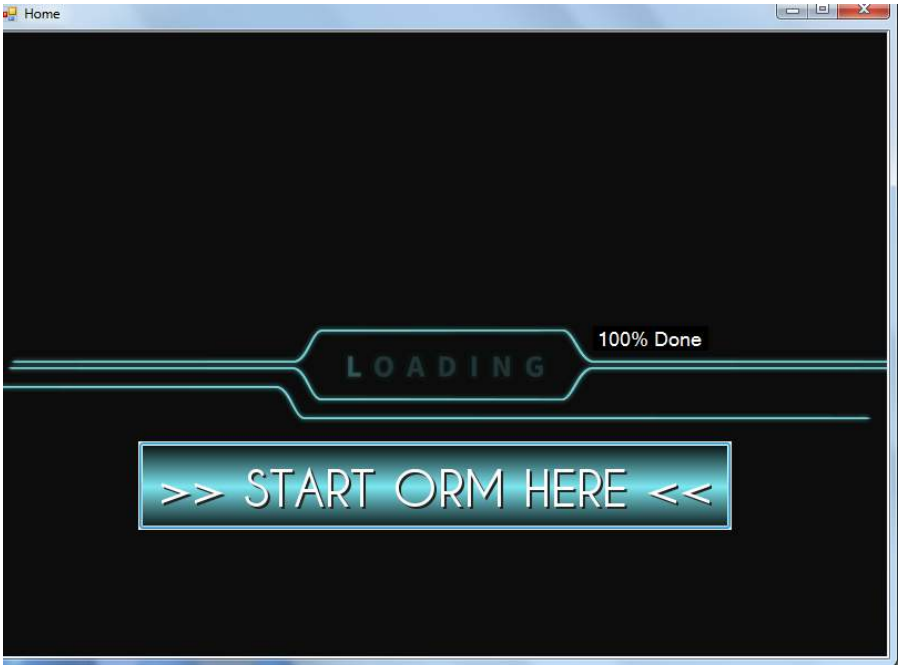


Figure 4.1 Screenshot of the opening page of our ORM prototype

Pressing the start button opens the dialog box shown in Figure 4.2.

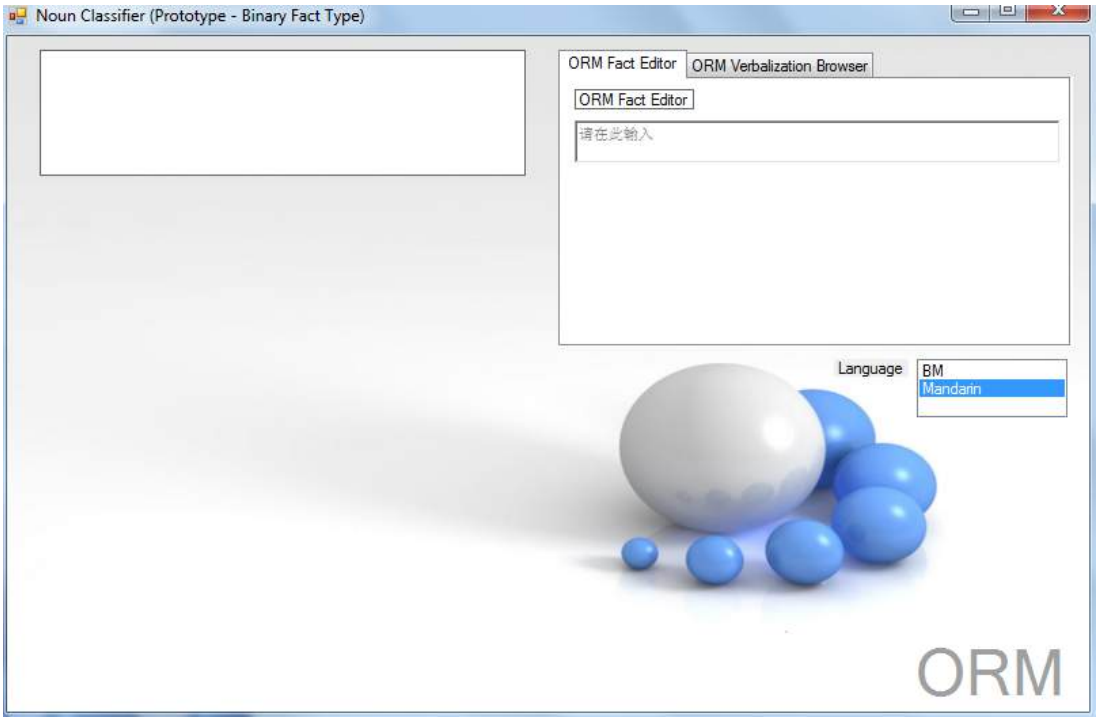


Figure 4.2 Screenshot of the first dialog box of our ORM prototype

The white box on the lefthand side is a drawing panel in which the model will appear later. To begin, we first select the relevant language (Bahasa Melayu or Mandarin) using the Language listbox shown. For this demonstration, we will use Mandarin. For this to work, the computer language pack should be set to Chinese (either traditional or simplified).

After choosing the language, we click the ORM Fact Editor tab and start to enter our ORM fact type in the textbox as shown in Figure 4.3. For this example, we enter a Mandarin reading for the fact type Country is the birthcountry of Politician. The fact type reading is entered in Pinyin (https://en.wikipedia.org/?title=Pinyin), a Romanized rendering of the Chinese phonetic system. The same sound may be used for many different characters, so a list of possible characters is displayed for the user to pick the one intended, as shown for 'shi' in Figure 4.3.

The object type names are entered in square brackets so the system can distinguish them from predicate readings. The system displays object type names in red colour, using blue colour for predicate text.

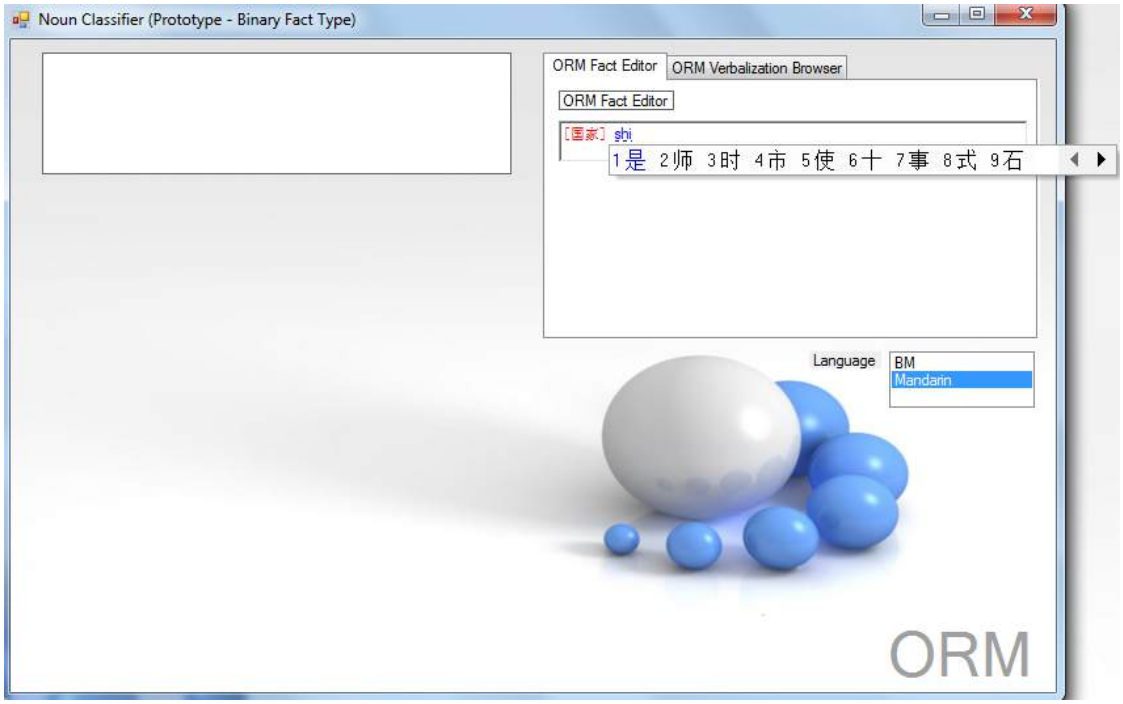


Figure 4.3 Entering a fact type reading in Mandarin in the ORM Fact Editor

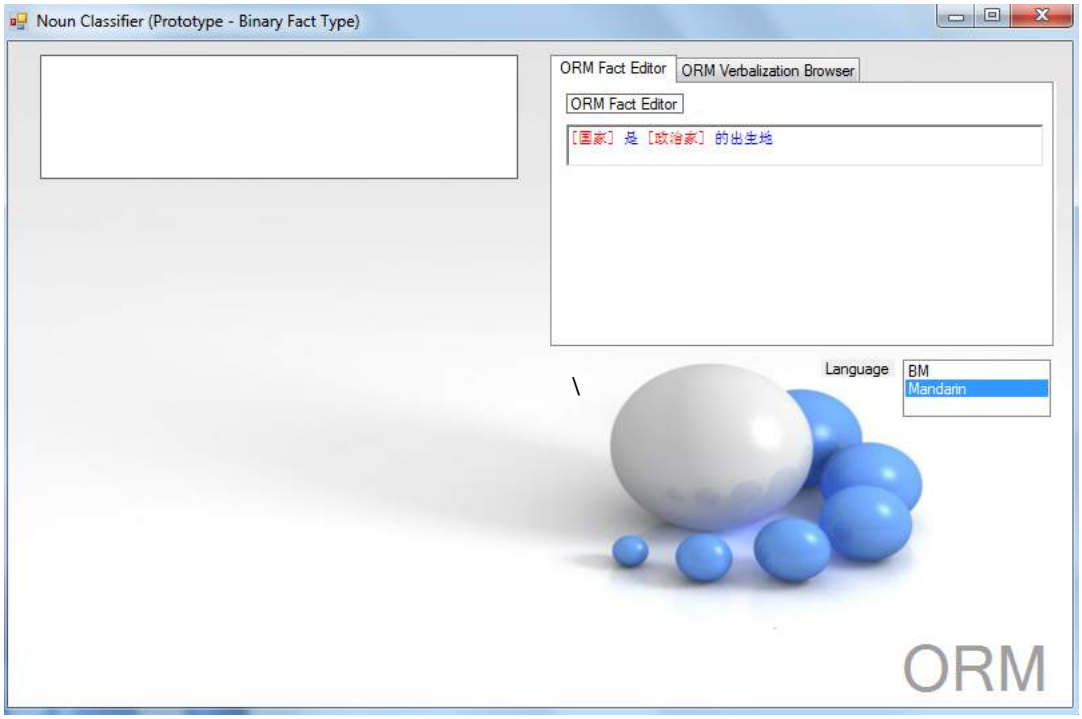


Figure 4.4 **A fact type reading has been entered in Mandarin in the ORM Fact Editor**

Figure 4.4 shows a screenshot after the whole fact type (国家是政治家的出生地。) has been entered. Pressing Ctrl+Enter now causes an ORM diagram to be automatically generated, as shown in Figure 4.5.

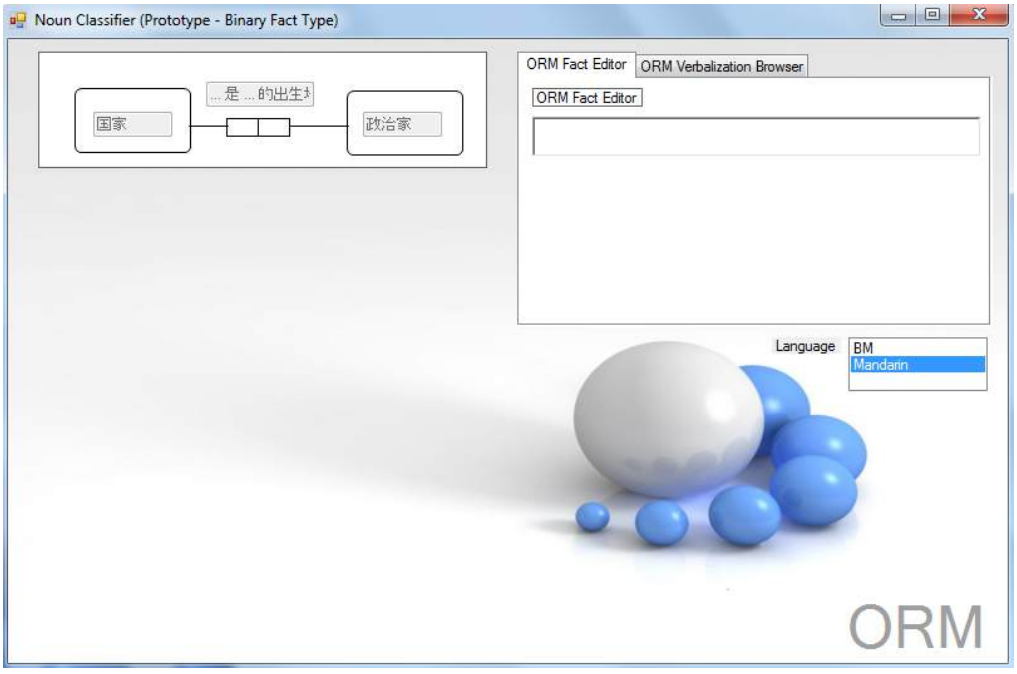


Figure 4.5 **An ORM diagram for the textually entered fact type is generated**

In the ORM diagram, the object type names (here, 国家 and 政治家) appear inside rounded rectangle shapes for the object types, while the predicate reading (here, …是…的出生地) appears above the predicate shape (here, two conjoined role boxes), with an ellipsis "…" used for each placeholder for an object type.

For each object type, we need to choose the relevant noun classifer. Double clicking an object type shape opens the Classifier Properties window, with the object type name (e.g. '政治家) displayed in the Noun field, as shown in Figure 4.6. A list of possible types of noun phrase is displayed in a list box. If the relevant type appears in the displayed list, we simply click it to indicate our choice. If the list is too long to fit in the displayed list, press the search button to find the relevant item.

In our example, '政治家' (Politcian) is best categorized as of type '动物–人类' (Living being – Human), so we choose '动物–人类'. Once we have selected the relevant type of noun phrase, a list of suggested noun classifers is automatically displayed in the Classifier box, as shown in Figure 4.6. We now click the relevant item from this list to make our choice of noun classifer. In this example, we choose the classifer '名'.

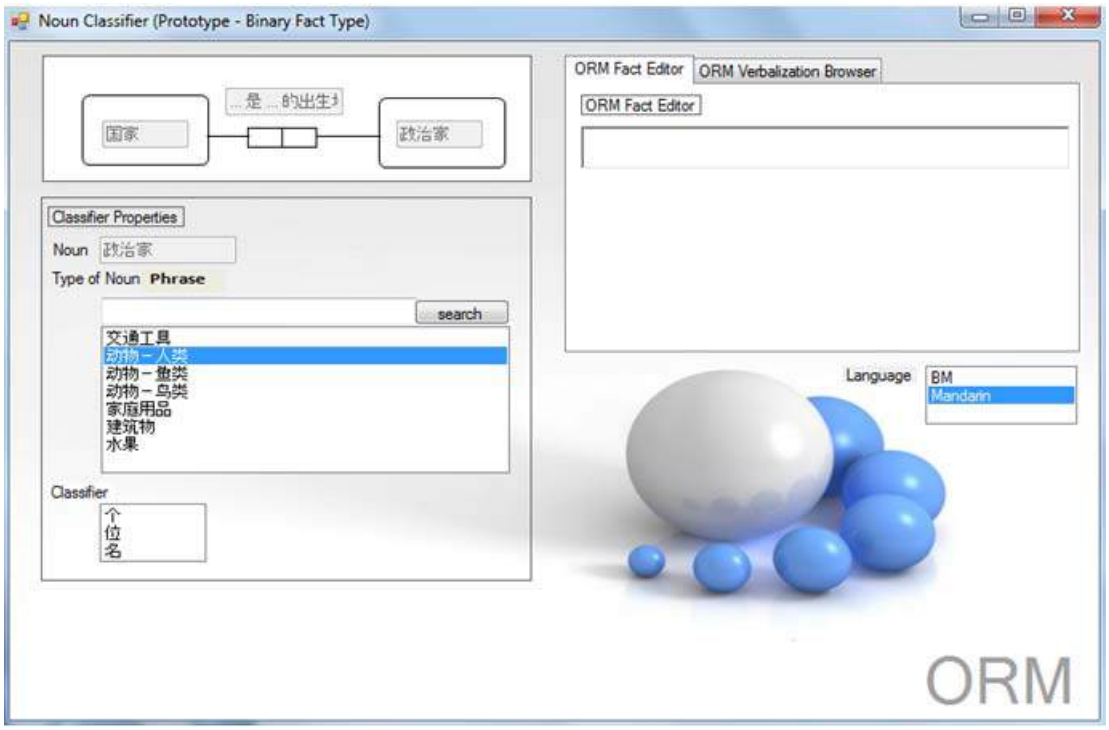


Figure 4.6 Choosing the relevant type of noun phrase

Having chosen the noun classifer '名' for '政治家'., we can see our choice displayed in a tooltip by hovering the mouse move over the '政治家' entity type box, as shown in Figure 4.7.

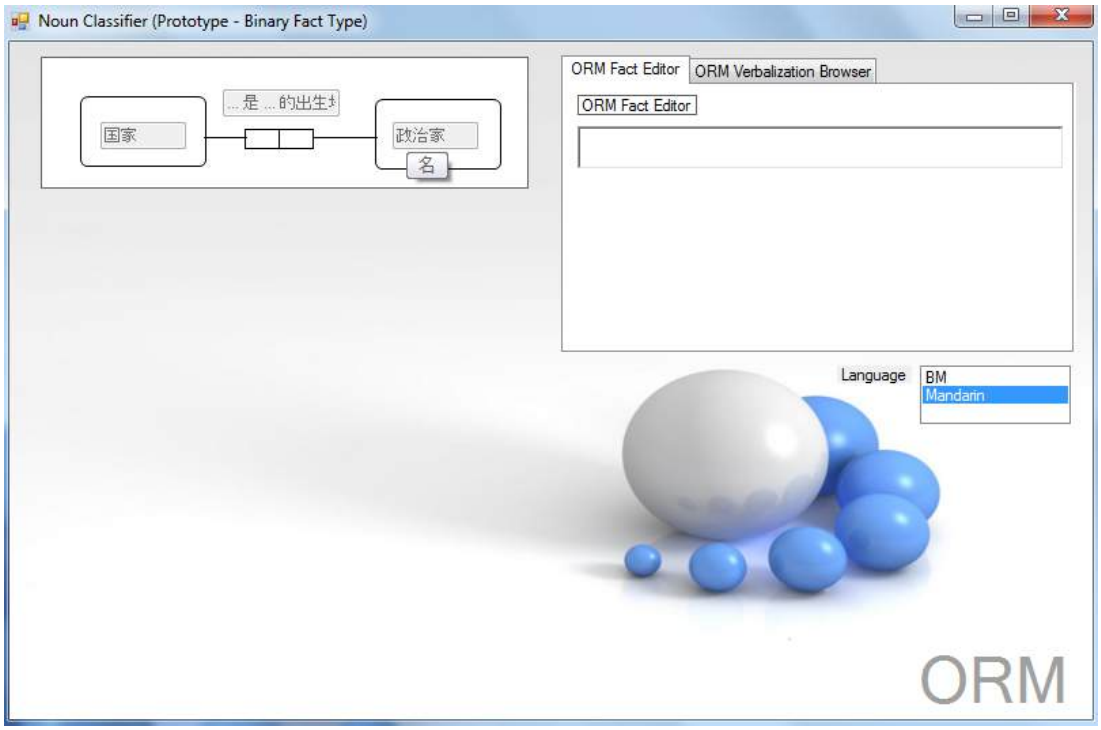


Figure 4.7 **The noun classier '名' chosen for the righthand object type is displayed**

The noun classifer for the lefthand entity type 国家 (Country) is slected in a similar way by double clicking the entity type and accessing the classifier properties content. In this case, we choose the classifier '个' for '国家' (see Figure 4.8).

Figure 4.8 **The noun classier '个' chosen for the righthand object type is displayed**

4.1.2 Internal uniqueness constraints on binary fact types

After choosing the noun classifiers for all the object types, we now right click the fact type roles to display a drop-down list of the possible internal uniqueness constraint pattens for the binary fact type. These are displayed as: *n:*1 (many-to-1); *m:n* (many-to-many); 1:1 (one-to-one) and 1:*n* (one-to-many). For our example fact type (**国家是政治家的出生地)**, one country may be the birth country of many politicians, so we choose the 1:*n* uniqueness constraint pattern, as shown in Figure 4.9.

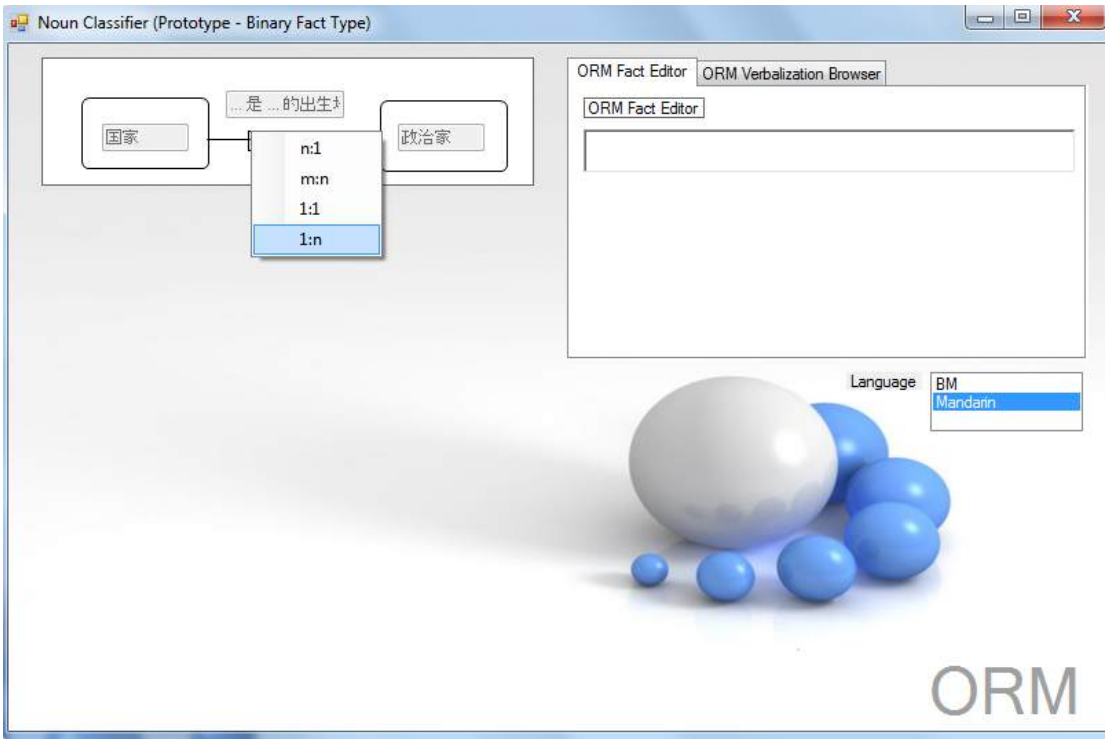


Figure 4.9 Choosing the uniqueness constraint pattern for a binary fact type

On the ORM diagram, a uniqueness constraint bar is now automatically added over the righthand role to indicate the 1:*n* constraint pattern. We now double click the roles, and click the tab ORM Verbalization Browser tab to view the automatically generated Chinese verbalization, which includes the relevant noun classifiers. By default, the positive verbalization is displayed, as shown in Figure 4.10. The following colouring scheme is used to distinguish different components of the

verbalization: **purple** (object type names); **green** (predicate readings); **blue** (logical words, e.g. quantifiers); **red** (noun classifiers—introduced for Chinese and BM).

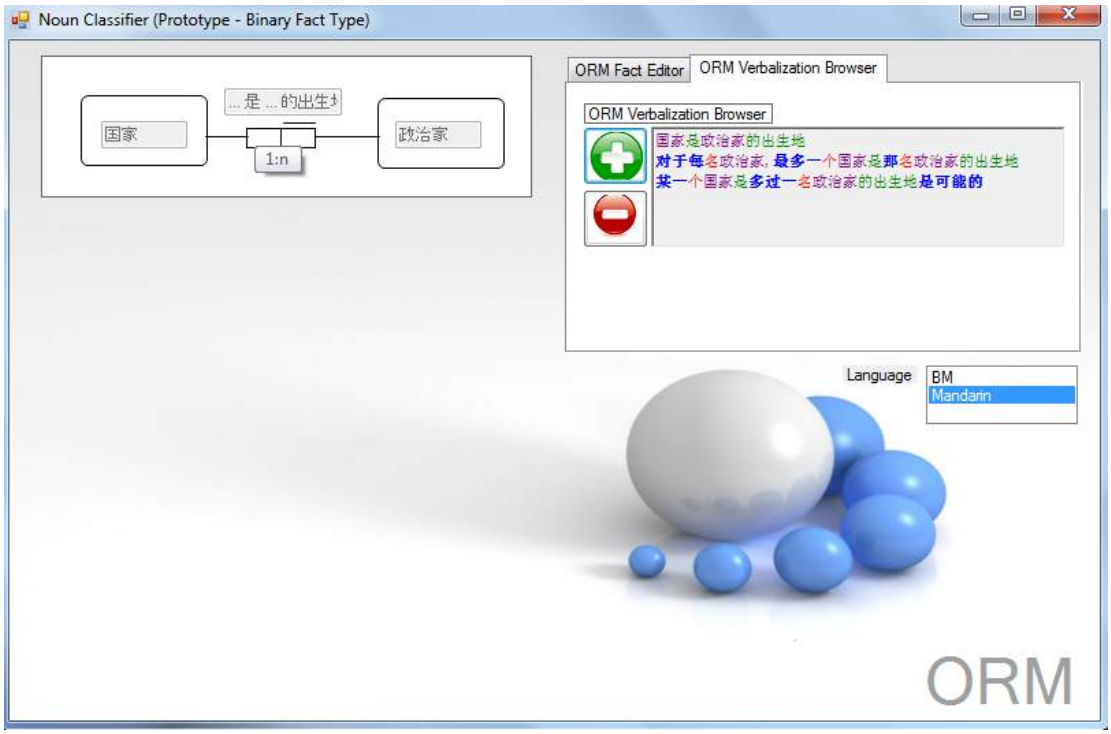


Figure 4.10 **Display and positive verbalization of an internal uniqueness constraint**

Table 4.1 shows the fact type verbalization as well as the positive verbalization of this uniqueness constraint (including the absence of a uniqueness constraint on the other role) in both Mandarin and English (our prototype does not generate the English verbalization).

Table 4.1 **Positive verbalization of a uniqueness constraint in Mandarin and English**

Mandarin	English
国家是政治家的出生地 对于每一名政治家，最多一个国家是那一名政治家的出生地 某一个国家是多过一名政治家的出生地是可能的	Country is the birthcountry of Politician For eachPolitician, at most one Country is the birthcountry of that Politician It is possible that some Country is the birthcountry of more than one Politician

To access the positive verbalization, double click the predicate (default setup) or just click the '+' (plus for positive) button. To access the negative verbalization of the constraint, double click the '–' (minus for negative) button, as shown in Figure 4.11.

Figure 4.11 Negative verbalization of an internal uniqueness constraint

Table 4.2 shows the fact type verbalization as well as the positive verbalization of this uniqueness constraint pattern in both Mandarin and English.

Table 4.2 Negative verbalization of a uniqueness constraint in Mandarin and English

Mandarin	English
国家是政治家的出生地 对于某一名政治家，多过一个国家是那一名政治家的出生地是不可能的 某一个国家是多过一名政治家的出生地是可能的	Country is the birthcountry of Politician It impossible that for some Politician, more than one Countryis the birthcountry of that Politician It is possible that some Country is the birthcountry of more than one Politician

The other internal uniqueness constraint patterns for binary fact types are handled in a similar fashion. Figure 4.12 illustrates the positive verbalization for the *n:*1 pattern on the Mandarin version of the fact type Politician was born in Country. Figure 4.13 illustrates the positive verbalization for the 1:1 pattern on the Mandarin version of the fact type President was born in Country.

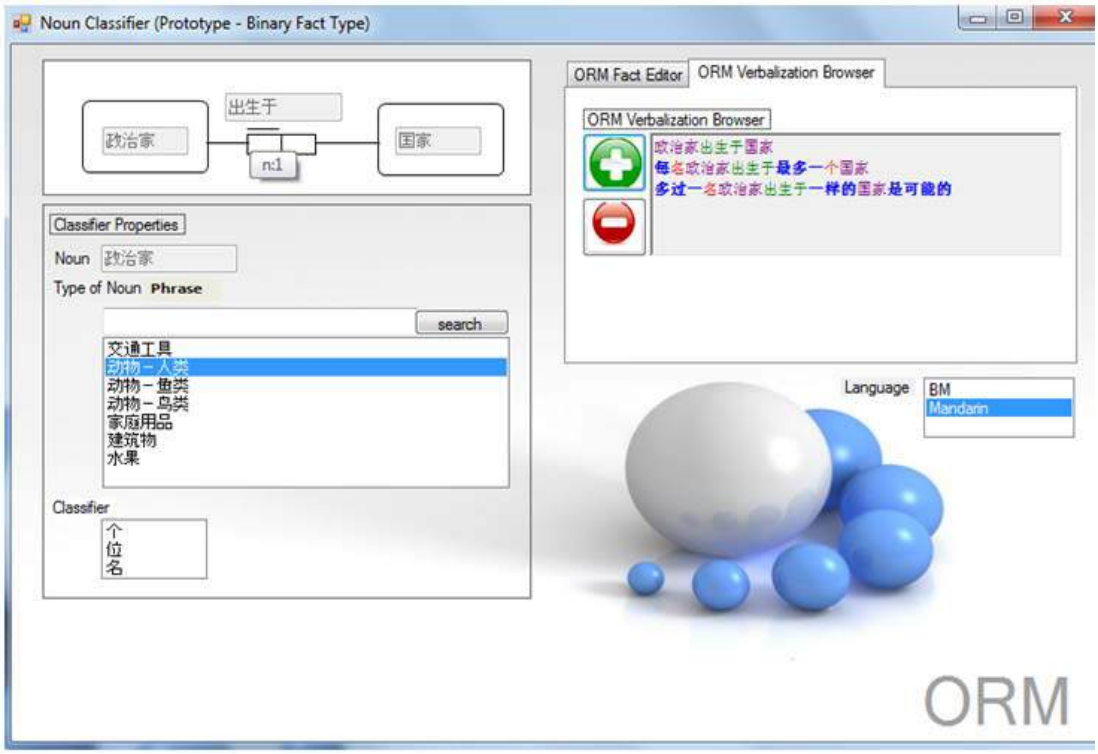


Figure 4.12 Positive verbalization of an *n*:1 uniqueness constraint pattern in Mandarin that imply ' Politician was born in Country'

Figure 4.13 Positive verbalization of a 1:1 uniqueness constraint pattern in Mandarin that imply 'President was born in Country'

4.1.3 Noun classifiers and uniqueness constraints on binary fact types in Bahasa Melayu

Noun classifiers and uniqueness constraints for Bahasa Melayu may be entered in a similar way to that described for Mandarin. Figure 4.14 shows a screenshot of our prototype after entering and constraining the BM version of the *n:*1 English fact type Politician was born in Country.

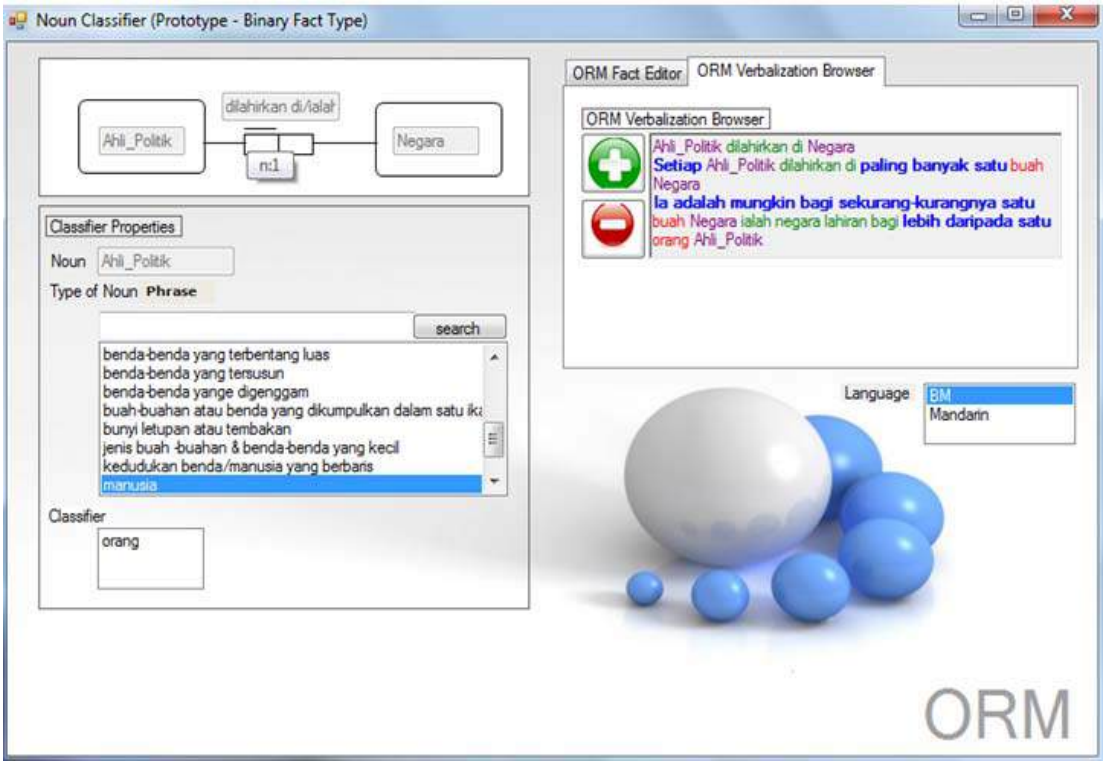


Figure 4.14 Positive verbalization of an *n:*1 uniqueness constraint pattern in BM for ' Politician was born in Country'

4.1.4 Simple mandatory role constraints

Figure 4.15 shows a screenshot of our prototype after entering and constraining the BM version of the English fact type Politiican was born in Country. The Mandarin version is shown for verbalizing the combination of the mandatory role and uniqueness constraint, corresponding to the English verbalization 'Each Politician was born in exactly one Country').

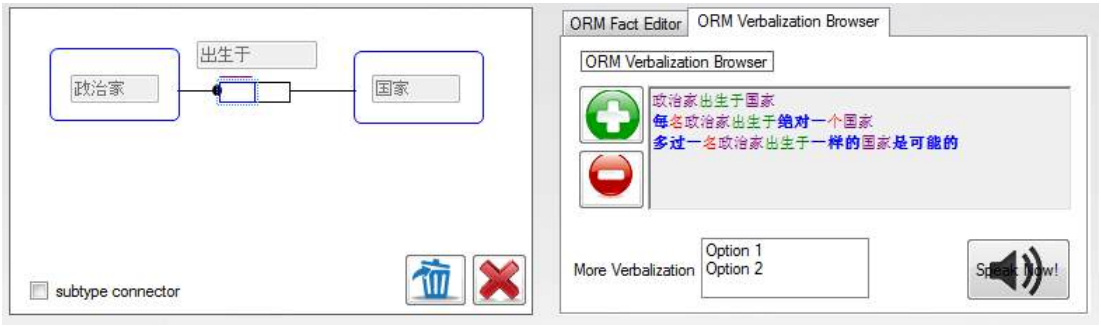


Figure 4.15 Positive verbalization of a combined uniqueness and mandatory role constraint in Mandarin for the fact type ' [Politician] was born in [Country]' verbalized as 'Each Politician was born in exactly one Country'

{ view languagepicker.wmv **&** n1relationship,ismandatory,NC,narratorspeaker.wmv **video cip for demo}**

Figure 4.16 Improved version of the ORM Fact Editor for entering fact types without spacing between each word in Mandarin (compare to figure 4.4).

Figure 4.17 Supporting ORM models where fact types share a common object type. Here, 'Country is bornplace for Politician' and 'Country has Name' share 'Country'.

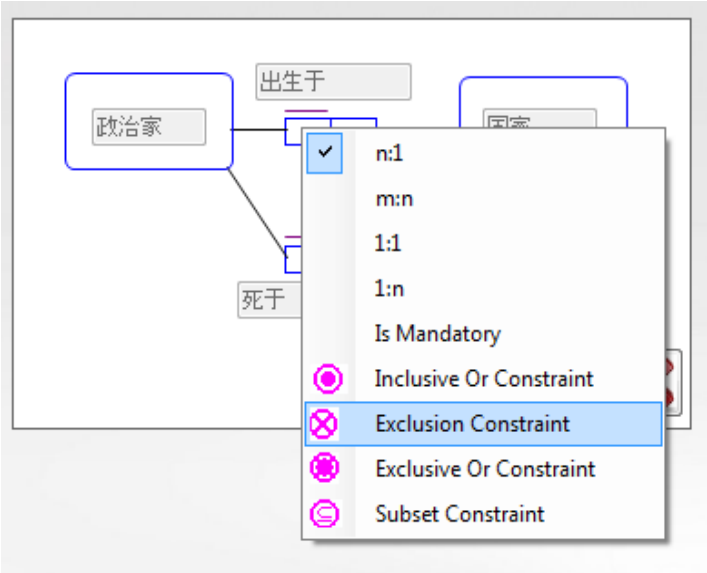


Figure 4.18 Extended version of constraint context menu for accessing exclusion, inclusive- or, exlusive-or and subset constraints.

Figure 4.19 Example of pair-exclusion constraint between 'Politician was born in Country' and 'Politician was died in Country'.

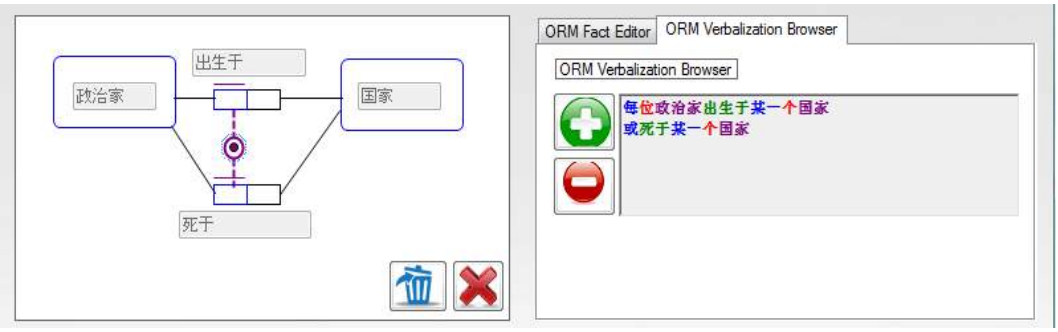


Figure 4.20 Example of inclusive-or constraint between Politician's roles in 'Politician was born in Country' and 'Politican was died in Country'

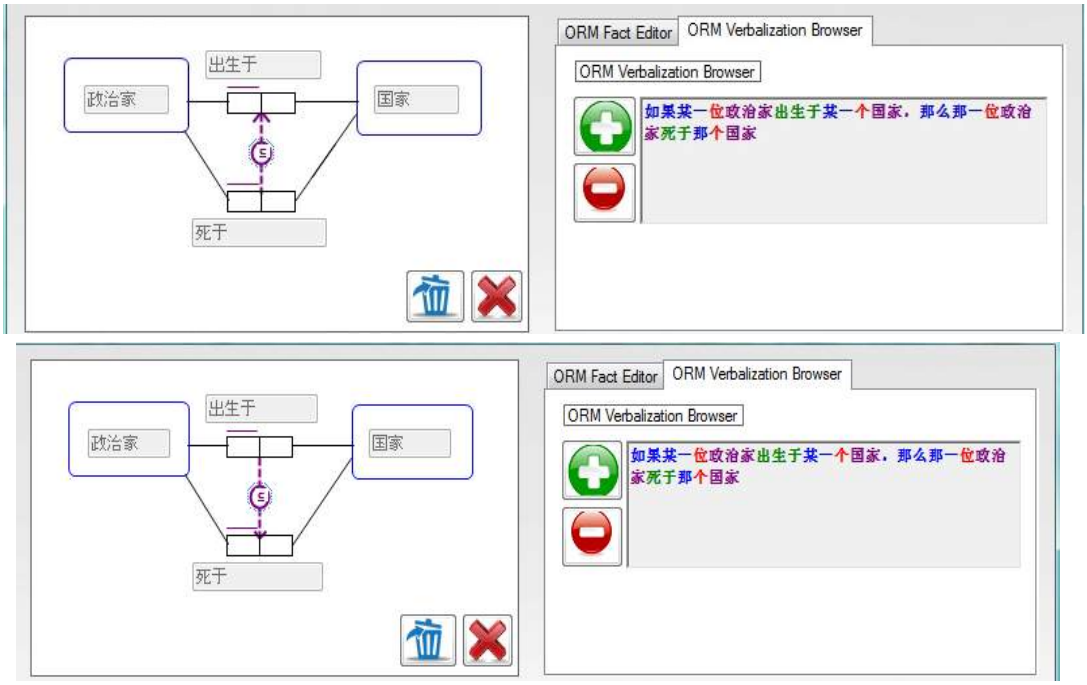


Figure 4.21 Examples of subset constraint support (both directions) between 'Politician was born in Country' and 'Politican died in Country'

Figure 4.22 illustrates one option for verbalizating the inclusive-or constraint '**Each** Student has joined **some** Club **or** obtains **some** Scholarship'.

Figure 4.22 Example of inclusive-or constraint verbalization (Option 1)

Figure 4.23 illustrates a second option for verbalizating the inclusive-or constraint '**For each** Student, **at least one of the following holds: that** Student has joined **some** Club; **that** Student obtains **some** Scholarship'**.**

Figure 4.23 Example of inclusive-or constraint verbalization (Option 2)

{view inclusiveor(default&addon verbalization).wmv video clip for demo}

Figure 4.24, Figure 4.25 and Figure 4.26 illustrate Mandarin support for selecting a frequency constraint of 3..5 for Student's role in the fact type Student is enrolled in Course.

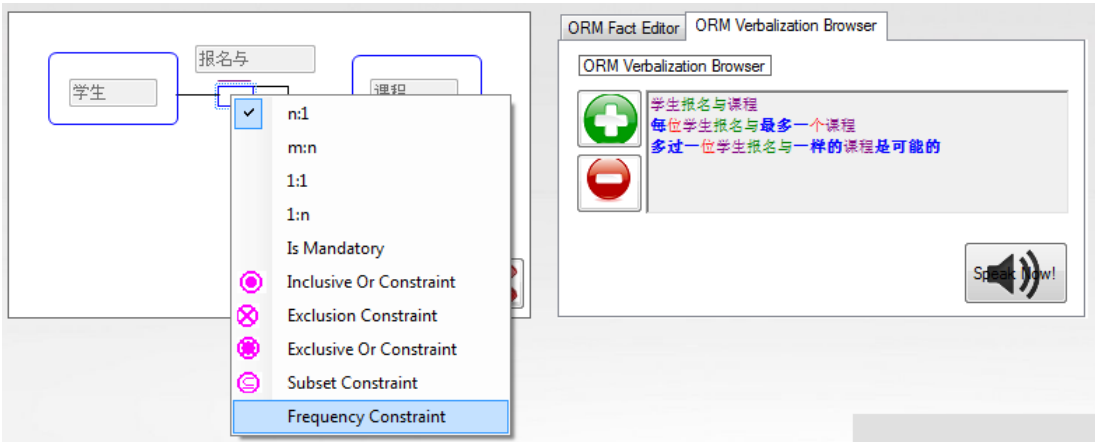


Figure 4.24 Choosing to add a frequency constraint

{view frequencyconstraint.wmv video clip for demo}

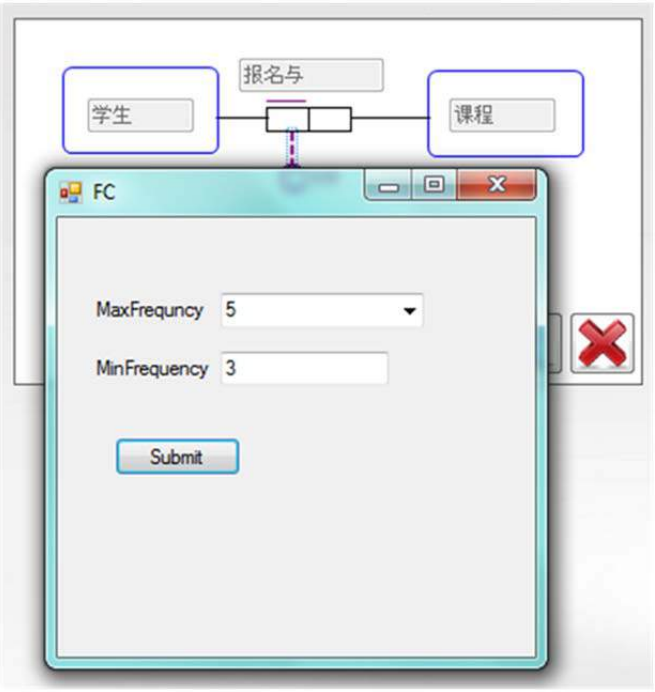


Figure 4.25 Setting the range of the frequency constraint to 3..5

Figure 4.26 Displaying and verbalizing the frequency constraint

Figure 4.27 and Figure 4.28 illustrate Mandarin support for selecting a frequency constraint of at least 2 for Student's role in the fact type Student is enrolled in Course.

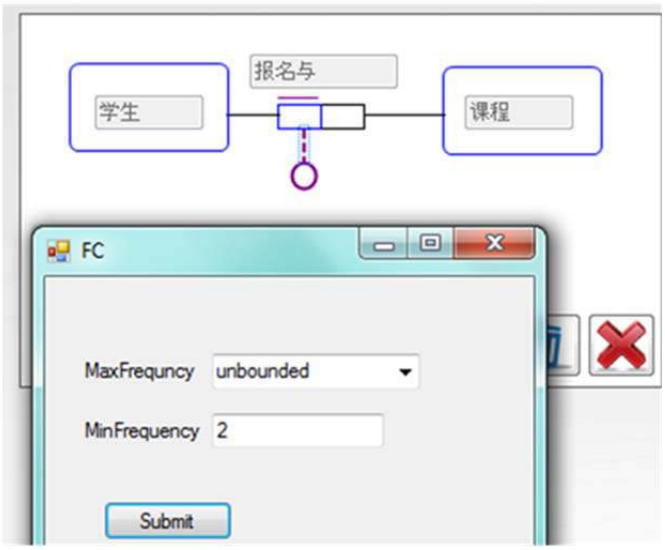


Figure 4.27 Setting a frequency constraint to be "at least 2"

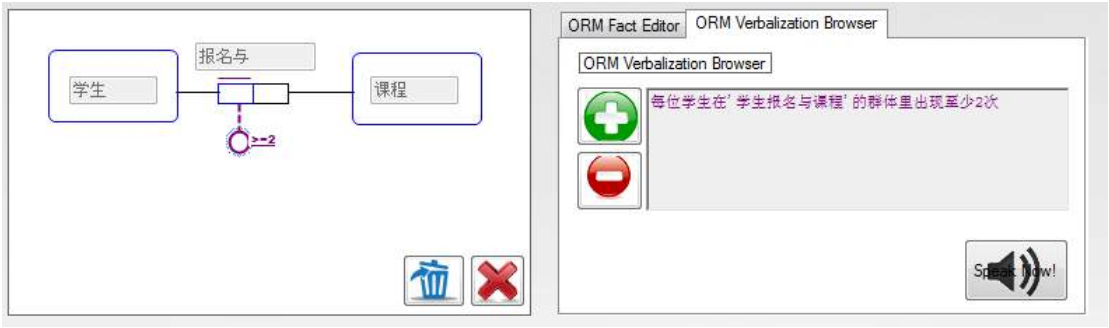


Figure 4.28 Displaying and verbalizing an "at least 2" frequency constraint

Figure 4.29 and Figure 4.30 illustrate Mandarin support for selecting a frequency constraint of at most 3 for Student's role in the fact type Student is enrolled in Course.

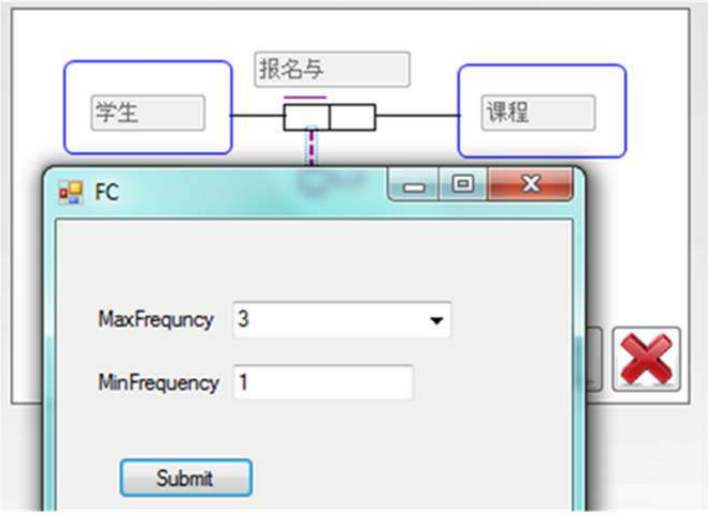


Figure 4.29 Setting a frequency constraint to be "at most 3"

Figure 4.30 Displaying and verbalizing an "at most 3" frequency constraint

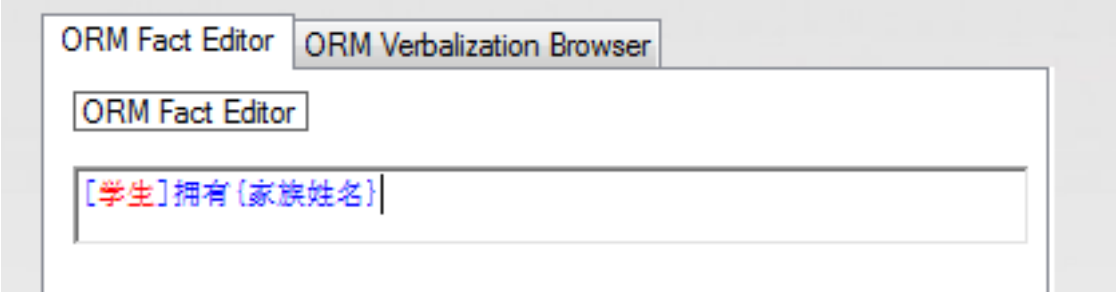


Figure 4.31 Entry of the fact type '[Student] has {FamilyName}', with the value type in braces.

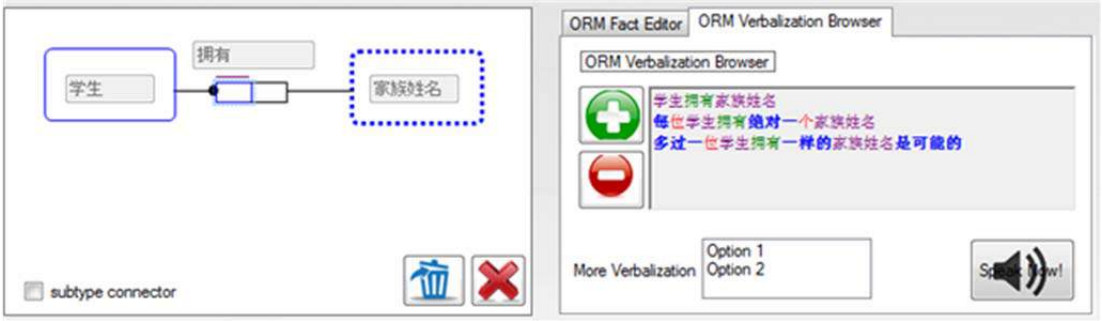


Figure 4.32 Dotted line display for the value type FamilyName, and verbalization of the constraint on the n:1 fact type 'Student has FamilyName' as 'Each Student has exactly one FamilyName'.

Figure 4.33 shows the fact types Student was born on Date; Student entered school on Date for which we will add a value-comparison constraint.

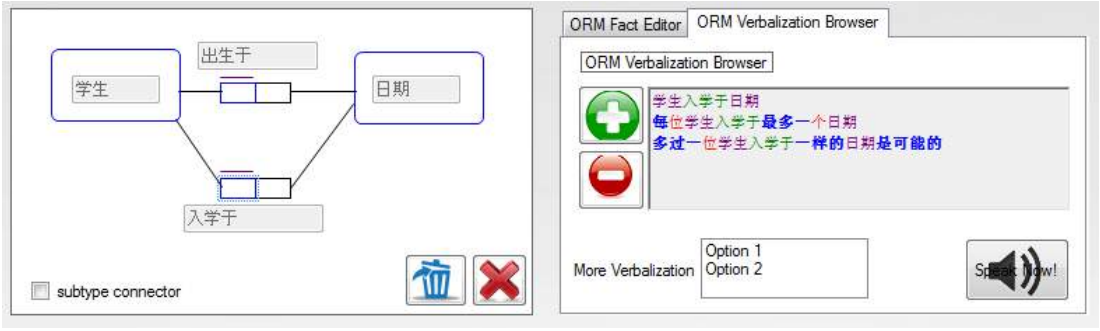


Figure 4.33 Entering fact types to constrain with a value-comparison constraint

{view valuecomparison.wmv video clip for demo

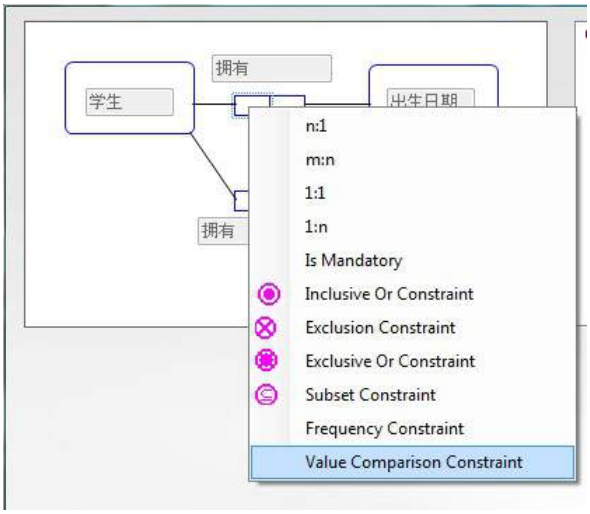


Figure 4.34 Choosing to add a value-comparison constraint

Figure 4.35 Initial display of the value-comparison constraint (without comparator)

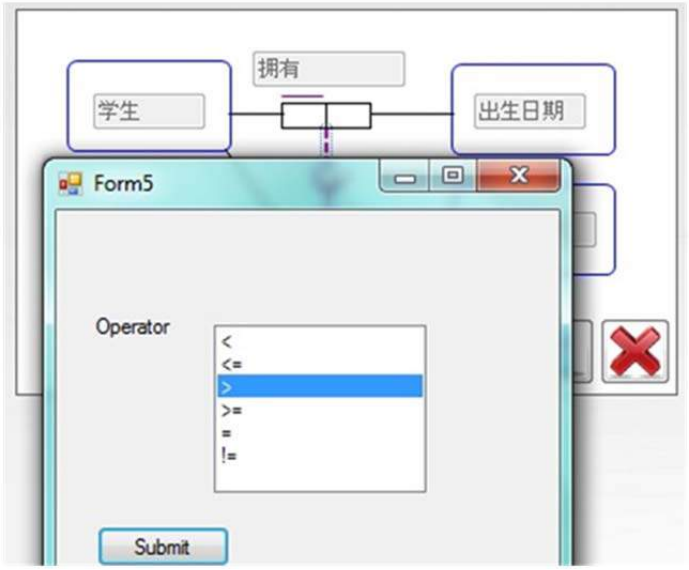


Figure 4.36 Selecting the value-comparison operator 'is greater than'

Figure 4.37 Displaying and verbalizing a value-comparison constraint

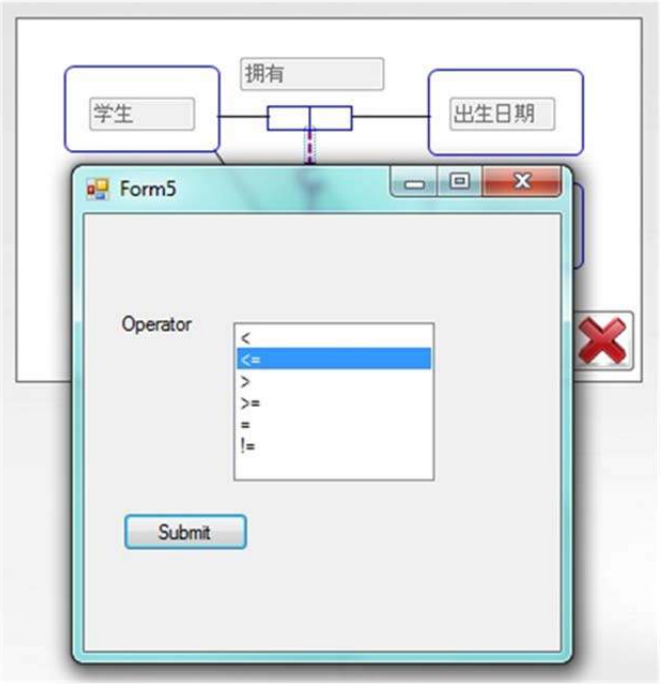


Figure 4.38 Selecting the value-comparison operator 'is less than or equal to'

Figure 4.39 Displaying and verbalizing an 'is less than or equal to' value-comparison constraint

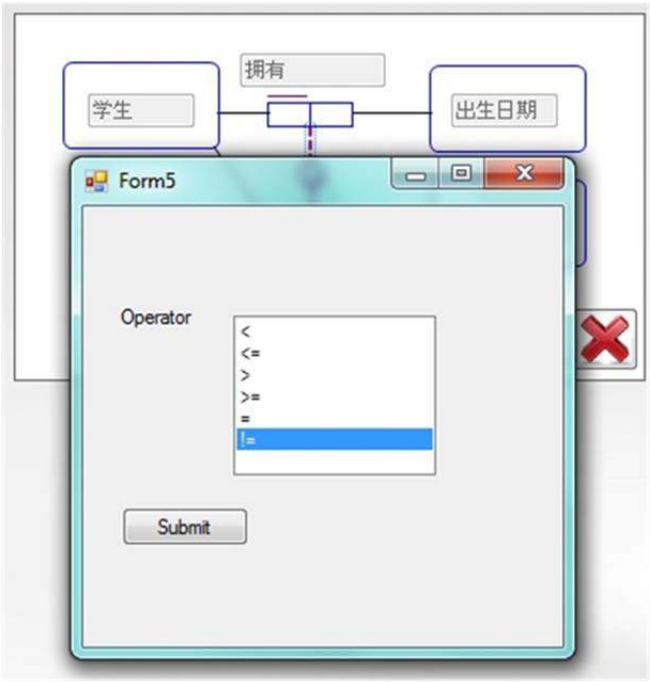


Figure 4.40 Selecting the value-comparison operator 'is not equal to'

Figure 4.41 Displaying and verbalizing an 'is not equal to' value-comparison constraint

Figure 4.42 illustrates entry in Mandarin of the homogeneous fact type Student likes Student.

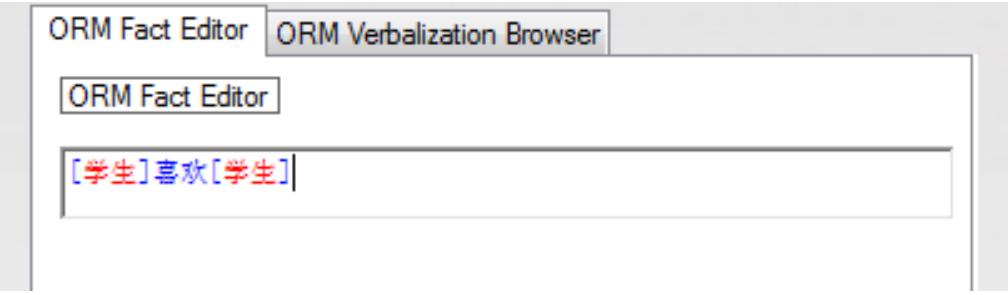


Figure 4.42 Entering a homogeneous fact type

{view ringconstraint.wmv video clip for demo}

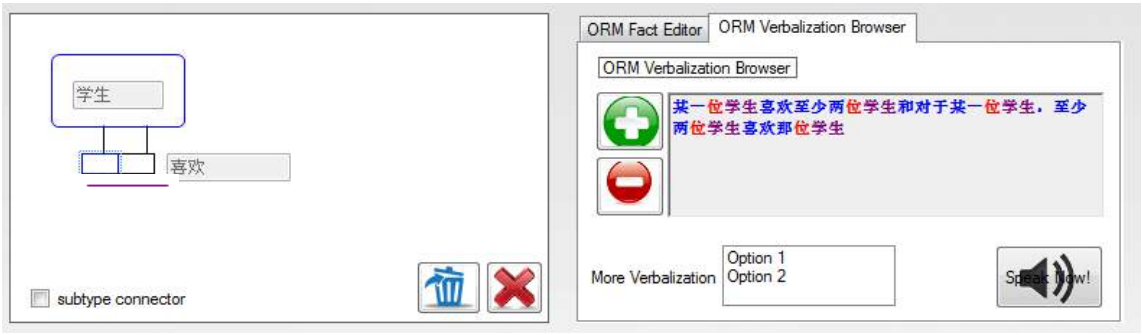


Figure 4.43 Displaying a homogeneous fact type

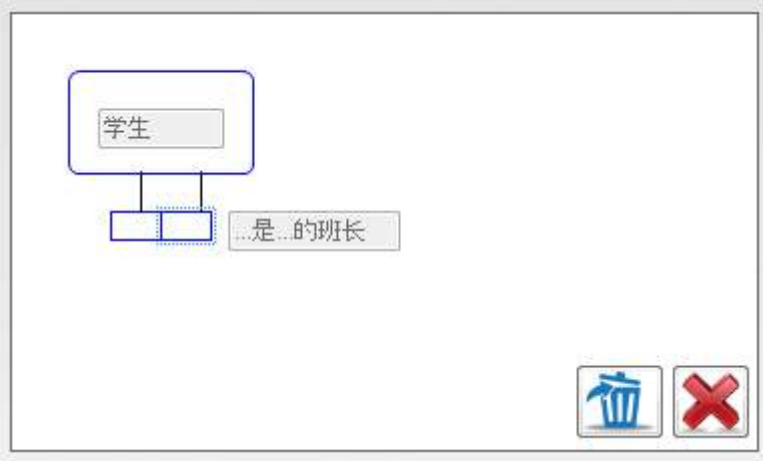


Figure 4.44 Selecting the roles to apply a ring constraint over 'Student is class monitor for Student'

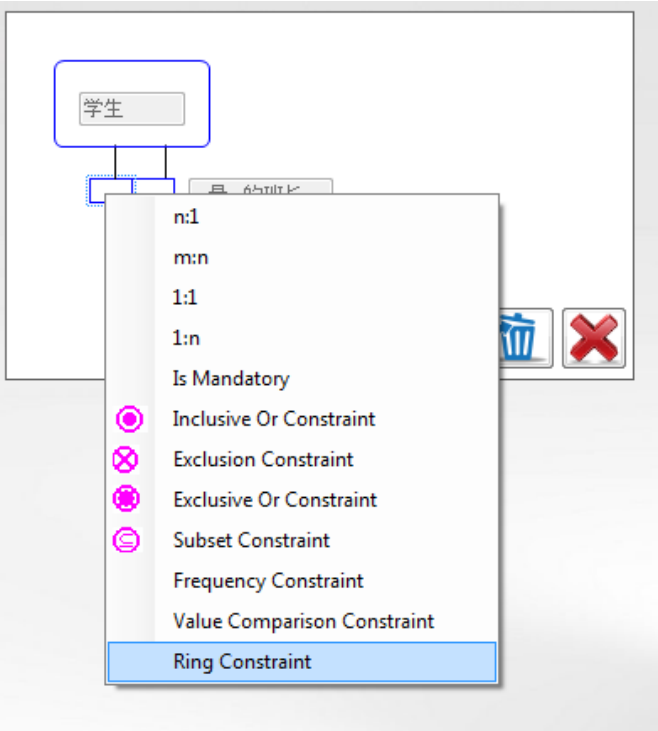


Figure 4.45 Choosing to add a ring constraint

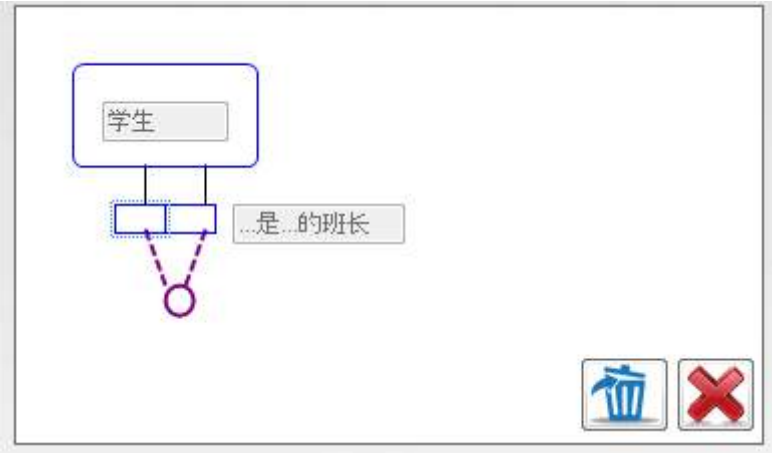


Figure 4.46 Initial display of a ring constraint (specific type not yet specified)

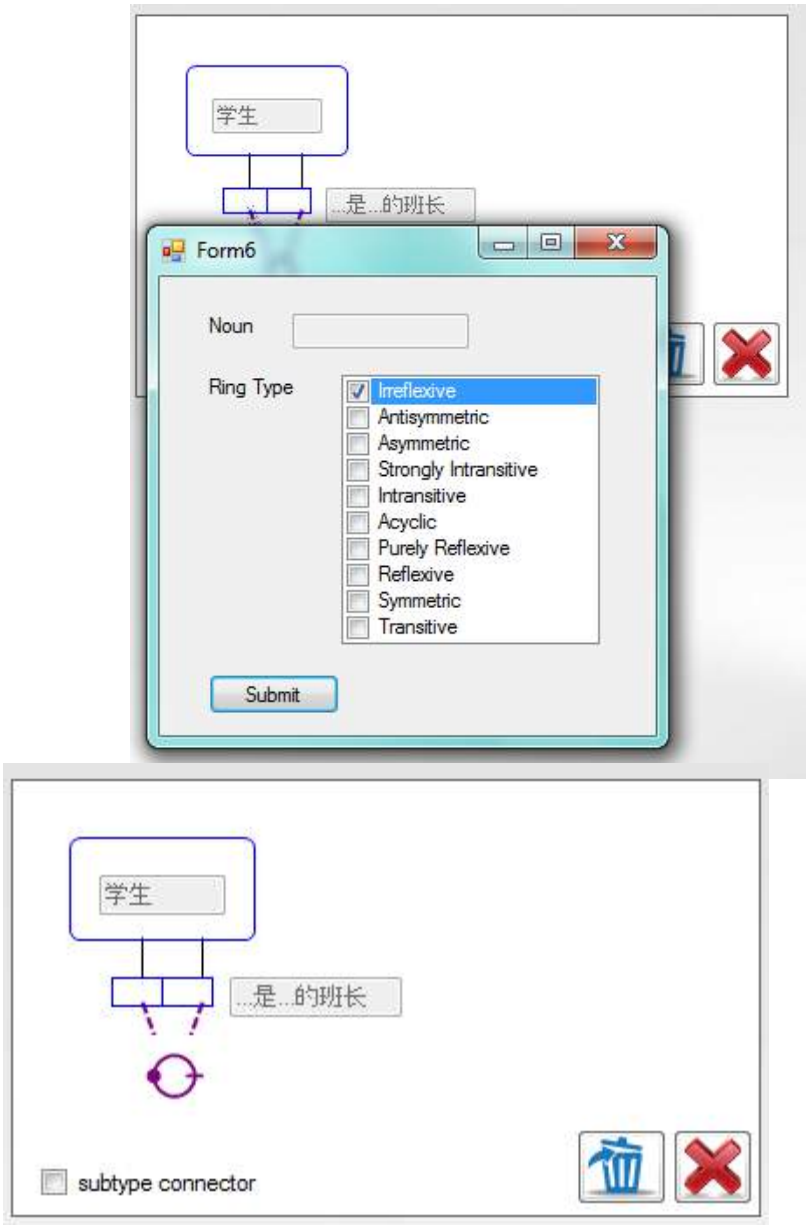


Figure 4.47 Selecting an irreflexive ring constraint

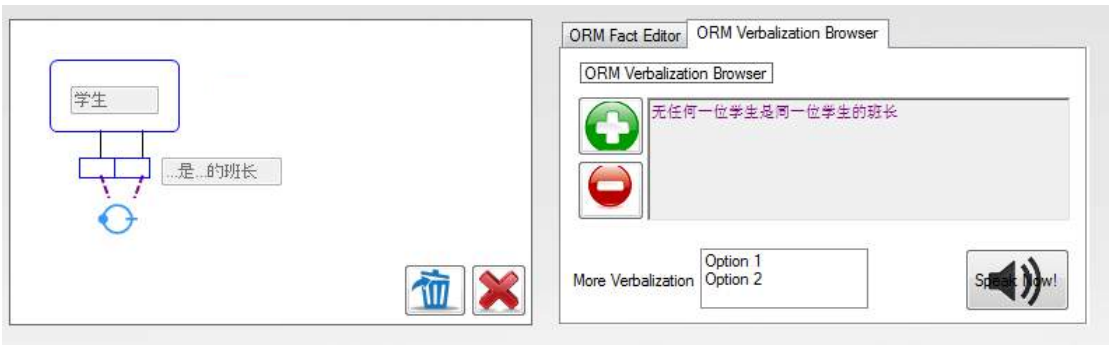


Figure 4.48 Displaying and verbalizing an irreflexive ring constraint

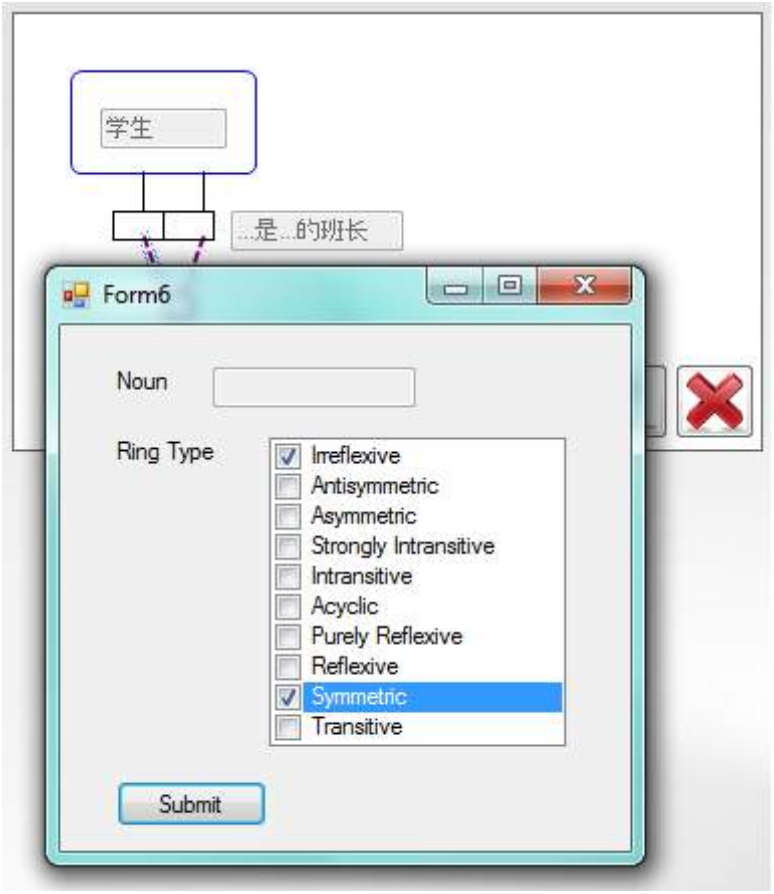


Figure 4.49 **Selecting a combination of irreflexive and symmetric ring constraints**

Figure 4.50 **Displaying and verbalizing irreflexive and symmetric ring constraints**

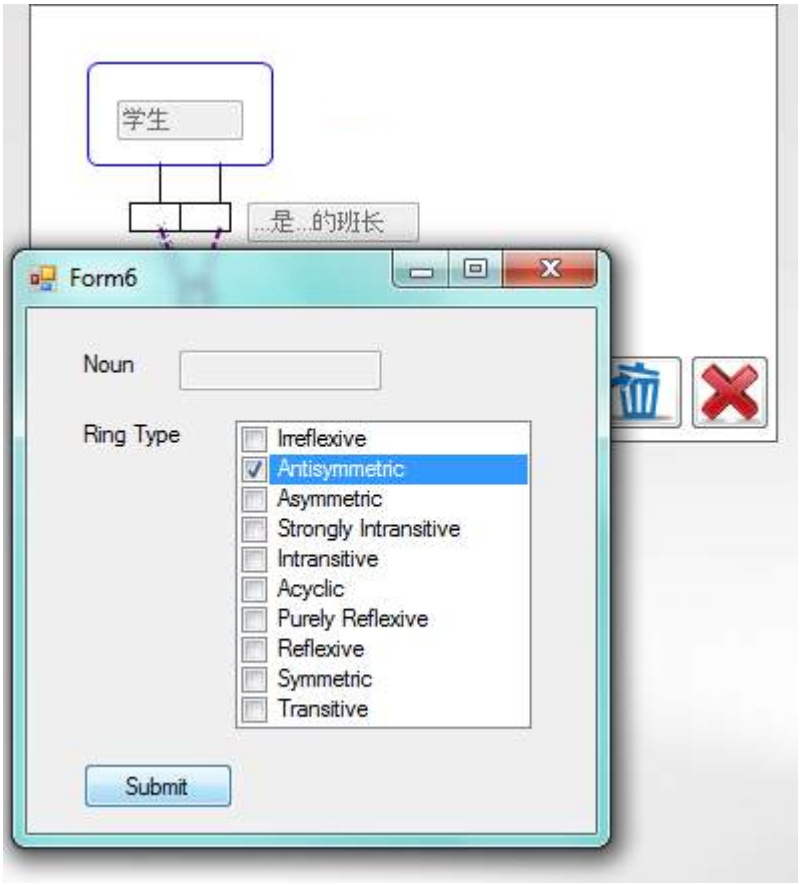


Figure 4.51 Selecting an antisymmetric ring constraint

Figure 4.52 **Displaying and verbalizing an antisymmetric ring constraint**

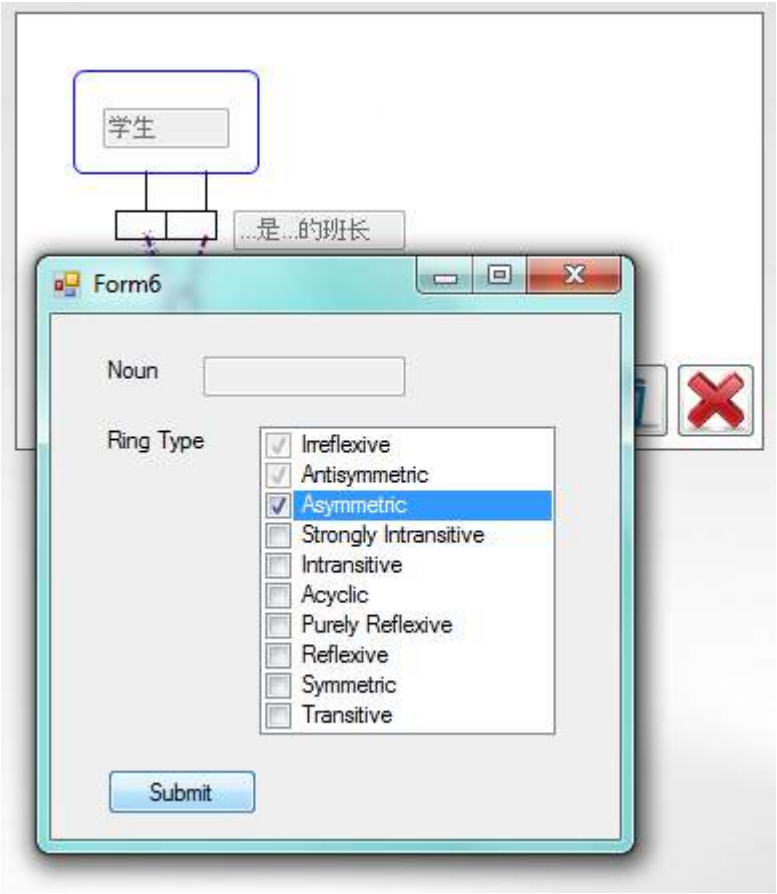


Figure 4.53 **Selecting an asymmetric ring constraint**

Figure 4.54 **Displaying and verbalizing an asymmetric ring constraint**

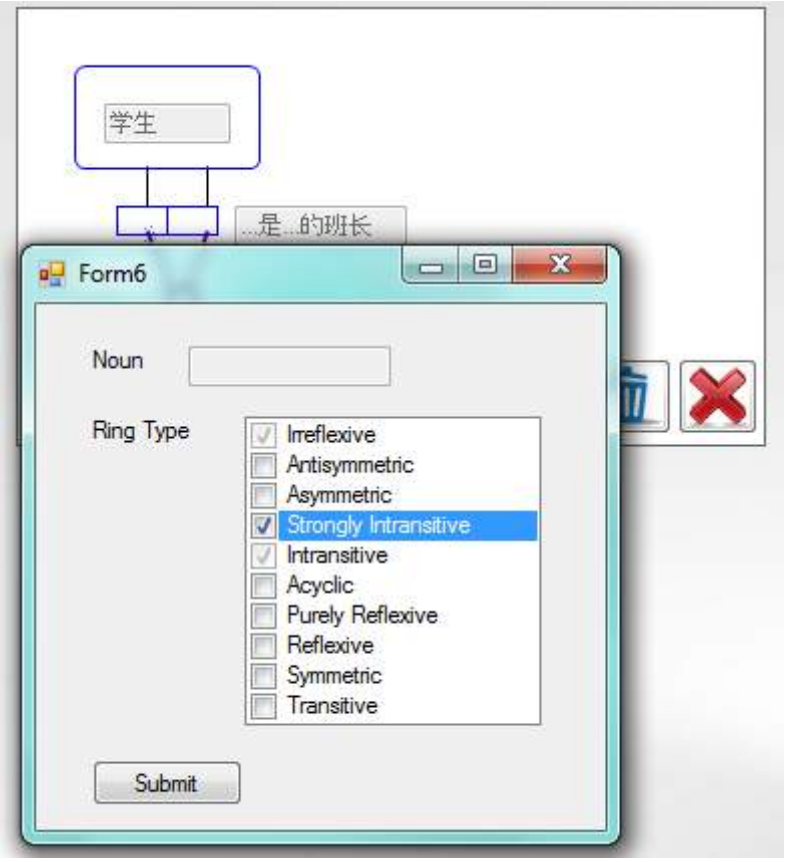


Figure 4.55 **Selecting a strongly intransitive ring constraint**

Figure 4.56 Displaying and verbalizing a strongly intransitive ring constraint

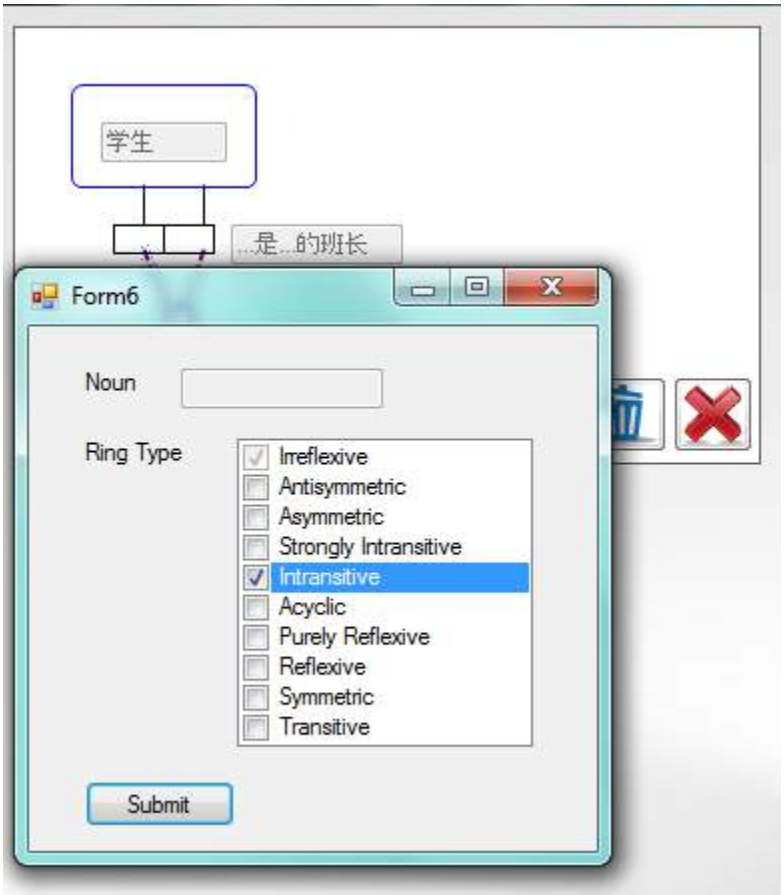


Figure 4.57 Selecting an intransitive ring constraint

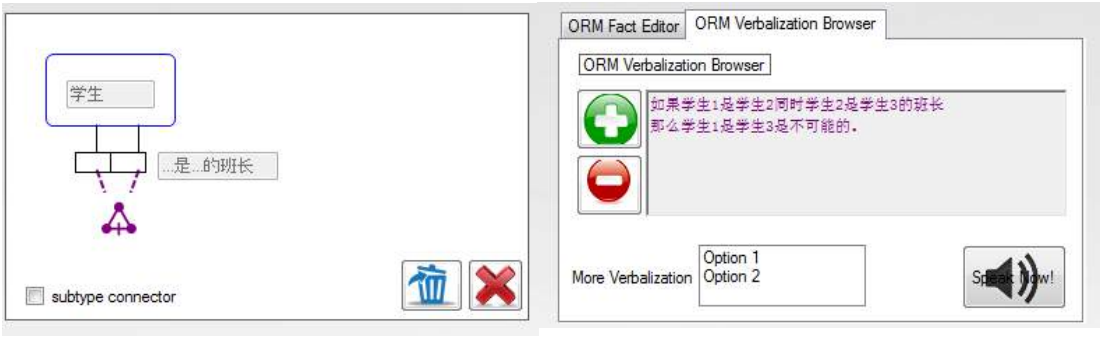


Figure 4.58 Displaying and verbalizing an intransitive ring constraint

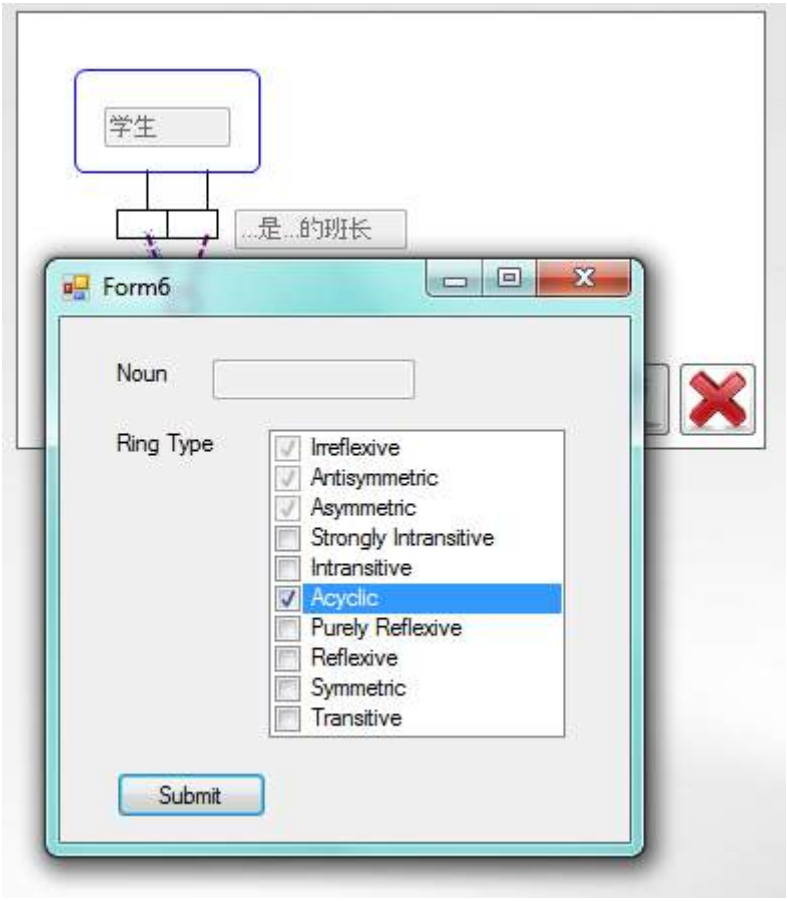


Figure 4.59 Selecting an acyclic ring constraint

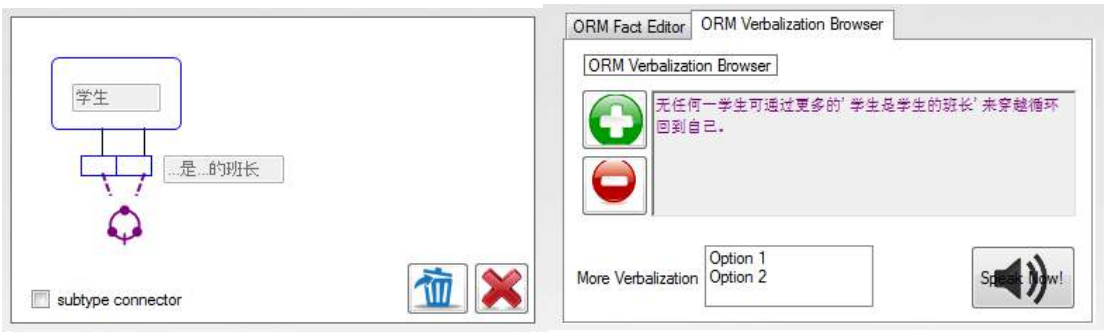


Figure 4.60 **Displaying and verbalizing an acyclic ring constraint**

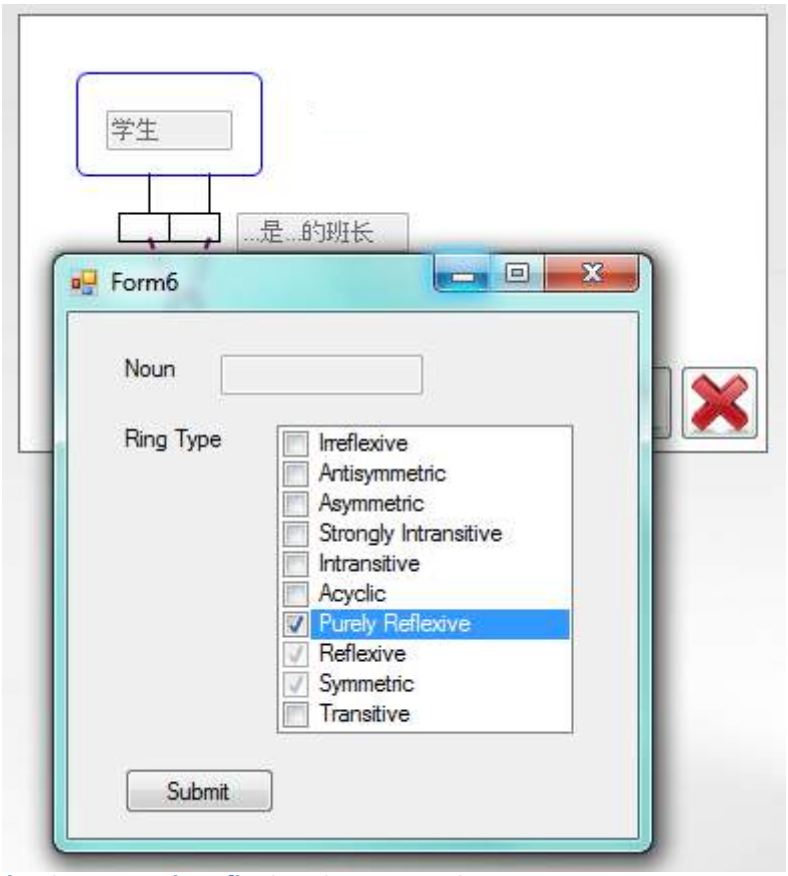


Figure 4.61 **Selecting a purely reflexive ring constraint**

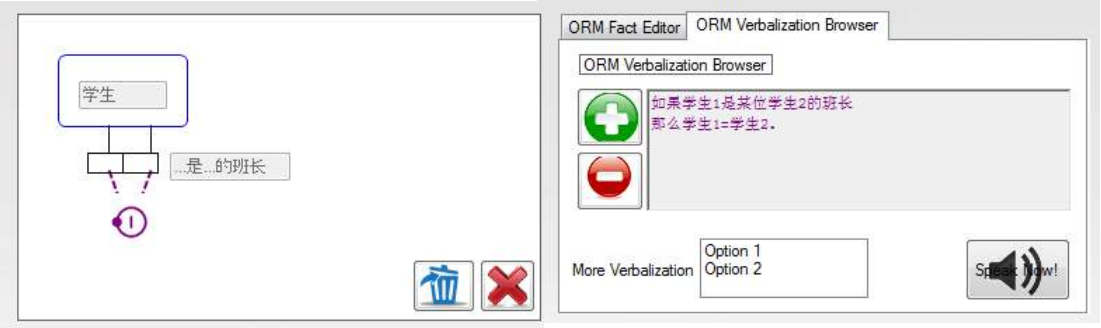


Figure 4.62 **Displaying and verbalizing a purely reflexive ring constraint**

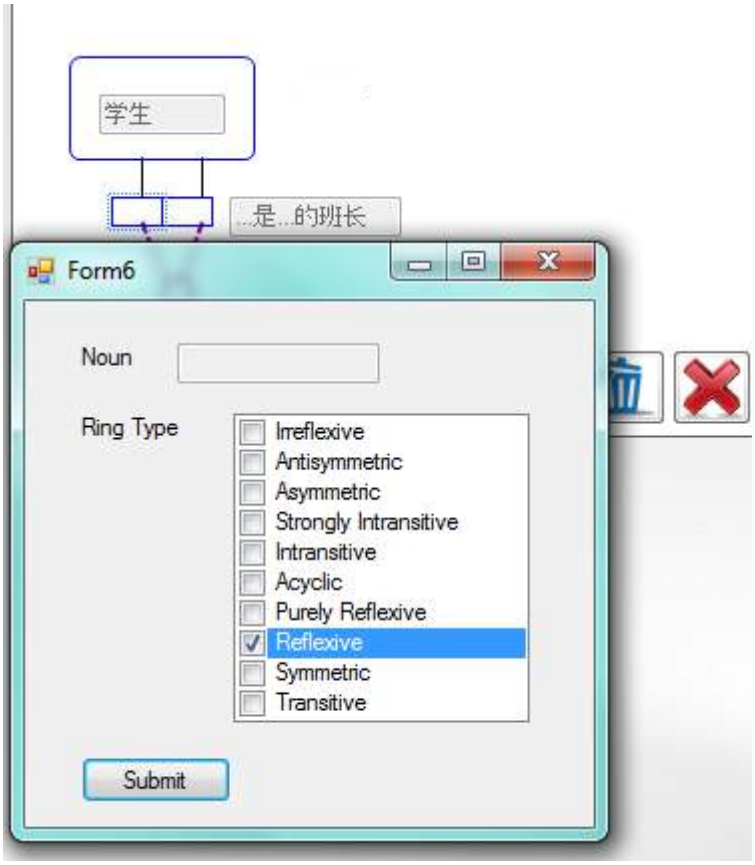


Figure 4.63 **Selecting a reflexive ring constraint**

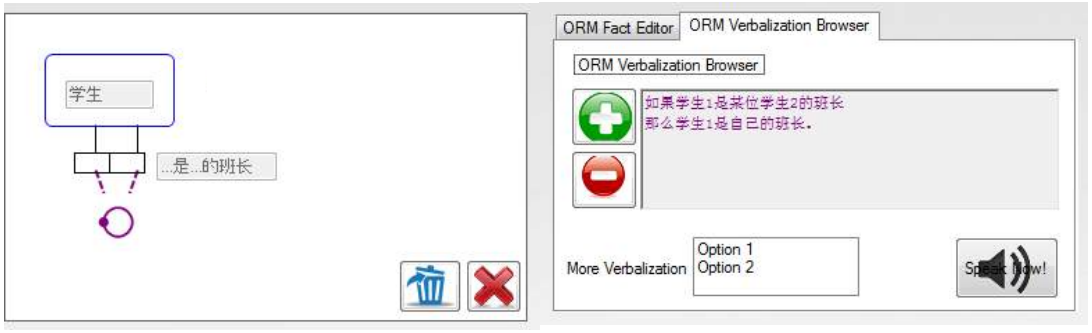


Figure 4.64 Displaying and verbalizing a reflexive ring constraint

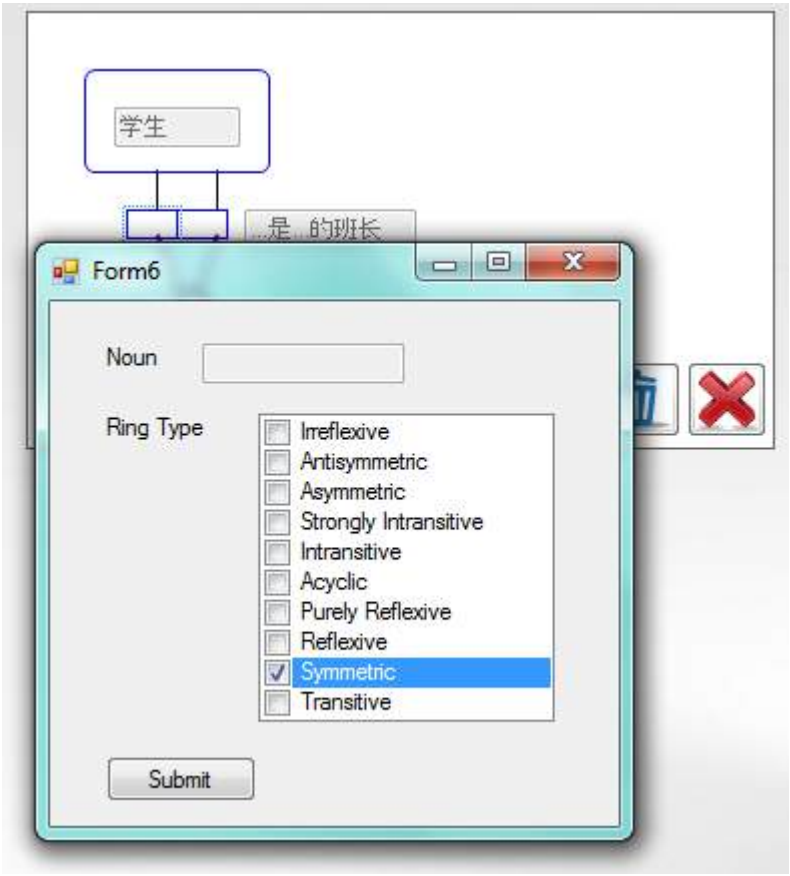


Figure 4.65 Selecting a symmetric ring constraint

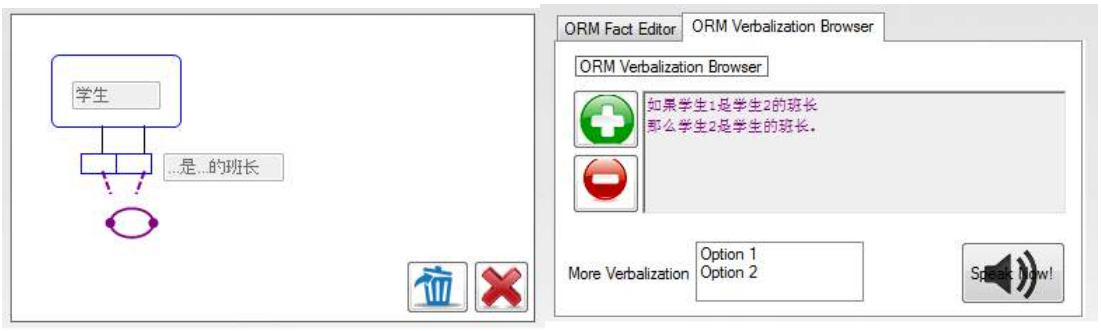


Figure 4.66 Displaying and verbalizing a symmetric ring constraint

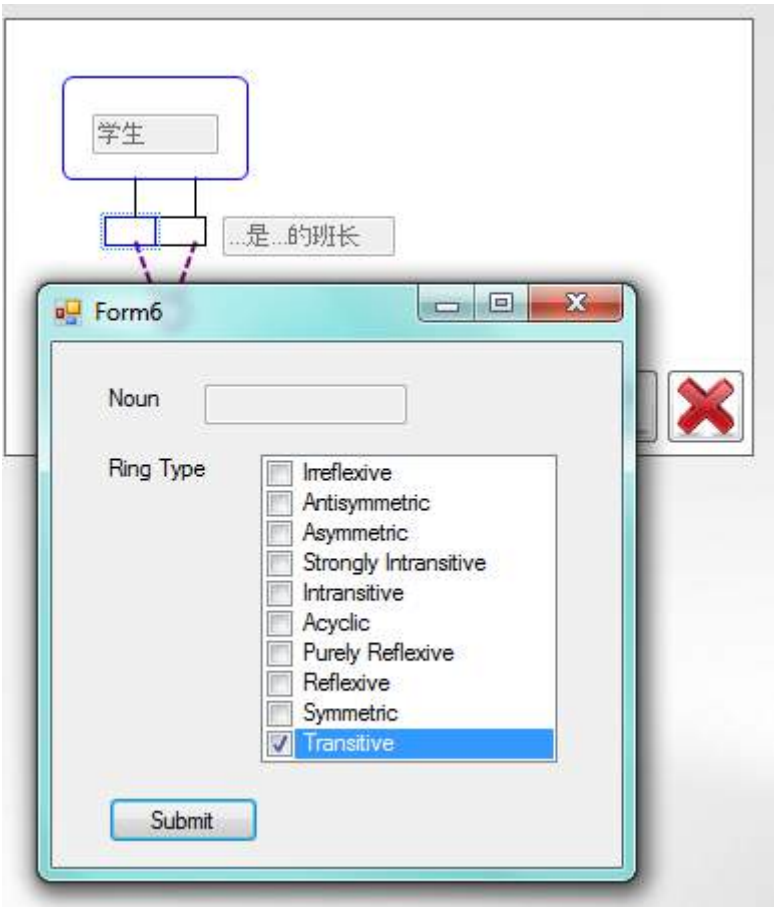


Figure 4.67 Selecting a transitive ring constraint

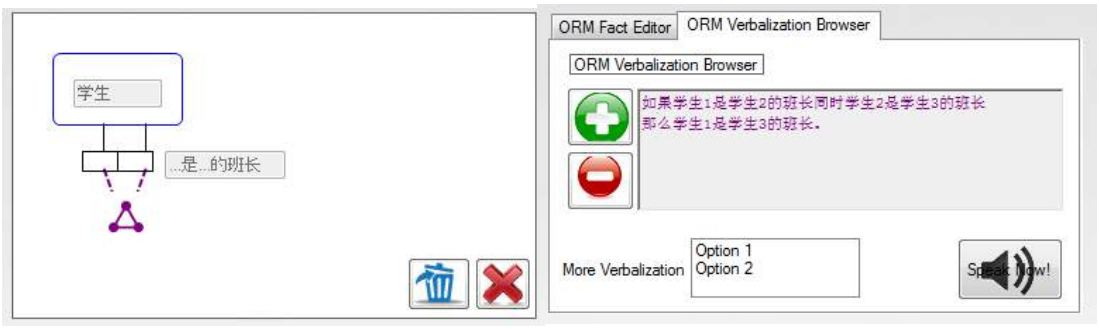


Figure 4.68 Displaying and verbalizing a transitive ring constraint

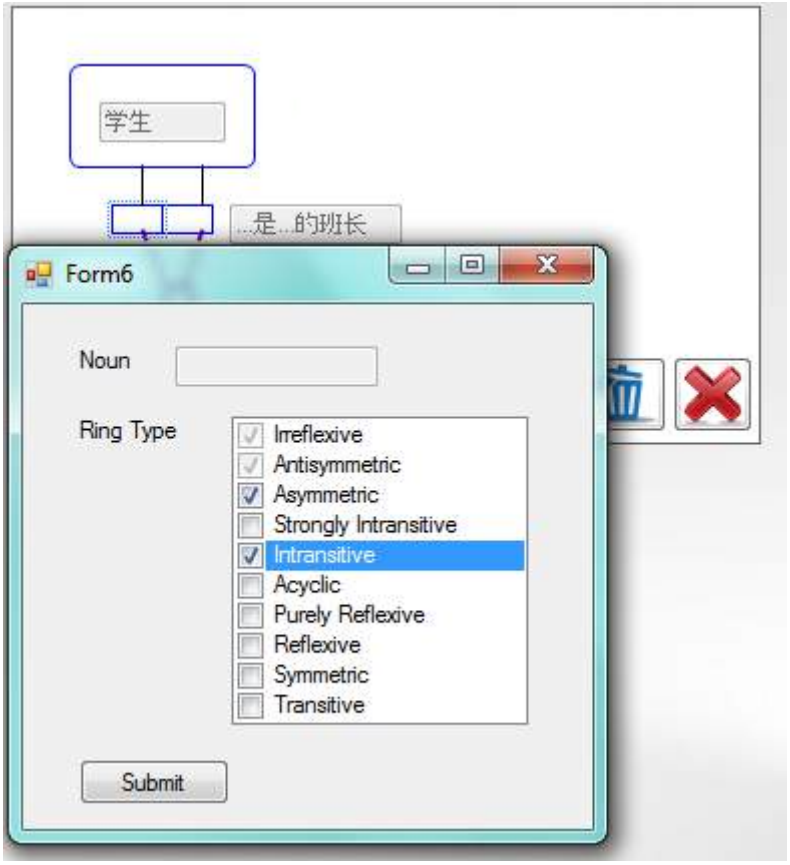


Figure 4.69 Selecting an symmetric and intransitive ring constraint combination

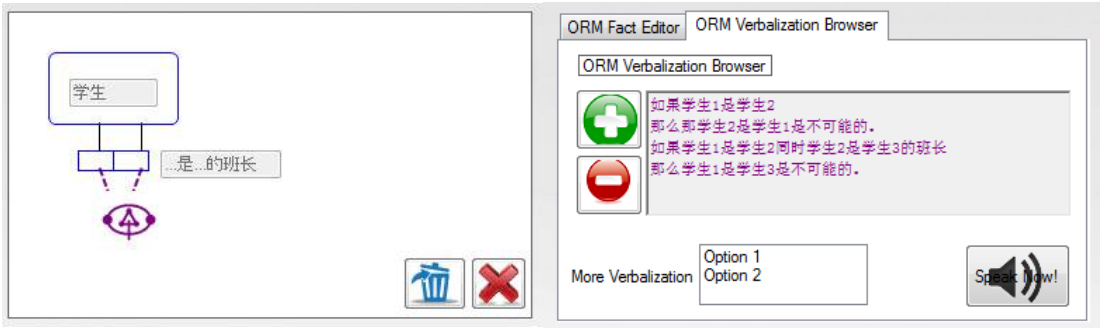


Figure 4.70 Displaying and verbalizing asymmetric and intransitive ring constraints

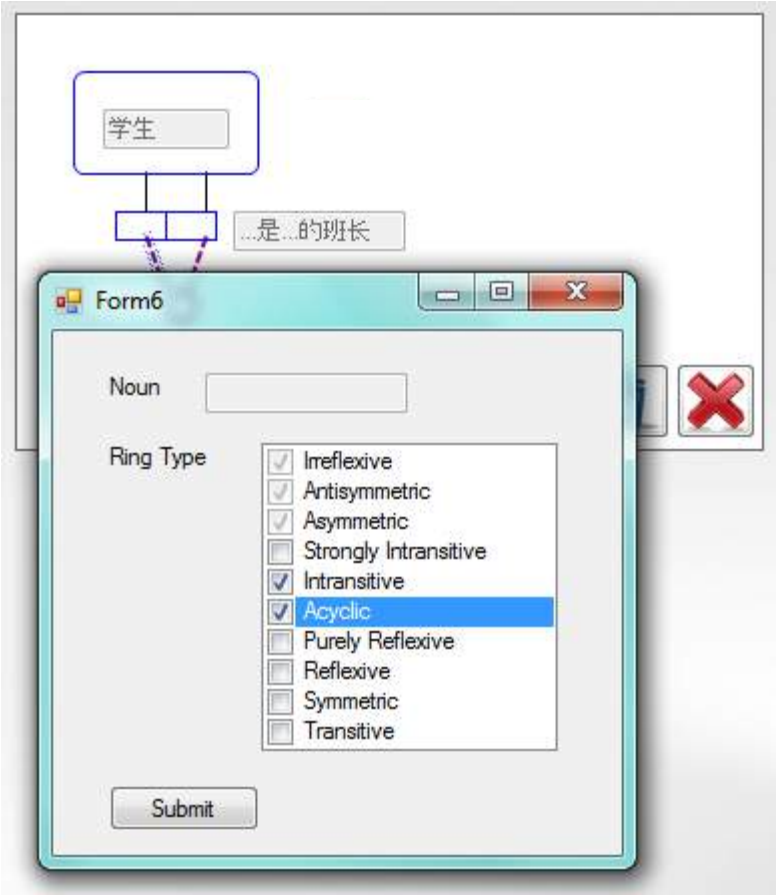


Figure 4.71 Selecting an acyclic and intransitive ring constraint combination

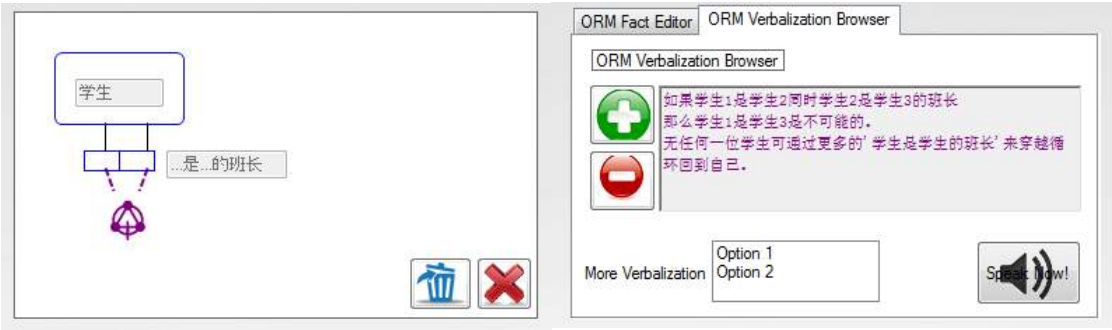


Figure 4.72 Displaying and verbalizing acyclic and intransitive ring constraints

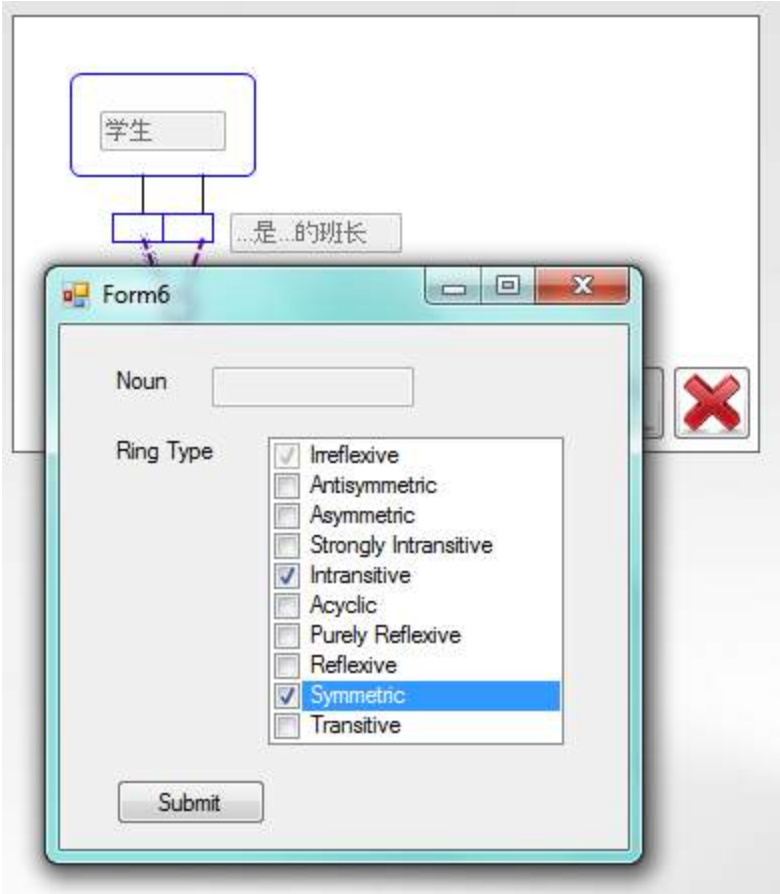


Figure 4.73 Selecting an intransitive and symmetric ring constraint combination

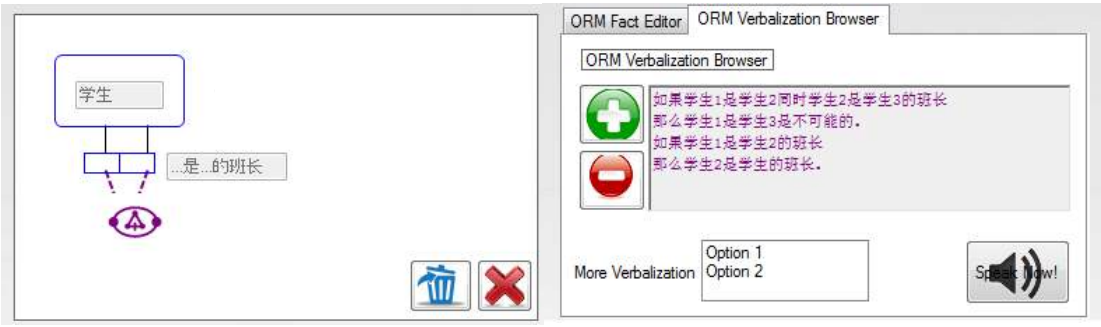


Figure 4.74 Displaying and verbalizing intransitiveand symmetric ring constraints

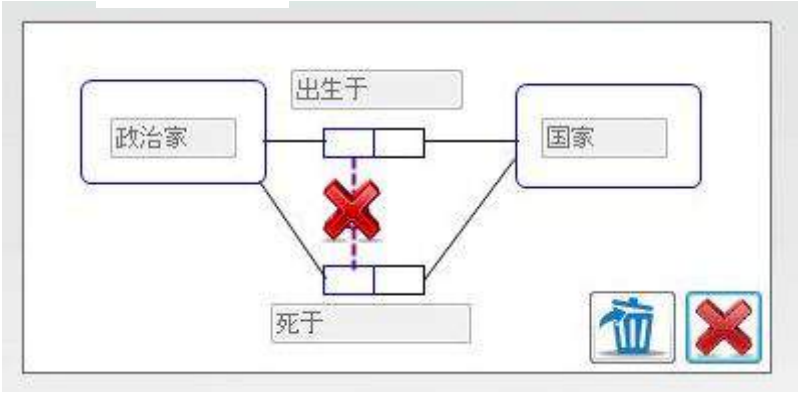


Figure 4.75 Deleting a constraint by moving an X-cursor over its shape

Figure 4.76 Entering and displaying a ternary fact type in Mandarin for 'Student obtains Mark in Course'

view ternary.wmv video clip for demo}

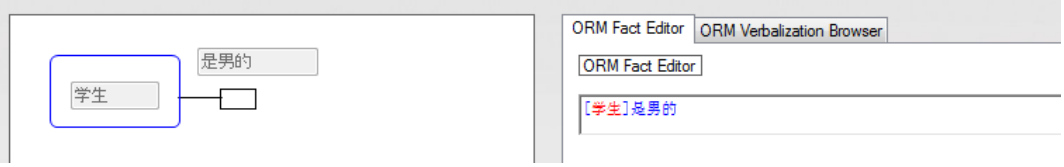


Figure 4.77 Entering and displaying a unary fact type in Mandarin for 'Student is a male'

{view unary.wmv video clip for demo}

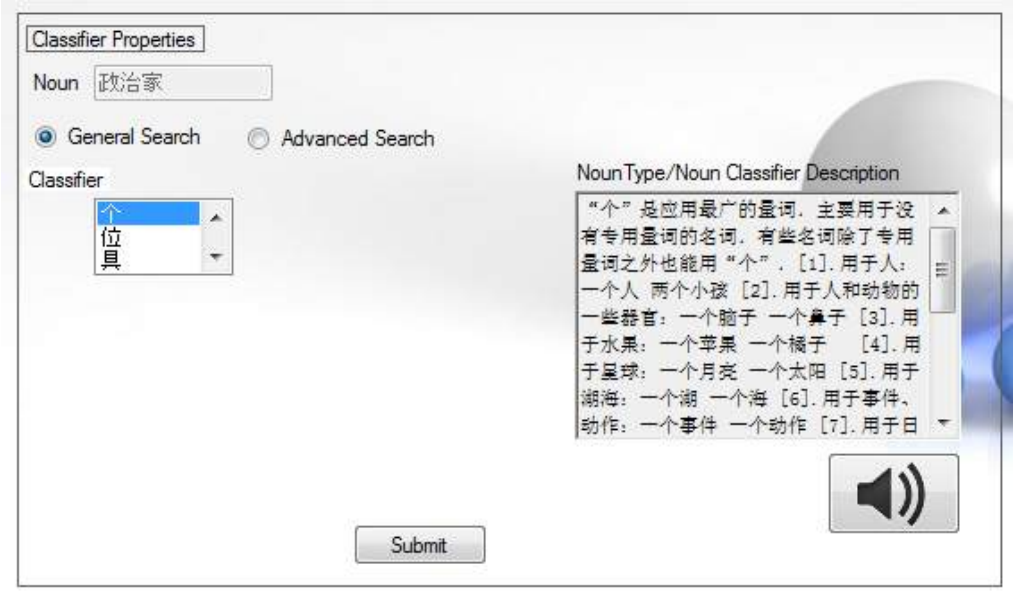


Figure 4.78 General search for noun types and descriptions with speech narrator

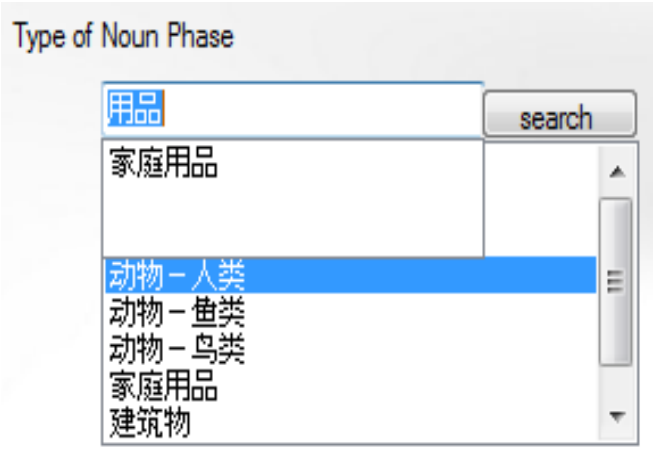


Figure 4.79 Searching for noun phrase types via specific list by typing any character in one word; and for the specific searched word the matching character will show up.

Figure 4.80 Advanced search for noun phrase via auto-complete mode by typing the first character with/without subsequent characters of the word. In this case, all the possible matching words will show up automatically.

{view searchspecificword&autocompleteword.wmv video clip for demo}

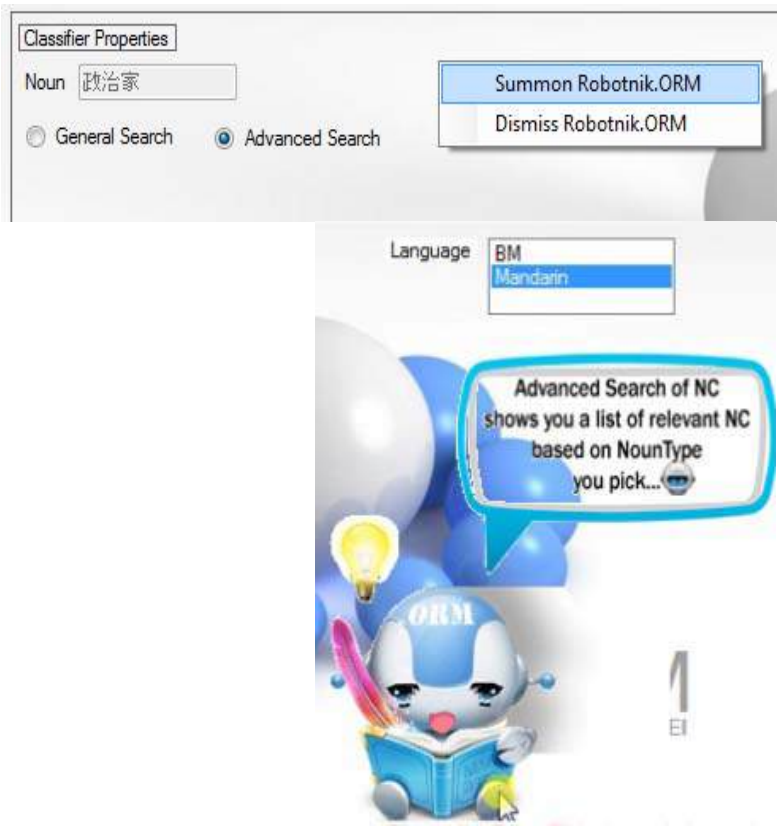


Figure 4.81 Invoking the Robotnik "Little Helper" for assistance with verbalizations in ORM.

{view robotnikorm.wmv video clip for demo}

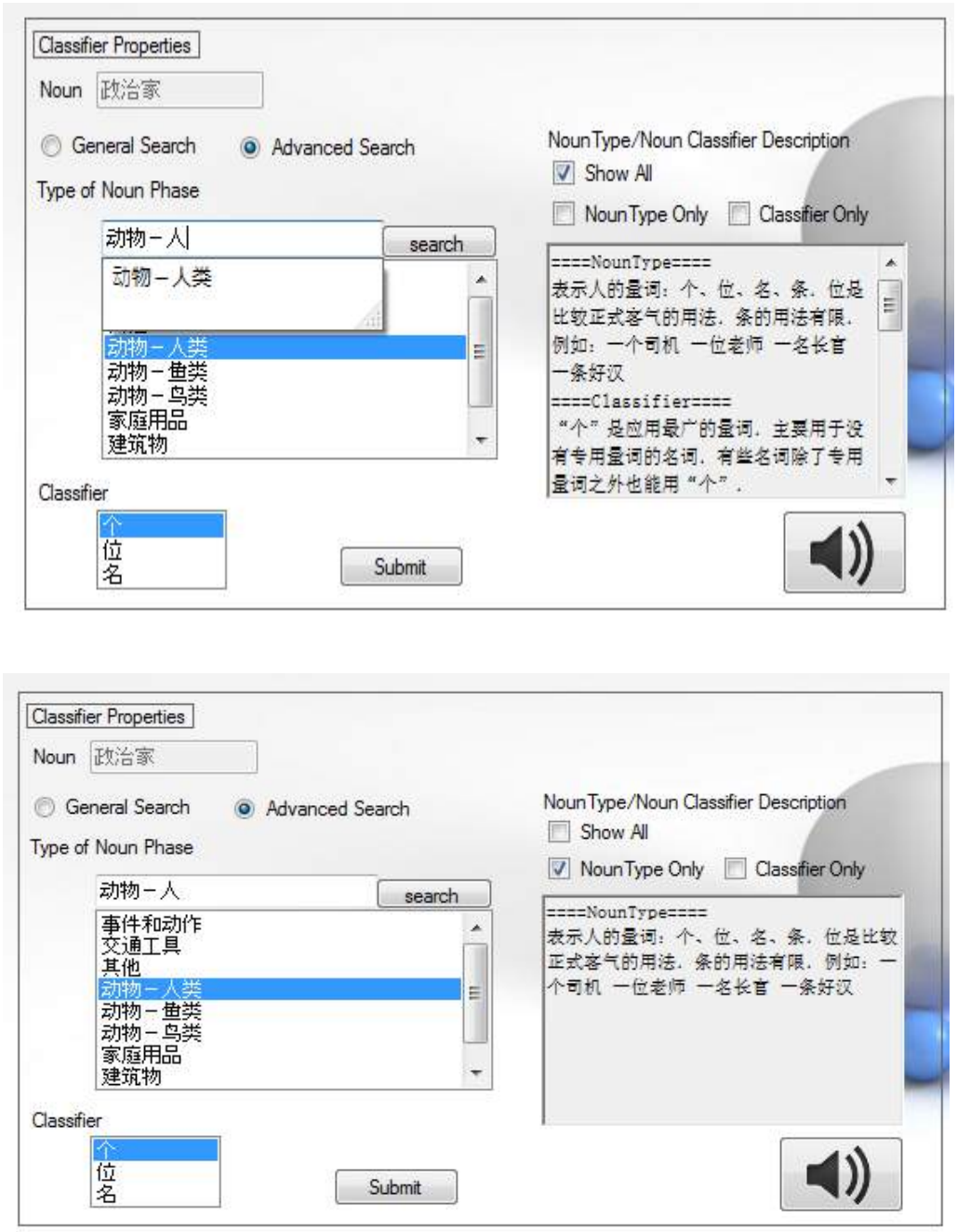


Figure 4.82 Advanced search by accessing noun type/noun classifier description using the speech narrator

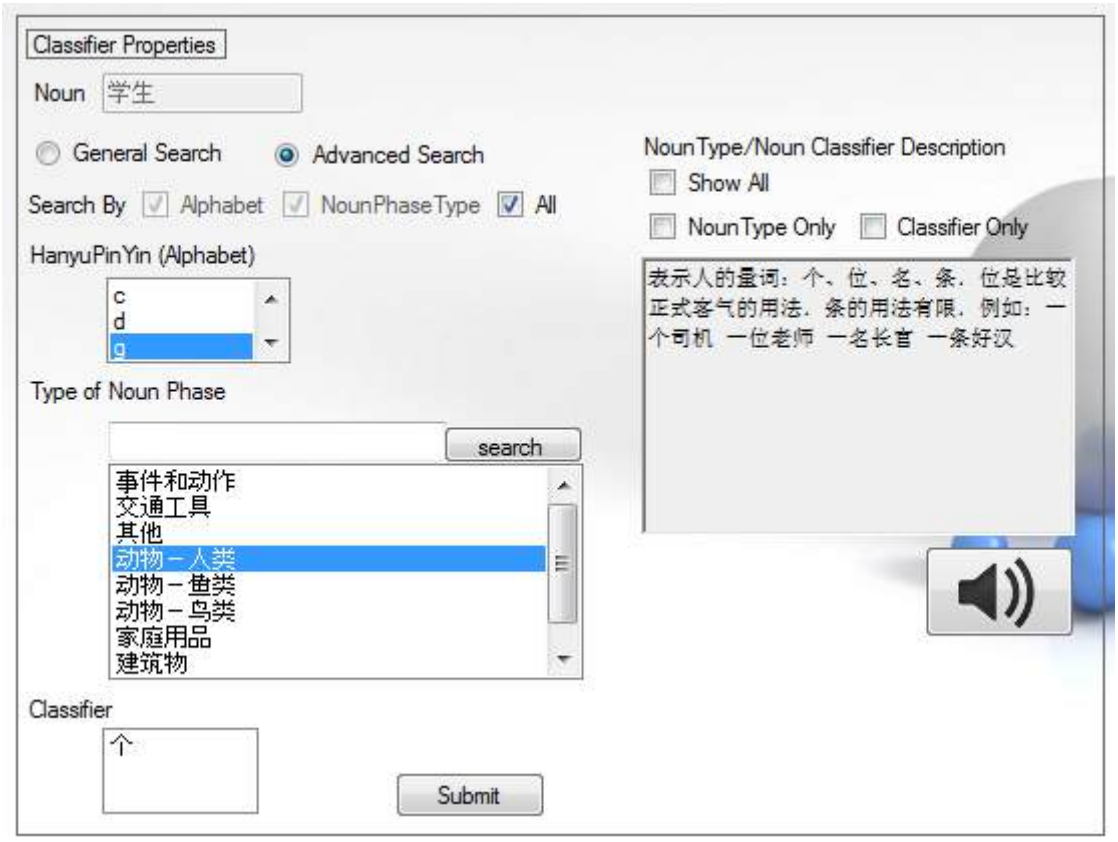


Figure 4.83 Advanced search using alphabetically sorting and noun types

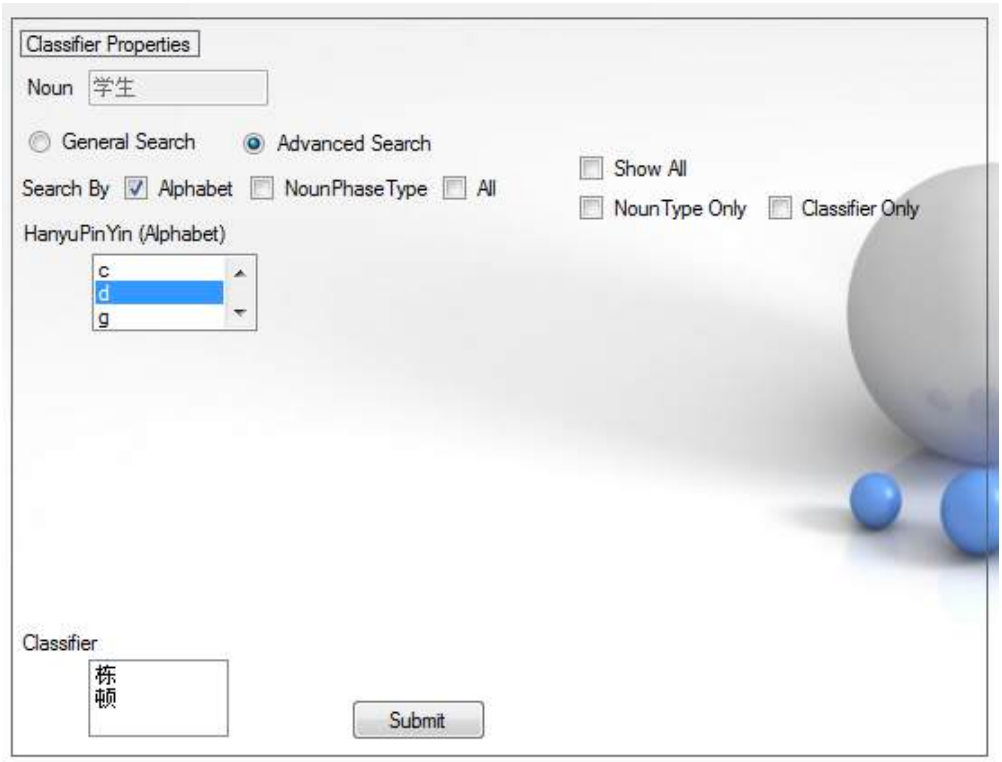


Figure 4.84 Advanced search using alphabetically sorting only

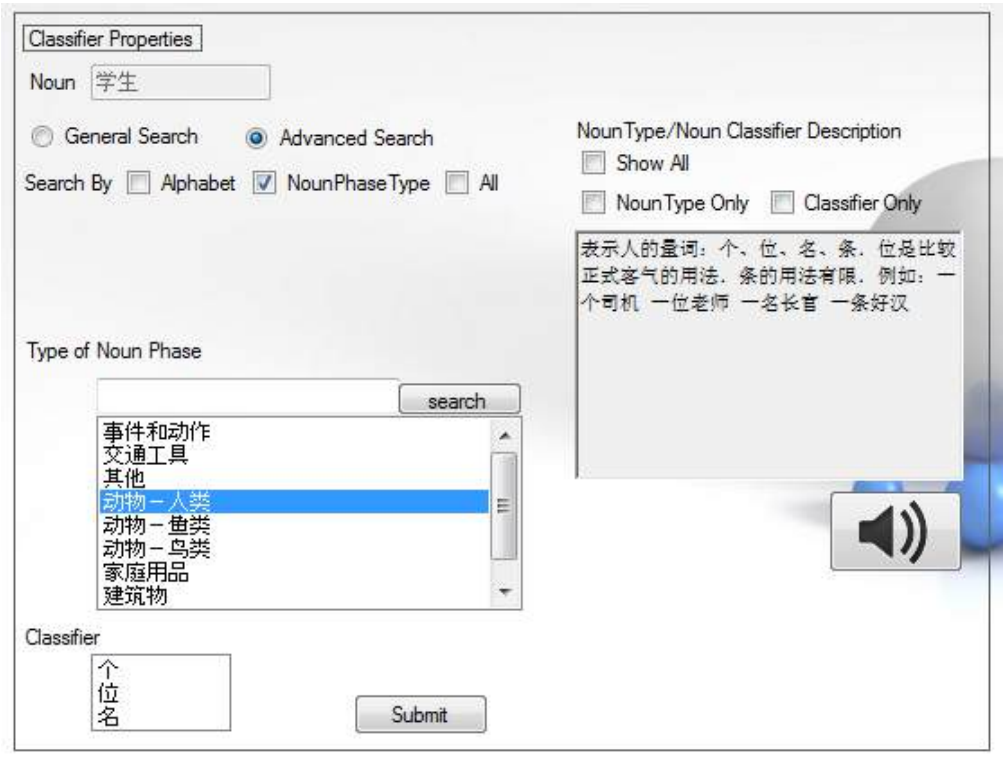


Figure 4.85 Advanced search using noun phrase type only

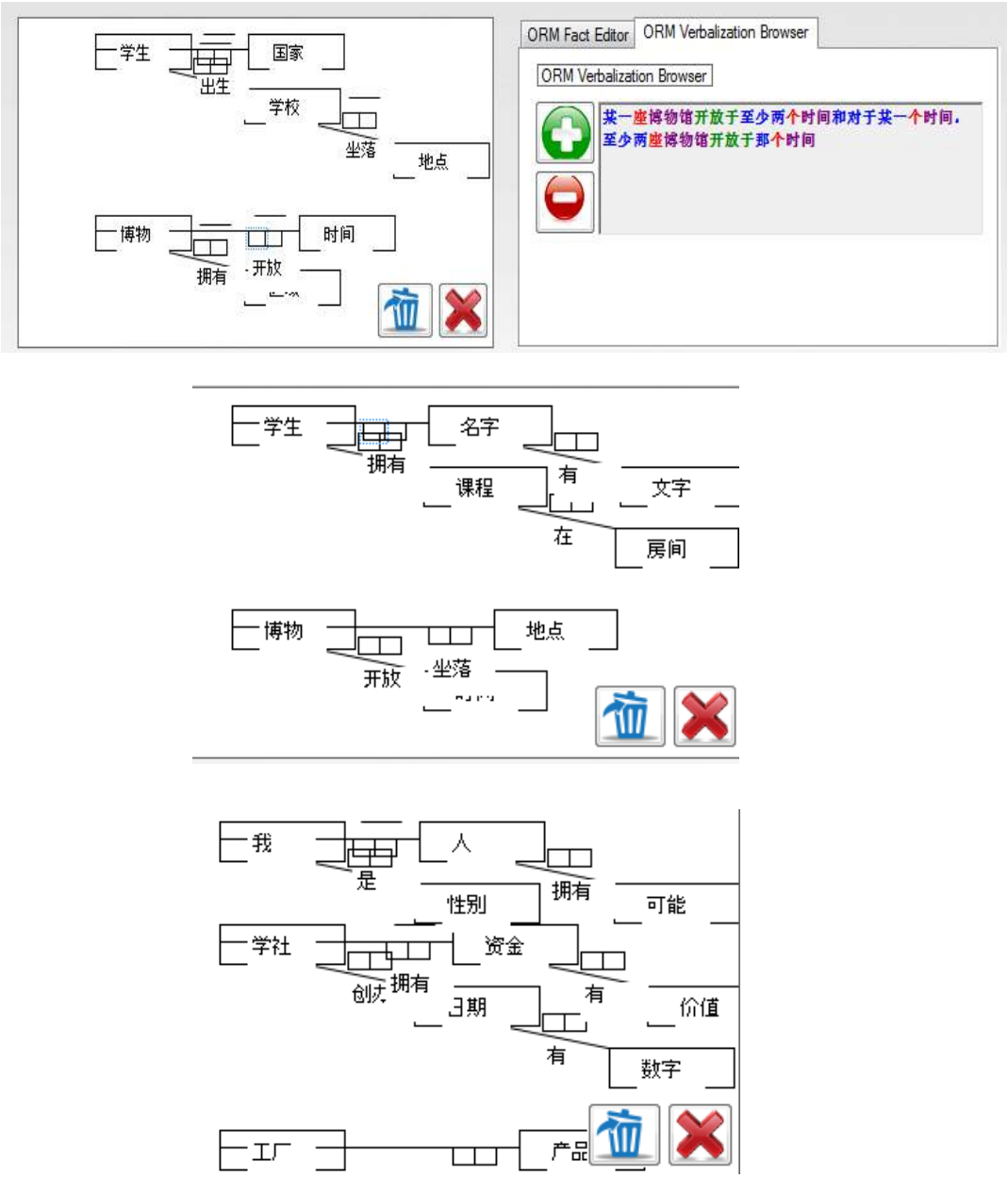


Figure 4.86 Support formodels with lare numbers of fact types

{ view [100+factentry.avi](#) video clip for demo}

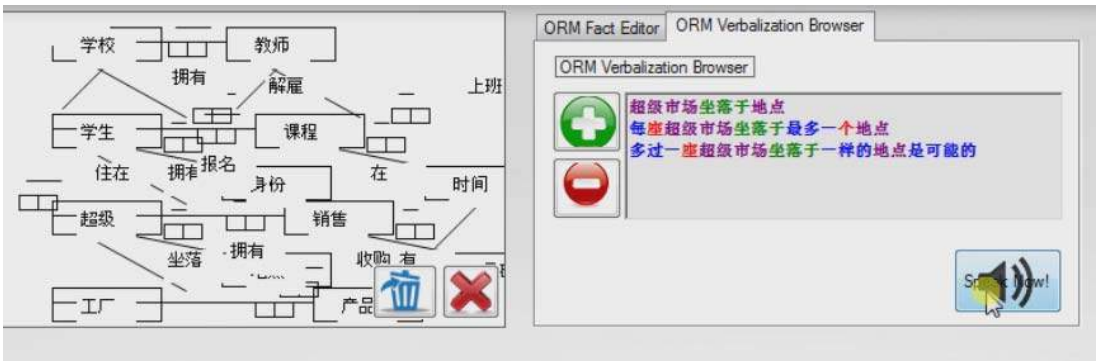


Figure 4.87 Verbalization with speech narrator (real time text to speech)

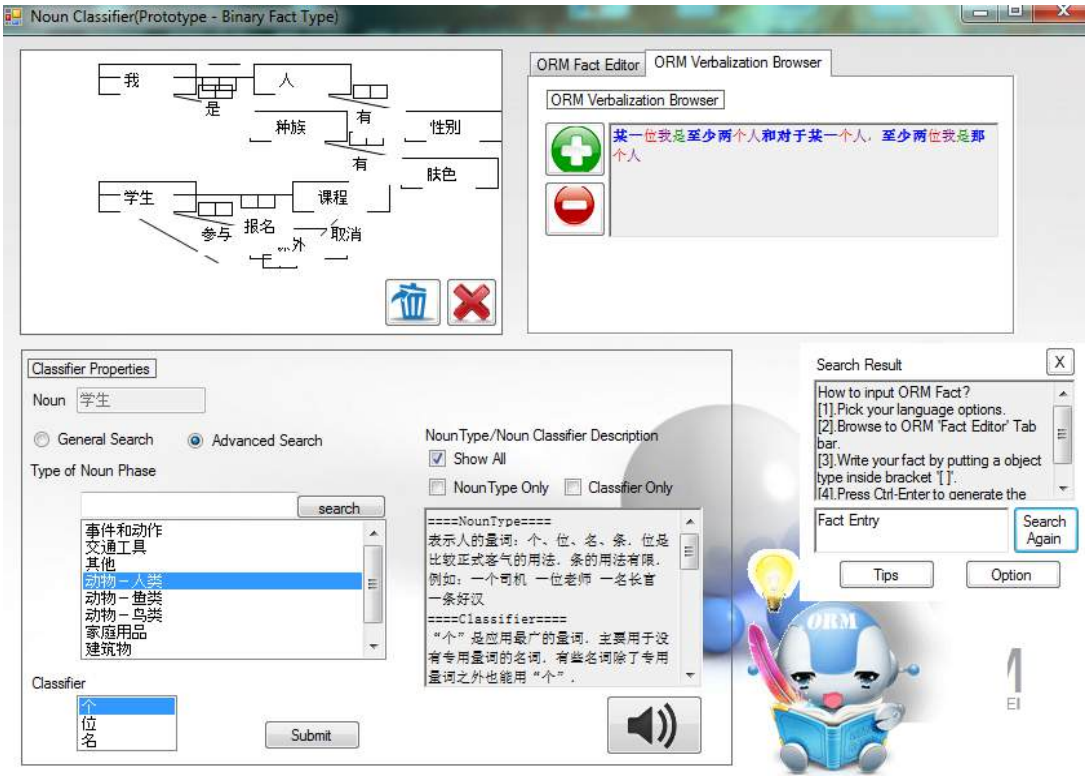


Figure 4.88 Overview of our prototype showing multiple features

Simple Subtyping [view simpleSubtyping.wmv video clip for following demo demo]

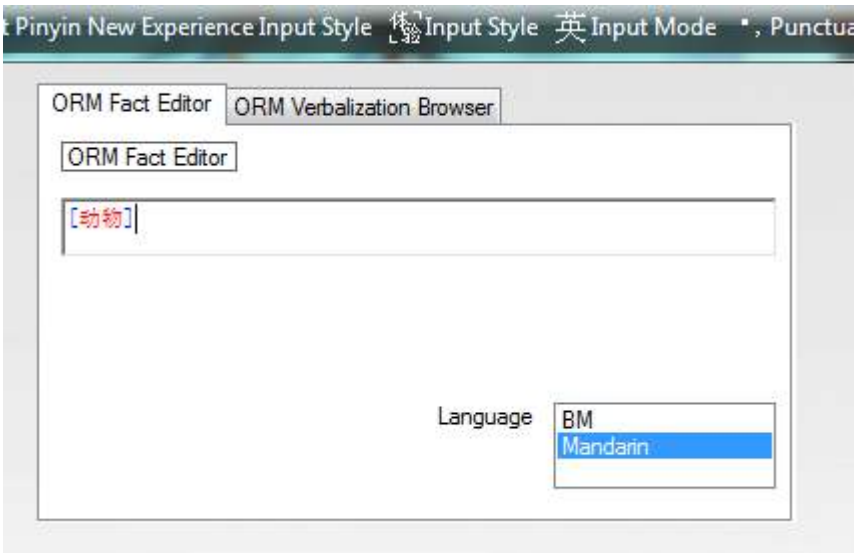


Figure 4.89 Inputting the object type Animal in Mandarin

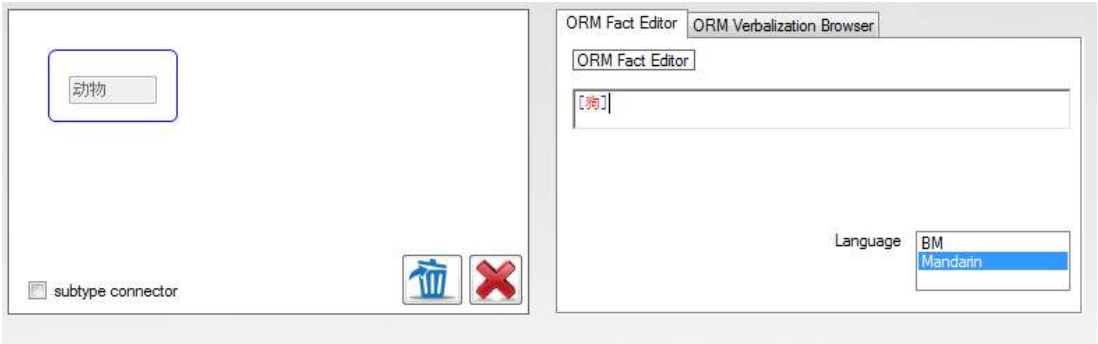


Figure 4.90 Inputting the object type Dog in Mandarin

Figure 4.91 shows a screenshot of the subtyping connection from Dog to Animal as well as its verbalization in Mandarin.

Figure 4.91 Adding and verbalizing a subtyping connection in Mandarin

4. 2 C# Code

In this section, we provide a representative sample of the C# code that we wrote to implement our prototype, covering verbalization, model creation, classifier setup and verbalization output in Bahasa Melayu and Mandarin.

```
---------------------------------------------------------------
1. MODEL CREATE SAMPLE CODE
---------------------------------------------------------------
int index3 = str.IndexOf('[');
            index3 = str.IndexOf('[', index3 + 1);
int index4 = str.IndexOf(']');
            index4 = str.IndexOf(']', index4 + 1);
textBox11.Text = str.Substring(index4 + 1);
if (textBox11.Text != String.Empty)
            {

string value = str.Substring(index3 + 1);
string value2 = str.Substring(index4 + 1);

int b = value.Length - (value2.Length+1) ;


                tb2.Text = str.Substring(index3 + 1, b);

result = str.Substring(index5 + 1, index6 - index5 - 1);

                tb3.Text = "..." + result +"..." + textBox11.Text;

result = result.Trim();
                textBox11.Text = textBox11.Text.Trim();
            }
```

Explanation: This set of C# commands inputs the user's Chinese predicate readings to insert into the generated model textbox. As discussed, the arrangement of typical Chinese phrases differs from English and BM. For example, Chinese: 人是国家的总统. English: Person is president of Country. As you can see, the predicate reading in English shown above uses infix layout, with the predicate reading placed right in middle of two object type nouns. In contrast, Chinese predicate readings require mixfix layout using placeholders within the predicate reading for the object type names. So, this set up helps to solve this issue.

---

2. Basic CLASSIFIER SETUP SAMPLE CODE

```
if (listBox1.SelectedIndex >= 0)
        {

if (listBox4.SelectedItem.ToString() == "Mandarin")
          {
SqlConnection hookup2 = new
SqlConnection("Server=Lim-PC\\SQLEXPRESS;Database=Noun;" +
"Trusted_Connection=True;");


SqlCommand sqlCmd2 = newSqlCommand
                ("Select DISTINCT Class From Noun2 Where Noun = N'" +
listBox1.SelectedItem.ToString() + "'", hookup2);

hookup2.Open();

SqlDataReader reader2 = sqlCmd2.ExecuteReader();

if (reader2.HasRows)
              {

while (reader2.Read())
                {

string name2 = Convert.ToString(reader2["Class"]);
// textBox6.Text = name2;
listBox2.Items.Add(name2);
                }

reader2.Close();
hookup2.Close();
              }
          }
```

Explanation: This set of C# commands is to retrieve a list of classifiers based on the chosen type for noun phrase for Chinese language. Since Chinese words are Unicode based, an 'N' character is placed in front of the searched values in order to match the data in the database (unlike BM and English).

---

3. VERBALIZATION OUTPUT SAMPLE CODE

```
richTextBox1.Text = tb.Text + result + tb2.Text + textBox11.Text + "\r\n" +
"对于至少一" + textBox6.Text + tb2.Text + "," + "多过一" +
                textBox5.Text + tb.Text + result + "那一" + textBox6.Text + tb2.Text
+
                textBox11.Text + "是不可能的" + "\r\n"
                + "至少一" + textBox5.Text + tb.Text + result + "多过一" +
textBox6.Text + tb2.Text
                + textBox11.Text + "是可能的";
```

Explanation: This set of C# command shows a portion of the Chinese verbalization result that is generated based on the chosen uniqueness constraint pattern for binary fact types.

OTHER Sample C# Code in ORM prototype

IsMandatory/ relationship (1:1,1:n,n:1,m:n)

1) To access the popup menu and to generate the required symbol

```
if (Convert.ToString(item.Text) == "Is Mandatory") in this case,
{
   theShape11.Parent = canvas;
   theShape11.Size = new System.Drawing.Size(6, 7);
   theShape11.Location = new System.Drawing.Point(144, 54);
   theShape11.BackStyle = BackStyle.Opaque;
   theShape11.BackColor = Color.Black;
   p4.Controls.Add(canvas);
   mandcheck = “绝对”;
    }
if (textBox9.Text == "n:1")
{
   ls5.Parent = canvas;
   ls5.StartPoint = new System.Drawing.Point(149, 48);
```

```
            ls5.EndPoint = new System.Drawing.Point(172, 48);
            ls5.BorderColor = Color.Purple;
            p4.Controls.Add(canvas);
        else if (textBox9.Text == "1:1")
        {
//shape object code list here
}
else if //others relationship
```

Type of fact (Unary, Binary, Ternary) & Types of Binary object combination

```
            string word = richTextBox2.Text;
            int count = word.Split('[').Length - 1;

            if (count ==1 ) // unary fact type
            {
                p2.Location = new System.Drawing.Point(130, 19);
                p2.Size = new System.Drawing.Size(92, 28);
                p4.Controls.Add(p2);
                tb1.Location = new System.Drawing.Point(43, 47);
                tb1.Size = new System.Drawing.Size(63, 20);
                p4.Controls.Add(tb1);
                tb1.Enabled = false;
                tb3.Location = new System.Drawing.Point(3, 4);
                tb3.Size = new System.Drawing.Size(86, 20);
                p2.Controls.Add(tb3);
                tb3.Enabled = false;
                canvas.Parent = this;
                theShape.Parent = canvas;
                theShape.Size = new System.Drawing.Size(92, 51);
                theShape.Location = new System.Drawing.Point(28, 28);
                theShape.CornerRadius = 5;
                theShape.BorderColor = Color.Blue;
```

```
            theShape2.Parent = canvas;
            theShape2.StartPoint = new System.Drawing.Point(119, 58);
            theShape2.EndPoint = new System.Drawing.Point(150, 58);
            theShape3.Parent = canvas;
            theShape3.Size = new System.Drawing.Size(25, 14);
            theShape3.Location = new System.Drawing.Point(149, 52);
            p4.Controls.Add(canvas);

            tb3.Text = str.Substring(index2 + 1);
        }
        else if (count == 2) // binary fact type ( main focus over here)
        {
          if (textBox11.Text == String.Empty)// if predicate text is sit between
two objects (example: [学生] 报读于 [学校])
            {
            string value = str.Substring(index3 + 1);
            int b = value.Length - 1;


            tb2.Text = str.Substring(index3 + 1, b);

            checkstr = str.Substring(index3 + 1, b);

            tb3.Text = str.Substring(index5 + 1, index6 - index5 - 1);
            }
            else if (textBox11.Text != String.Empty)// if predicate text is sits
between and after the objects involved(example: [学生] 是 [学生]的班长)
            {
            string value = str.Substring(index3 + 1);
            string value2 = str.Substring(index4 + 1);

            int b = value.Length - (value2.Length + 1);
```

```
                string bubu = str.Substring(str.LastIndexOf('[') + 1).TrimStart();

                textBox16.Text = bubu.Substring(0, bubu.IndexOf(']'));

                tb2.Text = str.Substring(index3 + 1, b);
                checkstr = str.Substring(index3 + 1, b);

                result = str.Substring(index5 + 1, index6 - index5 - 1);

                tb3.Text = "..." + result + "..." + textBox11.Text;

                result = result.Trim();
                textBox11.Text = textBox11.Text.Trim();
            }
            if (tb2.Text == tb1.Text) // if  1nd fact entry's heading object type similar with its ending object type , then straightaway connect to it (example: [学生]是 [学生]的班长)
            {
              /* p2.Location = new System.Drawing.Point(130, 19);
                p2.Size = new System.Drawing.Size(92, 28);
                p4.Controls.Add(p2);*/
                tb1.Location = new System.Drawing.Point(43, 47);
                tb1.Size = new System.Drawing.Size(63, 20);
                p4.Controls.Add(tb1);
                tb1.Enabled = false;
                tb3.Location = new System.Drawing.Point(108,98);
                tb3.Size = new System.Drawing.Size(86, 90);
                p4.Controls.Add(tb3);
                tb3.Enabled = false;
                canvas.Parent = this;
                theShape.Parent = canvas;
```

```
            theShape.Size = new System.Drawing.Size(92, 51);
            theShape.Location = new System.Drawing.Point(28, 28);
            theShape.CornerRadius = 5;
            theShape.BorderColor = Color.Blue;
            theShape2.Parent = canvas;
            theShape2.StartPoint = new System.Drawing.Point(64, 78);
            theShape2.EndPoint = new System.Drawing.Point(64, 98);
            theShape3.Parent = canvas;
            theShape3.Size = new System.Drawing.Size(25, 14);
            theShape3.Location = new System.Drawing.Point(49, 98);
            theShape4.Parent = canvas;
            theShape4.Size = new System.Drawing.Size(25, 14);
            theShape4.Location = new System.Drawing.Point(74, 98);
            theShape5.Parent = canvas;
            theShape5.StartPoint = new System.Drawing.Point(94, 78);
            theShape5.EndPoint = new System.Drawing.Point(94, 98);
            p4.Controls.Add(canvas);
            gogo = 1;
        }
```

else if(tb2.Text != tb1.Text) if current fact entry's heading object type different with 1st it ending object type , create an entire new object and connect to it (example: [学生]报读于[课程])

```
        {
            p2.Location = new System.Drawing.Point(130, 19);
            p2.Size = new System.Drawing.Size(92, 28);
            p4.Controls.Add(p2);
            tb1.Location = new System.Drawing.Point(43, 47);
            tb1.Size = new System.Drawing.Size(63, 20);
            p4.Controls.Add(tb1);
            tb1.Enabled = false;
            tb2.Location = new System.Drawing.Point(258, 47);
            tb2.Size = new System.Drawing.Size(63, 20);
```

```
            p4.Controls.Add(tb2);
            tb2.Enabled = false;
            tb3.Location = new System.Drawing.Point(3, 4);
            tb3.Size = new System.Drawing.Size(86, 20);
            p2.Controls.Add(tb3);
            tb3.Enabled = false;
            canvas.Parent = this;
            theShape.Parent = canvas;
            theShape.Size = new System.Drawing.Size(92, 51);
            theShape.Location = new System.Drawing.Point(28, 28);
            theShape.CornerRadius = 5;
            theShape.BorderColor = Color.Blue;
            theShape2.Parent = canvas;
            theShape2.StartPoint = new System.Drawing.Point(119, 58);
            theShape2.EndPoint = new System.Drawing.Point(150, 58);
            theShape3.Parent = canvas;
            theShape3.Size = new System.Drawing.Size(25, 14);
            theShape3.Location = new System.Drawing.Point(149, 52);
            theShape4.Parent = canvas;
            theShape4.Size = new System.Drawing.Size(25, 14);
            theShape4.Location = new System.Drawing.Point(174, 52);
            theShape5.Parent = canvas;
            theShape5.StartPoint = new System.Drawing.Point(199, 58);
            theShape5.EndPoint = new System.Drawing.Point(246, 58);
            theShape6.Parent = canvas;
            theShape6.Size = new System.Drawing.Size(92, 51);
            theShape6.Location = new System.Drawing.Point(245, 30);
            theShape6.CornerRadius = 5;
            theShape6.BorderColor = Color.Blue;
            p4.Controls.Add(canvas);
        }
    }
```

```
else if (count > 2) // ternary fact type
{
   p2.Location = new System.Drawing.Point(130, 19);
   p2.Size = new System.Drawing.Size(92, 28);
   p4.Controls.Add(p2);
   tb1.Location = new System.Drawing.Point(43, 47);
   tb1.Size = new System.Drawing.Size(63, 20);
   p4.Controls.Add(tb1);
   tb1.Enabled = false;
   tb2.Location = new System.Drawing.Point(288, 47);
   tb2.Size = new System.Drawing.Size(63, 20);
   p4.Controls.Add(tb2);
   tb2.Enabled = false;
   tb3.Location = new System.Drawing.Point(3, 4);
   tb3.Size = new System.Drawing.Size(86, 20);
   p2.Controls.Add(tb3);
   tb3.Enabled = false;
   tbb.Location = new System.Drawing.Point(155, 110);
   tbb.Size = new System.Drawing.Size(70, 20);
   p4.Controls.Add(tbb);
   tbb.Enabled = false;
   canvas.Parent = this;
   theShape.Parent = canvas;
   theShape.Size = new System.Drawing.Size(92, 51);
   theShape.Location = new System.Drawing.Point(28, 28);
   theShape.CornerRadius = 5;
   theShape.BorderColor = Color.Blue;
   theShape2.Parent = canvas;
   theShape2.StartPoint = new System.Drawing.Point(119, 58);
   theShape2.EndPoint = new System.Drawing.Point(150, 58);
   theShape3.Parent = canvas;
   theShape3.Size = new System.Drawing.Size(25, 14);
```

```
theShape3.Location = new System.Drawing.Point(149, 52);
theShape4.Parent = canvas;
theShape4.Size = new System.Drawing.Size(25, 14);
theShape4.Location = new System.Drawing.Point(174, 52);
theShape14.Parent = canvas;
theShape14.Size = new System.Drawing.Size(25, 14);
theShape14.Location = new System.Drawing.Point(199, 52);
theShape5.Parent = canvas;
theShape5.StartPoint = new System.Drawing.Point(224, 58);
theShape5.EndPoint = new System.Drawing.Point(271, 58);
theShape6.Parent = canvas;
theShape6.Size = new System.Drawing.Size(92, 51);
theShape6.Location = new System.Drawing.Point(271, 30);
theShape6.CornerRadius = 5;
theShape6.BorderColor = Color.Blue;
ls14.Parent = canvas;
ls14.StartPoint = new System.Drawing.Point(187, 66);
ls14.EndPoint = new System.Drawing.Point(187, 93);
theShape15.Parent = canvas;
theShape15.Size = new System.Drawing.Size(92, 51);
theShape15.Location = new System.Drawing.Point(143, 93);
theShape15.CornerRadius = 5;
theShape15.BorderColor = Color.Blue;
p4.Controls.Add(canvas);
if (textBox11.Text == String.Empty)
{
   string value = str.Substring(index3 + 1);
   int b = value.Length - 1;


   tb2.Text = str.Substring(index3 + 1, b);
```

```
            checkstr = str.Substring(index3 + 1, b);

            tb3.Text = str.Substring(index5 + 1, index6 - index5 - 1);
        }
        else if (textBox11.Text != String.Empty)
        {

            string value = str.Substring(index3 + 1);
            string value2 = str.Substring(index4 + 1);

            int b = value.Length - (value2.Length + 1);

            string bubu = str.Substring(str.LastIndexOf('[') + 1).TrimStart();

            int indexbubu = str.LastIndexOf('[');

            tb2.Text = bubu.Substring(0, bubu.IndexOf(']'));

            tbb.Text = str.Substring(index3 + 1, b);
            checkstr = str.Substring(index3 + 1, b);

            result = str.Substring(index5 + 1, index6 - index5 - 1);

            string result2 = str.Substring(index4 + 1, indexbubu - index4 - 1);

            tb3.Text = "..." + result + "..." + result2 + "...";

            result = result.Trim();
            textBox11.Text = textBox11.Text.Trim();
        }

    }
```

```
    testing = 1;
    richTextBox2.Text = null;
    tb3.Text = tb3.Text.Trim();
}
else if (testing == 1)
{
        String str2 = richTextBox2.Text;
        int indexa = str2.IndexOf('[');
        int index2a = str2.IndexOf(']');
        check = str2.Substring(indexa + 1, index2a - 1);

        int index3a = str2.IndexOf('[');
        index3a = str2.IndexOf('[', index3a + 1);
        int index4a = str2.IndexOf(']');
        index4a = str2.IndexOf(']', index4a + 1);

        string test = str2.Substring(index4a + 1);

        int index5a = str2.IndexOf(']');
        int index6a = str2.IndexOf('[');
        index6a = str2.IndexOf('[', index6a + 1);


        if (test == String.Empty)
        {
            string value = str2.Substring(index3a + 1);
            int b = value.Length - 1;


            tb5.Text = str2.Substring(index3a + 1, b);
            check2 = tb5.Text;
```

```
            tb6.Text = str2.Substring(index5a + 1, index6a - index5a - 1);
        }
        else if (test != String.Empty)
        {

            string value = str2.Substring(index3a + 1);
            string value2 = str2.Substring(index4a + 1);

            int b = value.Length - (value2.Length + 1);

            tb5.Text = str2.Substring(index3a + 1, b);
            check2 = tb5.Text;

            result = str2.Substring(index5a + 1, index6a - index5a - 1);

            tb6.Text = "..." + result + "..." + test;

            result = result.Trim();

        }

            if (tb1.Text == check || tb2.Text == check) // create subsequent fact
entry and check any similarity of object type that inherited from previous fact entry's
object type
            {
                if (tb1.Text == check && tb5.Text != checkstr)
                {
                    if (test3 == 0)
                    {
                        ls6.Parent = canvas;
                        ls6.StartPoint = new System.Drawing.Point(117, 78);
                        ls6.EndPoint = new System.Drawing.Point(150, 126);
```

```
                theShape8.Parent = canvas;
                theShape8.Size = new System.Drawing.Size(25, 14);
                theShape8.Location = new System.Drawing.Point(149, 120);
                theShape9.Parent = canvas;
                theShape9.Size = new System.Drawing.Size(25, 14);
                theShape9.Location = new System.Drawing.Point(174, 120);
                ls7.Parent = canvas;
                ls7.StartPoint = new System.Drawing.Point(199, 126);
                ls7.EndPoint = new System.Drawing.Point(246, 126);
                theShape10.Parent = canvas;
                theShape10.Size = new System.Drawing.Size(92, 51);
                theShape10.Location = new System.Drawing.Point(245, 98);
                theShape10.CornerRadius = 5;
                theShape10.BorderColor = Color.Blue;
                tb5.Enabled = false;
                tb5.Location = new System.Drawing.Point(257, 114);
                tb5.Size = new System.Drawing.Size(70, 20);
                p4.Controls.Add(tb5);
                tb6.Enabled = false;
                tb6.Location = new System.Drawing.Point(123, 139);
                tb6.Size = new System.Drawing.Size(100, 20);
                p4.Controls.Add(tb6);
                p4.Controls.Add(canvas);

                test3 = 1;
            }
        }
        else if (tb1.Text == check && tb5.Text == checkstr) // if 1st fact
entry's heading object type similar with 2nd fact entry's object type, thus create a line
straightaway connect from it
        {
            if (test3 == 0)
```

```
                {
                    ls6.Parent = canvas;
                    ls6.StartPoint = new System.Drawing.Point(117, 78);
                    ls6.EndPoint = new System.Drawing.Point(150, 126);
                    theShape8.Parent = canvas;
                    theShape8.Size = new System.Drawing.Size(25, 14);
                    theShape8.Location = new System.Drawing.Point(149, 120);
                    theShape9.Parent = canvas;
                    theShape9.Size = new System.Drawing.Size(25, 14);
                    theShape9.Location = new System.Drawing.Point(174, 120);
                    ls7.Parent = canvas;
                    ls7.StartPoint = new System.Drawing.Point(199, 127);
                    ls7.EndPoint = new System.Drawing.Point(244, 67);
                    tb6.Enabled = false;
                    tb6.Location = new System.Drawing.Point(123, 139);
                    tb6.Size = new System.Drawing.Size(100, 20);
                    p4.Controls.Add(tb6);
                    p4.Controls.Add(canvas);

                    test3 = 1;
                }
            }
            else if (tb2.Text == check && tb5.Text != checkstr) //if 1st fact
entry object type different with 2nd fact entry's object type, create an entire new
object
            {
                if (test3 == 0)
                {
                    ls6.Parent = canvas;
                    ls6.StartPoint = new System.Drawing.Point(277, 79);
                    ls6.EndPoint = new System.Drawing.Point(161, 119);
                    theShape8.Parent = canvas;
```

```
            theShape8.Size = new System.Drawing.Size(25, 14);
            theShape8.Location = new System.Drawing.Point(149, 120);
            theShape9.Parent = canvas;
            theShape9.Size = new System.Drawing.Size(25, 14);
            theShape9.Location = new System.Drawing.Point(174, 120);
            ls7.Parent = canvas;
            ls7.StartPoint = new System.Drawing.Point(199, 126);
            ls7.EndPoint = new System.Drawing.Point(246, 126);
            theShape10.Parent = canvas;
            theShape10.Size = new System.Drawing.Size(92, 51);
            theShape10.Location = new System.Drawing.Point(245, 98);
            theShape10.CornerRadius = 5;
            theShape10.BorderColor = Color.Blue;
            tb5.Enabled = false;
            tb5.Location = new System.Drawing.Point(257, 114);
            tb5.Size = new System.Drawing.Size(70, 20);
            p4.Controls.Add(tb5);
            tb6.Enabled = false;
            tb6.Location = new System.Drawing.Point(123, 139);
            tb6.Size = new System.Drawing.Size(100, 20);
            p4.Controls.Add(tb6);
            p4.Controls.Add(canvas);

            test3 = 1;
        }
    }
}
else if(tb1.Text != check || tb2.Text != check)
{
    tb4.Text = check;

    if(test3==0)
```

```
{
theShape7.Parent = canvas;
theShape7.Size = new System.Drawing.Size(92, 51);
theShape7.Location = new System.Drawing.Point(26, 98);
theShape7.CornerRadius = 5;
theShape7.BorderColor = Color.Blue;
tb4.Enabled = false;
tb4.Location = new System.Drawing.Point(40, 115);
tb4.Size = new System.Drawing.Size(70, 20);
p4.Controls.Add(tb4);
ls6.Parent = canvas;
ls6.StartPoint = new System.Drawing.Point(117, 128);
ls6.EndPoint = new System.Drawing.Point(148, 128);
theShape8.Parent = canvas;
theShape8.Size = new System.Drawing.Size(25, 14);
theShape8.Location = new System.Drawing.Point(149, 120);
theShape9.Parent = canvas;
theShape9.Size = new System.Drawing.Size(25, 14);
theShape9.Location = new System.Drawing.Point(174, 120);
ls7.Parent = canvas;
ls7.StartPoint = new System.Drawing.Point(199, 126);
ls7.EndPoint = new System.Drawing.Point(246, 126);
theShape10.Parent = canvas;
theShape10.Size = new System.Drawing.Size(92, 51);
theShape10.Location = new System.Drawing.Point(245, 98);
theShape10.CornerRadius = 5;
theShape10.BorderColor = Color.Blue;
tb5.Enabled = false;
tb5.Location = new System.Drawing.Point(257, 114);
tb5.Size = new System.Drawing.Size(70, 20);
p4.Controls.Add(tb5);
tb6.Enabled = false;
```

```
tb6.Location = new System.Drawing.Point(129, 89);
tb6.Size = new System.Drawing.Size(100, 20);
p4.Controls.Add(tb6);
p4.Controls.Add(canvas);
test3 = 1;
}
```

Inclusive-Or Constraint/ Exclusion/Exclusive-Or/Subset

1) To access the popup menu and check for the roles highlighted to generate the suitable constraint symbol

```
if (textBox9.Text == "Inclusive Or Constraint")//in this case, 'inclusive-or' is selected
{
  if ((blue == 1 & blue2 == 1) && (blue3 == 0 & blue4 ==0)) // same entity types (in front & at the end) for two facts
  {
      // shape object code list here
    InOrCo1 = 1;
  }
  else if ((blue == 1 & blue2 == 1) && (blue3 == 1 & blue4 == 1))// different entity type at the end for two facts
  {
   // shape object code list here
    InOrCo2 = 1;
  }
```

2) To generate dedicated verbalization constraint

```
if (InOrCo1 == 1) // in this case, will be 'inclusive-or'
{
  if (listBox7.SelectedItem.ToString() == "Option 1") // More Options verbalization – default and recommended InOrCo fact verbalization
  {
    richTextBox1.Text =
```

```
            "每一" + textBox5.Text + tb1.Text + value + "某一" +
            textBox6.Text + tb2.Text + "\r\n" +
            "或" + tb6.Text + "某一" + textBox6.Text + tb5.Text;
        }
        else if (listBox7.SelectedItem.ToString() == "Option 2")
        {
          richTextBox1.Text =
            "对于每一" + textBox5.Text + tb1.Text +",持有以下全部或至少一个条件:\n"
            +"那一"+textBox5.Text + tb1.Text+ value +
            "某一"+ textBox6.Text + tb2.Text + "\r\n" +
            "那一" + textBox5.Text + tb1.Text + tb6.Text +"某一"+ textBox6.Text + tb5.Text;
        }
```

Frequency Constraint

3) To access the popup menu and check for the roles highlighted to generate the FC icon

```
        if (textBox9.Text == "Frequency Constraint")
        {
          if (blue == 1)
          {
            ls10.Parent = canvas;
            ls10.StartPoint = new System.Drawing.Point(164, 66);
            ls10.EndPoint = new System.Drawing.Point(164, 86);
            ls10.BorderColor = Color.Purple;
            ls10.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;
            ls10.BorderWidth = 2;
            theShape12.Parent = canvas;
            theShape12.Size = new System.Drawing.Size(14, 15);
```

```
theShape12.Location = new System.Drawing.Point(157, 86);
theShape12.BorderColor = Color.Purple;
theShape12.BorderWidth = 2;
p4.Controls.Add(canvas);
FC = 1;
```

4) To call upon the FC window properties box

```
if (FC == 1)
{
   Form4 form4 = new Form4();
   form4.Show();
}
```

5) To determine the min& max frequency of the connected fact roles

```
maxF = comboBox1.Text;
minF = textBox3.Text;
if(comboBox1.Text =="unbounded")
{
   compare = ">=";
}
else if (comboBox1.Text != "unbounded")
{
   if (Convert.ToInt32(comboBox1.Text) > 1 &&
Convert.ToInt32(textBox3.Text) > 1)
   {
      compare = "...";
   }
   else if (Convert.ToInt32(comboBox1.Text) > 1 &&
Convert.ToInt32(textBox3.Text) == 1)
   {
      compare = "<=";
   }
}
```

6) To generate the compare symbol with the value in the FC constraint icon

```
if (FC == 1)
{
   lbl.Parent = canvas;
```

```
if (Form4.maxF == "unbounded")
{
   lbl.Text = Form4.compare + Form4.minF;
}
```

7) To generate the FC verbalization result

```
if (FC == 1)
{
   if (maxF != null && minF != null)
   {
      if (maxF != "unbounded")
      {
         max= Convert.ToInt32(maxF);
      }
      min = Convert.ToInt32(minF);

      if (max > 1 && min > 1) // at least min & at most max
      {
         richTextBox1.Text =
            "每一" + textBox5.Text + tb1.Text + "在'" + tb1.Text + value +
            tb2.Text + "'" + "的群体里出现" + "至少" + minF + "次和最多"
+ maxF + "次";
      }
```

Value comparison constraint

1) To access the popup menu and check for the roles highlighted to generate the VC icon

```
if (textBox9.Text == "Value Comparison Constraint")
{
   if ((blue == 1 & blue2 == 1) && (blue3 == 1 & blue4 == 1))
   {
      ls10.Parent = canvas;
      ls10.StartPoint = new System.Drawing.Point(174, 66);
```

```
ls10.EndPoint = new System.Drawing.Point(174, 86);
ls10.BorderColor = Color.Purple;
ls10.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;
ls10.BorderWidth = 2;
theShape12.Parent = canvas;
theShape12.Size = new System.Drawing.Size(14, 15);
theShape12.Location = new System.Drawing.Point(167, 86);
theShape12.BorderColor = Color.Purple;
theShape12.BorderWidth = 2;
ls11.Parent = canvas;
ls11.StartPoint = new System.Drawing.Point(174, 101);
ls11.EndPoint = new System.Drawing.Point(174, 122);
ls11.BorderColor = Color.Purple;
ls11.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;
ls11.BorderWidth = 2;
p4.Controls.Add(canvas);
VC = 1;
```

2) To call upon the VC window properties box

```
else if (VC == 1)
        {
           Form5 form5 = new Form5();
           form5.Show();
        }
```

3) To determine the type of comparison between the connected fact roles

```
if (listBox1.SelectedItem.ToString() == "<")
        {
           op = "少于"; //operator name
           compare = "<"; //operator symbol
        }
        else if (listBox1.SelectedItem.ToString() == "<=")
        {
           op = "少于或相同于";
           compare = "<=";
        }
```

```
else if (listBox1.SelectedItem.ToString() == ">")
{
   op = "大于";
   compare = ">";
}
else if (listBox1.SelectedItem.ToString() == ">=")
{
   op = "大于或相同于";
   compare = ">=";
}
```

4) To generate the compare operator in the VC constraint icon

```
if (VC == 1)
   {
        lbl.Parent = canvas;
        lbl.Text = Form5.compare;
        lbl.Location = new Point(171, 88);
        lbl.Size = new Size(50, 9);
        lbl.BackColor = Color.White;
        lbl.ForeColor = Color.Purple;
        lbl.Font = new Font(label1.Font.Name, 6, FontStyle.Bold |
FontStyle.Underline);
   }
```

5) To generate VC verbalization result

```
   if (VC == 1)
   {
        richTextBox1.Text =
         "对于每一" + textBox5.Text + tb1.Text + "," +
         tb2.Text + "是" + Form5.op + tb5.Text;
   }
```

Ring Constraint

1) To access the popup menu and check for the roles highlighted to generate the RC icon

```
if (textBox9.Text == "Ring Constraint")
{
   if (blue == 1 & blue3 ==1)
   {
      ls10.Parent = canvas;
      ls10.StartPoint = new System.Drawing.Point(64, 110);
      ls10.EndPoint = new System.Drawing.Point(74, 138);
      ls10.BorderColor = Color.Purple;
      ls10.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;
      ls10.BorderWidth = 2;
      theShape12.Parent = canvas;
      theShape12.Size = new System.Drawing.Size(14, 15);
      theShape12.Location = new System.Drawing.Point(74, 138);
      theShape12.BorderColor = Color.Purple;
      theShape12.BorderWidth = 2;
      ls11.Parent = canvas;
      ls11.StartPoint = new System.Drawing.Point(94, 110);
      ls11.EndPoint = new System.Drawing.Point(84, 138);
      ls11.BorderColor = Color.Purple;
      ls11.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;
      ls11.BorderWidth = 2;
      p4.Controls.Add(canvas);
      RC = 1;
```

2) To call upon the RC window properties box

```
else if (RC == 1)
      {
         Form6 form6 = new Form6();
```

```
            form6.Show();
        }
```

3) To determine type of ring constraint

```
for (int i = 0; i < checkedListBox1.Items.Count; i++)
        {
            if (checkedListBox1.GetItemChecked(i))
            {
                ringType =ringType+"+"+(string)checkedListBox1.Items[i];
            }
        }
        MessageBox.Show(ringType);
        count = ringType.Split('+').Length - 1;
        MessageBox.Show(count.ToString());
        if (check == false)
        {
            if (count >= 2 && count < 3) //combination of RC is selected in this case
            {
                int location = ringType.IndexOf('+');
                int location2 = ringType.IndexOf('+', location + 1);
                rT1 = ringType.Substring(1, location2 - 1);
                rT2 = ringType.Substring(location2 + 1);
                MessageBox.Show(rT1 + "\n" + rT2);
            }
            else if (count == 1)// Single RC is selected for this case
            {
                rT1 = ringType.Substring(1);
                rT2 = null;
                MessageBox.Show(rT1 + "NONE");
            }
        }
        else if (check ==true)
        {
            MessageBox.Show("rt1=" + rT1 + " rt2=" +rT2);
        }
            ringType = null;
    }

    private void checkedListBox1_SelectedIndexChanged_1(object sender,
EventArgs e)
    {

        if (checkedListBox1.SelectedIndex == 2) //to determine the auto check for
some RC when specific RC is checked by user.
        {
            rT1 = "Asymmetric";
```

```
            rT2 = null;
            checkedListBox1.SetItemChecked(2, true);
            checkedListBox1.SetItemChecked(0, true);
            checkedListBox1.SetItemCheckState(0, CheckState.Indeterminate);
            checkedListBox1.SetItemChecked(1, true);
            checkedListBox1.SetItemCheckState(1, CheckState.Indeterminate);
            check = true;
        }
```

4) To generate the RC typed of icon

```
        if (RC == 1)
        {
            string rT1 = Form6.rT1;
            string rT2 = Form6.rT2;
            if (rT1 == "Irreflexive" && rT2==null)
            {
                pictureBox3.Location = new System.Drawing.Point(74, 138);
                pictureBox3.Image                                                    =
Image.FromFile(@"C:\Users\Lim\Desktop\constraint\IrreflexiveBefore.png");
            }
```

5) To generate the RC verbalization result

```
            if (RC == 1)
            {
                if (rT1 != null & rT2 != null)
                {
                    if (rT1 == "Irreflexive" & rT2 == "Symmetric")
                    {
                        richTextBox1.Text =
                   "如果" + tb1.Text + "1 是" + tb2.Text + "2,\n 那么" +
                    tb2.Text +"2 是"+tb1.Text +"1"+textBox11.Text+"\n 无任何一"
                    + textBox5.Text + tb1.Text + "是同一" + textBox5.Text + tb1.Text +
textBox11.Text; ;
                    }
```

Auto-Language Picker

```
        if (listBox4.SelectedItem.ToString() == "Mandarin")//auto set the computer input language to Simplified Chinese
        {
            InputLanguage.CurrentInputLanguage = InputLanguage.FromCulture(new System.Globalization.CultureInfo("zh-CHS"));
        }
        else if (listBox4.SelectedItem.ToString() == "BM")//auto set the computer input language to en-US
        {
            InputLanguage.CurrentInputLanguage = InputLanguage.FromCulture(new System.Globalization.CultureInfo("en-US"));
```

Speech Narrator (*only with Verbalization Browser & NC Description Helper)

```
        SpVoice voice = new SpVoice();
        if (richTextBox1.Text != "")// to auto read out the generated verbalization output text for the dedicate facts
        {
            //SpVoice voice = new SpVoice();
            button10.Text = "Speak Now!";
            voice.Volume = 100;
            voice.Voice = voice.GetVoices(string.Empty, string.Empty).Item(6);
            voice.Speak(richTextBox1.Text, SpeechVoiceSpeakFlags.SVSFlagsAsync);
        }
        else //  prompt user with warning message for unavailability of ORM fact entry.
        {
```

MessageBox.Show("ORM verbalization is unavailable! Please enter your ORM fact first.", "Verbalization Alert", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

Advance/General search of NC along with NC Description Helper

1) Display option between general or advance search of NC

```
if (radioButton2.Checked) // if 'Advance Search' is selected
    {
      label12.Visible = true;
      checkBox4.Visible = true;
      checkBox5.Visible = true;
      checkBox6.Visible = true;
```

2) a) General search.

```
SqlCommand sqlCmd2 = new SqlCommand
   ("Select DISTINCT Class From NounClassifier3 ", hookup2);//to generate the list with all classifier for user to choose

hookup2.Open();

SqlDataReader reader2 = sqlCmd2.ExecuteReader();

if (reader2.HasRows)
{

  while (reader2.Read())
  {

    string name2 = Convert.ToString(reader2["Class"]);
    // textBox6.Text = name2;
```

```
            listBox2.Items.Add(name2);
        }
```

b) Advance Search. Display option between the type of area (alphabet only, noun type only or both) to search for generating the intended proper list

```
  if (checkBox6.Checked)// in this case, 'Both' is selected
        {
           checkBox4.Enabled = false; // 'Alphabet Only'
           checkBox5.Enabled = false;// 'NounType Only'
           checkBox4.Checked = true;
           checkBox5.Checked= true;
           label2.Visible = true;
           panel2.Visible = true;
           label13.Visible = true;
           listBox6.Visible = true;
           }

           SqlCommand sqlCmd = new SqlCommand("Select DISTINCT Noun From Noun2", hookup);// select all available nounType from the database to form the search list
           hookup.Open();
           SqlDataReader reader = sqlCmd.ExecuteReader();

           while (reader.Read())
           {

              name = Convert.ToString(reader["Noun"]);
              listBox1.Items.Add(name);
           }
```

```
SqlCommand sqlCmd = new SqlCommand("Select DISTINCT Alphabet From Noun2", hookup);// as well as select all available alphabet from database to form another serach list
hookup.Open();
SqlDataReader reader = sqlCmd.ExecuteReader();

while (reader.Read())
{

   name = Convert.ToString(reader["Alphabet"]);
   listBox6.Items.Add(name);
```

c) To generate the list of possible noun classifier based on the pick in step (2) example

```
SqlCommand sqlCmd = new SqlCommand("Select DISTINCT Class From Noun2 Where Alphabet = '" + listBox6.SelectedItem.ToString() + "' AND Noun ='" + listBox1.SelectedItem.ToString() + "'", hookup);
hookup.Open();
SqlDataReader reader = sqlCmd.ExecuteReader();

while (reader.Read())
{

   name = Convert.ToString(reader["Class"]);
   listBox2.Items.Add(name);
   strList.Add(name);
}

reader.Close();
hookup.Close();
}
```

d) To generate the Noun Type/Noun Classifier Description

```
if (checkBox1.Checked == true) // in this case, 'All' is selected
        {
           richTextBox3.Text = NounTypeDesc + "\n" + ClassifierDesc;
           checkBox2.Checked = false;
           checkBox3.Checked = false;
           mediaCheck1 = 1;
           mediaCheck2 = 0;
           mediaCheck3 = 0;
        }
```

Extra Search features (*Applied in searching for kind of nounType)

1) Auto-complete keyword field search

```
var autoCollection = new AutoCompleteStringCollection();
autoCollection.AddRange(strList.ToArray());
this.textBox7.AutoCompleteCustomSource = autoCollection;
this.textBox7.AutoCompleteMode = AutoCompleteMode.Suggest;
this.textBox7.AutoCompleteSource = AutoCompleteSource.CustomSource;
```

2) Specific keyword search

```
if (item.Contains(textBox7.Text))
                {
                   total = total + item;
                   listBox3.Items.Add(item);
                }
                searchCheck = 1;

           }
           listBox1.Items.Clear();
```

```
            for (int i = 0; i < listBox3.Items.Count; i++)
            {
                item2 = listBox3.Items[i].ToString();
                listBox1.Items.Add(item2);
            }
        }
         if(searchCheck ==0) // prompt error  message
         {
                MessageBox.Show("Search: No Matches Found",
                   "Error",MessageBoxButtons.OK,MessageBoxIcon.Exclamation,
                              MessageBoxDefaultButton.Button1);
                searchCheck = 0;
            }
```

ORM Helper Tool

1) To call upon the ORM Helper Tool

```
        if (item.Text == "Summon Robotnik.ORM")
            {
                pictureBox1.Visible = true;
                pictureBox1.Image                                                    =
Image.FromFile(@"C:\Users\Lim\Desktop\app_icon3v2.gif");
                picCheck = 1;
            }
        else if (item.Text == "Dismiss Robotnik.ORM")
            {
                pictureBox1.Visible = false;
                pictureBox1.Image = null;
                picCheck = 0;
            }
```

2) Determine the searching criteria for ORM help issue

```
Regex regExp = new Regex("fact entry|enter fact|key in fact|input fact|fact input");

foreach (Match match in regExp.Matches(questionBoxTool.Text))
{
   pictureBox1.Image = Image.FromFile(@"C:\Users\Lim\Desktop\app_iconFound.gif");
   pictureToolNotFound.Visible = false;
   searchResultBox.Visible = true;
   searchResultBox.Text = "How to input ORM Fact?\n" +
                  "[1].Pick your language options.\n" +
                  "[2].Browse to ORM 'Fact Editor' Tab bar.\n" +
                  "[3].Write your fact by putting a object type inside bracket '[ ]'.\n" +
                  "[4].Press Ctrl-Enter to generate the respective fact model to the left.";
   search = 1;
}
```

Assign the color theme for representing each verbalization keyword

```
Regex regExp = new Regex(tb2.Text); // object type

   foreach (Match match in regExp.Matches(richTextBox1.Text))
   {
      richTextBox1.Select(match.Index, match.Length);
      richTextBox1.SelectionColor = Color.Purple;
   }
   regExp = new Regex(tb1.Text); // object type
```

```
            foreach (Match match in regExp.Matches(richTextBox1.Text))
            {
               richTextBox1.Select(match.Index, match.Length);
               richTextBox1.SelectionColor = Color.Purple;
            }
            regExp = new Regex(value); //  predicate

            foreach (Match match in regExp.Matches(richTextBox1.Text))
            {
               richTextBox1.Select(match.Index, match.Length);
               richTextBox1.SelectionColor = Color.Green;
            }
            regExp = new Regex("每一|最多一|多过一|一样的|是可能的"); //
quantifier

            foreach (Match match in regExp.Matches(richTextBox1.Text))
            {
               richTextBox1.Select(match.Index, match.Length);
               richTextBox1.SelectionColor = Color.Blue;
               richTextBox1.SelectionFont = new Font(richTextBox1.SelectionFont,
FontStyle.Bold);
            }
            regExp = new Regex(textBox5.Text); //noun classifier

            foreach (Match match in regExp.Matches(richTextBox1.Text))
            {
               richTextBox1.Select(match.Index, match.Length);
               richTextBox1.SelectionColor = Color.Red;
            }
            regExp = new Regex(textBox6.Text); // noun classifier

            foreach (Match match in regExp.Matches(richTextBox1.Text))
```

```
            {
                richTextBox1.Select(match.Index, match.Length);
                richTextBox1.SelectionColor = Color.Red;
            }


        }
        if (textBox9.Text == "m:n")
        {
            string value = str.Substring(0, index);
            richTextBox1.Text = "某一" + textBox5.Text + tb1.Text + value + "至少两" +
                textBox6.Text + tb2.Text + "和对于某一" + textBox6.Text +
                tb2.Text + "，至少两" + textBox5.Text + tb1.Text + value + "那一" + textBox6.Text + tb2.Text;
        }
        if (textBox9.Text == "1:1")
        {
            string value = tb3.Text;
            richTextBox1.Text = tb1.Text + value + tb2.Text + "\r\n" +
                "每一" + textBox5.Text + tb1.Text + value + "最多一" +
                textBox6.Text + tb2.Text + "\r\n" +
                "对于每一" + textBox6.Text + tb2.Text + "," + "最多一" + textBox5.Text + tb1.Text + value
                + "那一" + textBox6.Text + tb2.Text;


            Regex regExp = new Regex(tb2.Text);

            foreach (Match match in regExp.Matches(richTextBox1.Text))
            {
```

```
            richTextBox1.Select(match.Index, match.Length);
            richTextBox1.SelectionColor = Color.Purple;
        }
        regExp = new Regex(tb1.Text);

        foreach (Match match in regExp.Matches(richTextBox1.Text))
        {
            richTextBox1.Select(match.Index, match.Length);
            richTextBox1.SelectionColor = Color.Purple;
        }
        regExp = new Regex(value);

        foreach (Match match in regExp.Matches(richTextBox1.Text))
        {
            richTextBox1.Select(match.Index, match.Length);
            richTextBox1.SelectionColor = Color.Green;
        }
        regExp = new Regex("每一|最多一|对于每一|那一");

        foreach (Match match in regExp.Matches(richTextBox1.Text))
        {
            richTextBox1.Select(match.Index, match.Length);
            richTextBox1.SelectionColor = Color.Blue;
            richTextBox1.SelectionFont = new Font(richTextBox1.SelectionFont,
FontStyle.Bold);
        }
        regExp = new Regex(textBox5.Text);

        foreach (Match match in regExp.Matches(richTextBox1.Text))
        {
            richTextBox1.Select(match.Index, match.Length);
            richTextBox1.SelectionColor = Color.Red;
```

```
        }
        regExp = new Regex(textBox6.Text);

        foreach (Match match in regExp.Matches(richTextBox1.Text))
        {
           richTextBox1.Select(match.Index, match.Length);
           richTextBox1.SelectionColor = Color.Red;
        }
      }
```

Create 100+ fact model entry (in binary only)

1. Create multiple fact entry model entry by auto assigning the identifier value and positioning the location of each object, predicate and etc.

```
 if (e.Control && e.KeyCode == Keys.Enter)
      {
        /*String str = richTextBox2.Text;
           int index = str.IndexOf('[');
           int index2 = str.IndexOf(']');
           check = str.Substring(index + 1, index2 - 1);

           int index3 = str.IndexOf('[');
           index3 = str.IndexOf('[', index3 + 1);
           string value1 = str.Substring(index3 + 1);
           int b1 = value1.Length - 1;
           string check2 = str.Substring(index3 + 1, b1);

           int index5 = str.IndexOf(']');
           int index6 = str.IndexOf('[');
           index6 = str.IndexOf('[', index6 + 1);
           string pre = str.Substring(index5 + 1, index6 - index5 - 1);

           int x = 0, y = 0, x2 = 0, y2 =0; int pa = 0, pb = 0; int xp = 0;
           int same = 0, same2 = 0;
```

```
LineShape lss = new LineShape();
    buttons[counterButton] = new RectangleShape();
    labels[counterButton] = new Label();
    if (counterButton > 51)
    {
       for (int i = 50; i <= counterButton; i++)
       {
          if (labels[i].Text == check)
          {
             if (labels[i].Text == check && labels[i].Text == check2)
             {
                checkbad = 1;
                checkbad2 = 1;
                same = 1;
                same2 = 1;
                break;
             }
             else
             {
                checkbad = 1;
                x = labels[i].Location.X;
                y = labels[i].Location.Y;
                pa = i;
                same = 1;
                continue;
             }
          }
          if (same == 0)
          {
             if (labels[i].Text != check)
             {
                checkbad = 0;
```

```
                }
            }
            if (same2 == 0)
            {
                if (labels[i].Text != check2)
                {
                    checkbad2 = 0;
                }
            }
             if(labels[i].Text == check2)
            {
                checkbad2 =1;
                pb = i;
                x2 = labels[i].Location.X;
                y2 = labels[i].Location.Y;
                same2 = 1;
                 continue;
            }
        }
    }
    same = 0; same2 = 0;
    textBox18.Text = checkbad.ToString()+ " " + checkbad2.ToString();
    if (checkbad == 1 && checkbad2 == 0)
    {
        pb = counterButton;
        buttons[counterButton].Parent = canvas;
        buttons[counterButton].Size = new System.Drawing.Size(75, 23);
        buttons[counterButton].Location = new
System.Drawing.Point(x+100, y+30);
        x2 = buttons[counterButton].Location.X;
        labels[counterButton].Size = new System.Drawing.Size(41, 23);
        labels[counterButton].Location = new Point(x + 117, y + 36);
```

```
                labels[counterButton].Text = check2;
                labels[counterButton].Tag = counterButton;
                buttons[counterButton].MouseDown += new
MouseEventHandler(ButtonClickOneEvent);
                buttons[counterButton].Tag = counterButton;
                p4.Controls.Add(canvas);
                p4.Controls.Add(labels[counterButton]);
                a = a + 1;
                counterButton = counterButton + 1;
                lss.Parent = canvas;
                lss.StartPoint = new System.Drawing.Point(x, y + 10);
                lss.EndPoint = new System.Drawing.Point(x2, y + 30);
                p4.Controls.Add(canvas);
                for (int i = 0; i <= 1; i++)
                {
                   if (i == 0)
                   {
                      ps = pa * 100 + pb;
                   }
                   if (i == 1)
                   {
                      ps = pb * 100 + pa;
                      xp = 13;
                   }
                   predicates[ps] = new RectangleShape();
                   predicates[ps].Parent = canvas;
                   predicates[ps].Size = new System.Drawing.Size(13, 10);
                   predicates[ps].Location = new System.Drawing.Point((x2 - 40) +
xp, y + 10);
                   predicates[ps].MouseDown += new
MouseEventHandler(ButtonClickOneEvent);
```

```
                predicates[ps].DoubleClick += new
EventHandler(ButtonClickTwiceEvent);
                predicates[ps].MouseUp += new
MouseEventHandler(MenuClickOneEvent);
                predicates[ps].Tag = ps;
                p4.Controls.Add(canvas);
            }
            ps = pa * 100 + pb;
            labelpredicates[ps] = new Label();
            labelpredicates[ps].Size = new System.Drawing.Size(41, 23);
            labelpredicates[ps].Location = new Point(x2 - 40, y + 28);
            labelpredicates[ps].Text = pre;
            labelpredicates[ps].Tag = ps;
            p4.Controls.Add(labelpredicates[ps]);
        }
        if (checkbad == 0 && checkbad2 == 0)
        {
            for (int i = 0; i <= 1; i++)
            {
                buttons[counterButton] = new RectangleShape();
                labels[counterButton] = new Label();
                buttons[counterButton].Parent = canvas;
                if (i == 0)
                {
                    pa = counterButton;
                    buttons[counterButton].Size = new System.Drawing.Size(75,
23);
                    buttons[counterButton].Location = new
System.Drawing.Point(60, 30 * a + 10);
                    x = buttons[counterButton].Location.X;
                    y = buttons[counterButton].Location.Y;
                    labels[counterButton].Size = new System.Drawing.Size(41, 23);
```

```
                    labels[counterButton].Location = new Point(77, (30 * a +
10 )+6);
                    labels[counterButton].Text = check;
                }
                if (i == 1)
                {
                    pb = counterButton;
                    buttons[counterButton].Size = new System.Drawing.Size(75,
23);
                    buttons[counterButton].Location = new
System.Drawing.Point(160+10 * a + 10, 30 * (a-1) + 10);
                    x2 = buttons[counterButton].Location.X;
                    labels[counterButton].Size = new System.Drawing.Size(41, 23);
                    labels[counterButton].Location = new Point((160+10 * a +
10)+17, (30 * (a-1)+ 10) + 6);
                    labels[counterButton].Text = check2;
                }
                labels[counterButton].Tag = counterButton;
                buttons[counterButton].MouseDown += new
MouseEventHandler(ButtonClickOneEvent);
                buttons[counterButton].Tag = counterButton;
                p4.Controls.Add(canvas);
                p4.Controls.Add(labels[counterButton]);
                a = a + 1;
                counterButton = counterButton + 1;
            }
            lss.Parent = canvas;
            lss.StartPoint = new System.Drawing.Point(x,y+10);
            lss.EndPoint = new System.Drawing.Point(x2,y+10);
            p4.Controls.Add(canvas);
            for (int i = 0; i <= 1; i++)
            {
```

```
                    if (i == 0)
                    {
                        ps = pa * 100 + pb;
                    }
                    if (i == 1)
                    {
                        ps = pb * 100 + pa;
                        xp=  13;
                    }
                    predicates[ps] = new RectangleShape();
                    predicates[ps].Parent = canvas;
                    predicates[ps].Size = new System.Drawing.Size(13, 10);
                    predicates[ps].Location = new System.Drawing.Point((x2 - 40)+xp,
y + 10);
                    predicates[ps].MouseDown                    +=                    new
MouseEventHandler(ButtonClickOneEvent);
                    predicates[ps].DoubleClick                  +=                    new
EventHandler(ButtonClickTwiceEvent);
                    predicates[ps].MouseUp                     +=                    new
MouseEventHandler(MenuClickOneEvent);
                    predicates[ps].Tag = ps;
                    p4.Controls.Add(canvas);
                    //textBox18.Text = ps.ToString();
                }
                ps = pa * 100 + pb;
                labelpredicates[ps] = new Label();
                labelpredicates[ps].Size = new System.Drawing.Size(41, 23);
                labelpredicates[ps].Location = new Point(x2 - 40, y + 28);
                labelpredicates[ps].Text = pre;
                labelpredicates[ps].Tag = ps;
                p4.Controls.Add(labelpredicates[ps]);
            }
```

```
                if (checkbad == 1 && checkbad2 == 1 && check == check2)
                {
                   textBox18.Text += "YESS";
                }
                if (checkbad == 1 && checkbad2 == 1 && check != check2)
                {

                   lss.Parent = canvas;
                   lss.StartPoint = new System.Drawing.Point(x+10, y + 20);
                   lss.EndPoint = new System.Drawing.Point(x+75, y + 56);
                   p4.Controls.Add(canvas);
                   LineShape lss2 = new LineShape();
                   lss2.Parent = canvas;
                   lss2.StartPoint = new System.Drawing.Point(x2 + 20, y2 + 20);
                   lss2.EndPoint = new System.Drawing.Point(x2 -10, y2 + 50);
                   p4.Controls.Add(canvas);
                   string cs = Convert.ToString(pa * 100 + pb);
                   Label lol = p4.Controls.Find("labelCS" + cs, false).FirstOrDefault()
as Label;
                   int CSXb = 0, CSXa =0, CSYb = 0, CSYa=0;
                   if (lol != null)
                   {

                      int counter = ((int)lol.Tag) + 1;
                      lol.Tag = counter;
                      lol.Text = ((pa * 100 + pb) + po).ToString();
                      CSXb = lol.Location.X;
                      CSYb = lol.Location.Y;
                      textBox18.Text += counter;
                      if (counter >= 3)
                      {
                         textBox18.Text += "WTH!!" + CSXb +" "+CSYb;
```

```
                    lol.Location = new Point(CSXb + 30, CSYb + 30);
                    CSXa = lol.Location.X;
                    CSYa = lol.Location.Y;
                    textBox18.Text +=  CSXa + " " + CSYa;
                }
                else
                {
                    lol.Location = new Point(((x2 - 40) * (counter++)), y *
(counter++) + 50);
                }
                CSXa = lol.Location.X;
                CSYa = lol.Location.Y;

                counteryes = 1;
            }
            else if (lol == null)
            {
                //textBox18.Text += "WTTTTTT!!";
                labelCounterS = new Label();
                labelCounterS.Location = new Point((x2 - 40), y + 50);
                labelCounterS.Text = ((pa * 100 + pb) + po).ToString();
                labelCounterS.Tag = 1;
                labelCounterS.Name = "labelCS" + Convert.ToString(pa * 100 +
pb);
                labelCounterS.Visible = false;
                p4.Controls.Add(labelCounterS);
                counteryes = 0;
            }
            for (int i = 0; i <= 1; i++)
            {
                if (i == 0)
                {
```

```
                    pss = pa * 100 + pb;
                }
                if (i == 1)
                {
                    pss = pb * 100 + pa;
                    xp = 13;
                }
                pss = pss + po;
                RectangleShape predicateslol = new RectangleShape();
                predicateslol.Parent = canvas;
                predicateslol.Size = new System.Drawing.Size(13, 10);
                if (counteryes == 1)
                {
                    predicateslol.Location = new System.Drawing.Point((CSXa-
60)+xp, CSYa - 50);
                }
                else
                {
                    predicateslol.Location = new System.Drawing.Point((x2 - 40) +
xp, y + 50);
                }
                // predicateslol.Click += new EventHandler(ButtonClickOneEvent);
                predicateslol.DoubleClick += new
EventHandler(ButtonClickTwiceEvent);
                predicateslol.MouseUp += new
MouseEventHandler(MenuClickOneEvent);
                predicateslol.Tag =pss;
                p4.Controls.Add(canvas);
                //textBox18.Text = predicateslol.Tag.ToString();
            }
            double pssCheck = (pa * 100 + pb) + po;
            labelpredicateslol = new Label();
```

```
            labelpredicateslol.Size = new System.Drawing.Size(41, 23);
            if (counteryes == 1)
            {
               labelpredicateslol.Location = new Point((CSXa - 30), CSYa -60);
            }
            else
            {
               labelpredicateslol.Location = new Point((x2 - 40) + 50, y + 28);
            }
            labelpredicateslol.Text = pre;
            labelpredicateslol.Name = "label" + pssCheck.ToString();
            p4.Controls.Add(labelpredicateslol);
            //textBox18.Text = labelpredicateslol.Name;
            po = po + 0.02;
            counteryes = 0;
         }
         if (checkbad2 == 1 && checkbad == 0)
         {
         }
         checkbad = 0; checkbad2 = 0;*/
      if (testing == 0)
      {
         String str = richTextBox2.Text;
         int index = str.IndexOf('[');
         int index2 = str.IndexOf(']');
         check = str.Substring(index + 1, index2 - 1);
         tb1.Text = str.Substring(index + 1, index2 - 1);

         int index3 = str.IndexOf('[');
         index3 = str.IndexOf('[', index3 + 1);
         int index4 = str.IndexOf(']');
         index4 = str.IndexOf(']', index4 + 1);
```

```
textBox11.Text = str.Substring(index4 + 1);

int index5 = str.IndexOf(']');
int index6 = str.IndexOf('[');
index6 = str.IndexOf('[', index6 + 1);
// Display a pop-up help topic to assist the user.
//textBox2.Text = "Yes";

//p3.Location = new System.Drawing.Point(17, 5);
//p3.Size = new System.Drawing.Size(339, 83);
// p4.Location = new System.Drawing.Point(25, 4);
//p4.Size = new System.Drawing.Size(313, 93);
string word = richTextBox2.Text;
int count = word.Split('[').Length - 1;
```

2. Create individual context drop down menu and check the predicate identifier value

```
RectangleShape predicates = sender as RectangleShape;
        string predicate = predicates.Tag.ToString();
        string length = predicate.Length.ToString();

        if (e.Button == MouseButtons.Right)
        {
           contextMenuStrip1.Show(Cursor.Position);
           detect = 1;

           if (predicates != null)
           {
              if (predicate.Contains("."))
```

```
{
    Label labellol = p4.Controls.Find("label" + predicate, true).First() as
Label;
    int dot = predicate.IndexOf(".");
    string p = predicate.Substring(0, dot);
    int a = Convert.ToInt32(p.Substring(0, 2));
    int b = Convert.ToInt32(p.Substring(2, 2));
    relationS = predicate;
    xS = predicates.Location.X;
    yS = predicates.Location.Y;
}
else
{
    if ((int)predicates.Tag >= 5051)
    {

        if (length == "4")
        {
            int a = Convert.ToInt32(predicate.Substring(0, 2));
            int b = Convert.ToInt32(predicate.Substring(2, 2));
            relation = a * 100 + b;
            relation2 = b * 100 + a;
        }
    }
```

3. Auto-assign the user-picked relationship value into an container with identifier for each generated fact model

```
if (relation != 0)
        {
    canvas.Shapes.Remove(lspredicates[relation]);
    int x = predicates[relation].Location.X;
    int y = predicates[relation].Location.Y;
```

```
                lspredicates[relation] = new LineShape();
                lspredicates[relation].Parent = canvas;
                lspredicates[relation].StartPoint = new System.Drawing.Point(x + 5, y
- 10);
                lspredicates[relation].EndPoint = new System.Drawing.Point(x + 29,
y - 10);
                lspredicates[relation].Tag = "m:n";
                p4.Controls.Add(canvas);
                relation = 0;
            }
            if (relationS != null)
            {
                //textBox18.Text = relationS;
                LineShape light =
canvas.Shapes.OfType<LineShape>().FirstOrDefault(ov => ov.Name == "LSZ" +
relationS);
                canvas.Shapes.Remove(light);
                LineShape lsS = new LineShape();
                lsS.Parent = canvas;
                lsS.StartPoint = new System.Drawing.Point(xS +5, yS - 10);
                lsS.EndPoint = new System.Drawing.Point(xS + 29, yS - 10);
                lsS.Name = "LSZ"+relationS;
                p4.Controls.Add(canvas);
                Label labellah = p4.Controls.Find("LS" + relationS,
true).FirstOrDefault() as Label;
                if (labellah != null)
                {
                  p4.Controls.Remove(labellah);
                }
                ll = new Label();
                ll.Name = "LS" + relationS;
                ll.Text = "m:n";
```

```
                p4.Controls.Add(ll);
                relationS = null;
            }
        }
        if (item.Text == "n:1")
        {
            InOrCoA = null;
            if (relation != 0)
            {
                canvas.Shapes.Remove(lspredicates[relation]);
                int x = predicates[relation].Location.X;
                int y = predicates[relation].Location.Y;
                lspredicates[relation] = new LineShape();
                lspredicates[relation].Parent = canvas;
                lspredicates[relation].StartPoint = new System.Drawing.Point(x + 5, y
- 10);
                lspredicates[relation].EndPoint = new System.Drawing.Point(x + 15,
y - 10);
                lspredicates[relation].Tag = "n:1";
                p4.Controls.Add(canvas);
                relation = 0;
            }
            if (relationS != null)
            {
                LineShape                    light                    =
canvas.Shapes.OfType<LineShape>().FirstOrDefault(ov => ov.Name == "LSZ" +
relationS);
                canvas.Shapes.Remove(light);
                LineShape lsS = new LineShape();
                lsS.Parent = canvas;
                lsS.StartPoint = new System.Drawing.Point(xS + 5, yS - 10);
                lsS.EndPoint = new System.Drawing.Point(xS + 15, yS - 10);
```

```
                lsS.Name = "LSZ" + relationS;
                p4.Controls.Add(canvas);
                Label labellah = p4.Controls.Find("LS" + relationS,
true).FirstOrDefault() as Label;
                if (labellah != null)
                {
                   p4.Controls.Remove(labellah);
                }
                ll = new Label();
                ll.Name = "LS" + relationS;
                ll.Text = "n:1";
                p4.Controls.Add(ll);
                relationS = null;
             }
          }*/
```

4. Auto-assign user picked noun classifier with identifier value

```
classifier[obj] = listBox2.SelectedItem.ToString();
textBox18.Text = classifier[obj];
```

5. Create verbalization for the selected fact model

```
public void ButtonClickTwiceEvent(Object sender, EventArgs e)
{
   panel3.Visible = true;
   label4.Visible = true;
   richTextBox1.Visible = true;
   button3.Visible = true;
   button4.Visible = true;
   String str = tb3.Text;
   int index = str.IndexOf('/');
   relation = 0;
```

```
        relationS = null;

        if (listBox4.SelectedItem.ToString() == "Mandarin")
        {
           RectangleShape predicates = sender as RectangleShape;
           string predicate = predicates.Tag.ToString();
           string length = predicate.Length.ToString();
           if (predicates != null)
           {
              if (predicate.Contains("."))
              {
                 Label labellol = p4.Controls.Find("label" + predicate, true).First() as
Label;
                 Label labellah = p4.Controls.Find("LS" + predicate, true).First() as
Label;
                 textBox18.Text = "TESTING"+labellah.Text;
                 int dot = predicate.IndexOf(".");
                 string p = predicate.Substring(0, dot);
                 int a = Convert.ToInt32(p.Substring(0, 2));
                 int b = Convert.ToInt32(p.Substring(2, 2));

                 if (labellah.Text == "m:n")
                 {
                    richTextBox1.Text = "某一" + classifier[a] + labels[a].Text +
labellol.Text + "至少两" +
                       classifier[b] + labels[b].Text + "和对于某一" + classifier[b] +
                       labels[b].Text + "，至少两" + classifier[a] + labels[a].Text +
labellol.Text + "那一" + classifier[b] + labels[b].Text;
                 }
                 if (labellah.Text == "n:1")
                 {
```

```
                richTextBox1.Text = labels[a].Text + labellol.Text + labels[b].Text
+ "\r\n" +
                "每一" + classifier[a] + labels[a].Text + labellol.Text + "最多一" +
                 classifier[b] + labels[b].Text + "\r\n" +
                 "多过一" + classifier[a] + labels[a].Text
                 + labellol.Text + "一样的" + labels[b].Text + "是可能的";
              }
            }
            else
            {
              if ((int)predicates.Tag >= 5051)
              {
                if (length == "4")
                {
                  int a = Convert.ToInt32(predicate.Substring(0, 2));
                  int b = Convert.ToInt32(predicate.Substring(2, 2));
                  int total = a * 100 + b;
                  string sum = classifier[a] + labels[a].Text +
labelpredicates[total].Text + classifier[b] + labels[b].Text;
                  string p =null;
                  if (p!="Inclusive Or Constraint")
                  {
                     p= lspredicates[total].Tag.ToString();
                    textBox18.Text = p;
                  }

                  if (p == "m:n")
                  {
                    richTextBox1.Text = "某一" + classifier[a] + labels[a].Text +
labelpredicates[total].Text + "至少两" +
                      classifier[b] + labels[b].Text + "和对于某一" + classifier[b]
+
```

```
                    labels[b].Text + "，至少两" + classifier[a] + labels[a].Text
+ labelpredicates[total].Text + "那一" + classifier[b] + labels[b].Text;
                }
                if(p == "n:1")
                {
                  richTextBox1.Text = labels[a].Text +
labelpredicates[total].Text + labels[b].Text + "\r\n" +
                  " 每 一 " + classifier[a] + labels[a].Text +
labelpredicates[total].Text + "最多一" +
                   classifier[b] + labels[b].Text + "\r\n" +
                   "多过一" + classifier[a] + labels[a].Text
                   + labelpredicates[total].Text + "一样的" + labels[b].Text + "
是可能的";
                }
                if (InOrCoA == "Inclusive Or Constraint")
                {
                  MessageBox.Show(blueA + " " + blueB);
                  int a1 = Convert.ToInt32(blueA.Substring(0, 2));
                  int b1 = Convert.ToInt32(blueA.Substring(2, 2));
                  int total1 = a1 * 100 + b1;
                  int a2 = Convert.ToInt32(blueB.Substring(0, 2));
                  int b2 = Convert.ToInt32(blueB.Substring(2, 2));
                  int total2 = a2 * 100 + b2;


                  richTextBox1.Text= "每一" + classifier[a1] + labels[a1].Text +
labelpredicates[total1].Text + "某一" +
                  classifier[b1] + labels[b1].Text + "\r\n" +
                  "或" + labelpredicates[total2].Text + "某一" + classifier[b2] +
labels[b2].Text;
```

Subtyping

1. **Input fact to create an object type acting as the supertype or any subsequent object type that acts as a subtype**

```
int count = word.Split('[').Length - 1;
if (count ==1 )//if counted the input text value to be 1
{
  if (subtypeIncrementCount == 0)// check whether it was the initial object type
      {
 //respected model code over here
}
    subtypeIncrementCount++;
        }
```

2. **To check the direction of linking between instance/subtype object and its supertype**

```
theShape7_Click(object sender, EventArgs e)
{
   if (checkBox7.Checked)// to check whether the 'subtype connector' is checked
   {
      if (subtype == 1)
      {
         subtype2 = 2;
      }
      else
      {
         subtype2 = 1;
      }
      if (subtype2 == 2) //in this case, if this object act as the instance
      {
         pictureBox3.Size = new Size(17, 29);
         pictureBox3.Location = new System.Drawing.Point(74, 80);
```

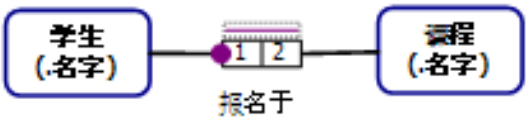
```
                pictureBox3.Image =
Image.FromFile(@"C:\Users\Lim\Desktop\constraint\subtypedown.png");
                }
            }
            ST = 1;
        }
```

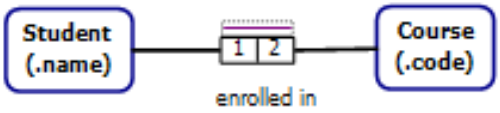
3. To verbalize the subtyping according to the direction of connection of instance/subtype object and supertype object

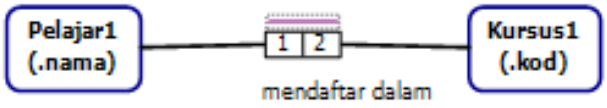
```
    if (ST == 1)
            {
                if(subtype == 2)
                {
                    richTextBox1.Text = "每一" + check5 + tb4.Text + "是" +
tb1.Text + "的类型。";
                }
                else if (subtype2 == 2)
                {
                    richTextBox1.Text = "每一" + textBox5.Text + tb1.Text + "是
" + tb4.Text + "的类型。";
                }
            }
```

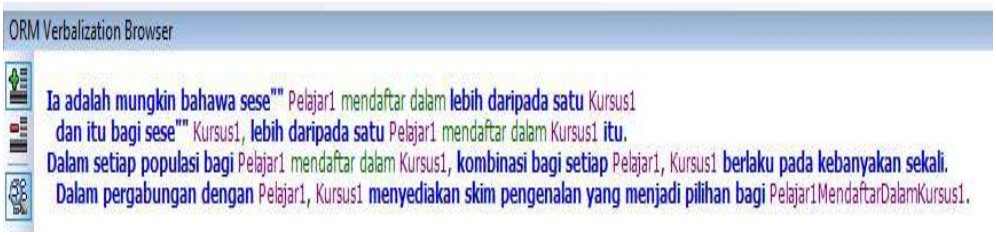
4.3 NORMA Extension: BM and Mandarin Verbalization

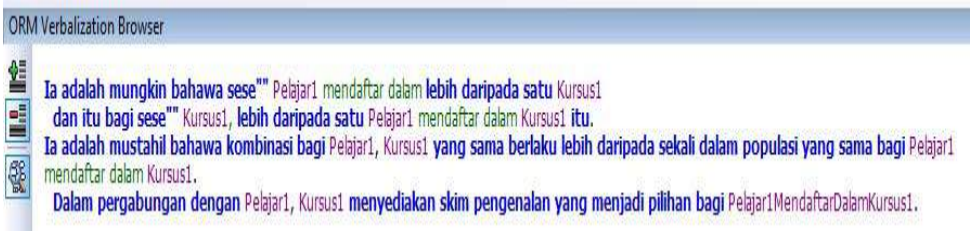
In this section, we illustrate via screenshots some of our extensions to the NORMA tool to support ORM model entry and verbalization both BM and Mandarin. At this stage, support for noun classifers has yet to be added. As NORMA's internal verbalization framework is to be replaced next year by a new tree-based verbalization framework, there is little motivation to provide a full NORMA implementation of our verbalization work, so we confine ourselves to a basic outline of the extensions required for the current framework.






Pelajar1
(.nama)
1
2
Kursus1
(.kod)
mendaftar dalam


ORM Verbalization Browser
Ia adalah mungkin bahawa sese"" Pelajar1 mendaftar dalam lebih daripada satu Kursus1
dan itu bagi sese"" Kursus1, lebih daripada satu Pelajar1 mendaftar dalam Kursus1 itu.
Dalam setiap populasi bagi Pelajar1 mendaftar dalam Kursus1, kombinasi bagi setiap Pelajar1, Kursus1 berlaku pada kebanyakan sekali.
Dalam pergabungan dengan Pelajar1, Kursus1 menyediakan skim pengenalan yang menjadi pilihan bagi Pelajar1MendaftarDalamKursus1.


ORM Verbalization Browser
Ia adalah mungkin bahawa sese"" Pelajar1 mendaftar dalam lebih daripada satu Kursus1
dan itu bagi sese"" Kursus1, lebih daripada satu Pelajar1 mendaftar dalam Kursus1 itu.
Ia adalah mustahil bahawa kombinasi bagi Pelajar1, Kursus1 yang sama berlaku lebih daripada sekali dalam populasi yang sama bagi Pelajar1 mendaftar dalam Kursus1.
Dalam pergabungan dengan Pelajar1, Kursus1 menyediakan skim pengenalan yang menjadi pilihan bagi Pelajar1MendaftarDalamKursus1.

4.4 Snippets for Noun Classifier Support Plugin (Mandarin and BM)

In this section, we briefly discuss the current, inner verbalization framework engine of NORMA for the core verbalization snippet files. A snippet is basically used to define a fixed text element that records with each identified quantifier a tag with a numbered address field (such as {0},{1}) to cater for ordering of object type combinations used in the generated verbalization. We identify possible snippets that are needed to plugin support for noun classifers in both BM and Mandarin. In both BM and Mandarin, a noun classifier is combined with the noun phrase for an object type to signify the kind of object that is being quantified. Hence, an extra snippet needs to be introduced to allow noun classifiers to be added, as discussed in the verbalization patterns covered in chapter 2. Table 4.3 provides a simple example of a component of a frequency constraint verbalization.

Table 4.3 Example verbalization component in English, Mandarin and BM

English	Mandarin	BM
More than 3 instances of Car	多过 3 辆车	Lebih banyak daripada 3buah Kereta

Instance of Car ⇔Cars,
Green→ denoted as Noun Classifier (NC) that inputted by user.

<Snippet type="NounClassifier"><![CDATA[{1},{0}]]></Snippet>
{1} →noun classifier {0}→ object type

4.4.1 Snippets to support noun classifiers in Mandarin

In Mandarin only, the use of the universal quantifier before a combination of two or more object types always uses '个'as the noun classifier, regardless of the kind of object. So for this case we need an extra snippet to hardwire in this choice, as follows. Here {0} is a replacement field to be filled in by the object type combination.

<Snippet type="UniversalQuantifierCombination" sign="positive">
<![CDATA[每一个{0}]]></Snippet>

The negative verbalization of this case uses the “the same” before a combination of object types, and this also requires ‘个’, regardless of the kind of object. So for this case we need an extra snippet to hardwire in this choice, as follows:

```
<Snippet type="IdentityReferenceQuantifierCombination">
<![CDATA[<span class="quantifier">同一个</span>{0}]]></Snippet>
```

For the case where a single object type is used (not a combination), the noun classifier varies according to the kind of object, so will be input via a new property which allows the user to choose the desired noun classier (of which there are about 150 possible choices). We use a replacement field to capture this choice of noun classifier. We reserve the snippets named “UniversalQuantifier” and ="IdentityReferenceQuantifier" for the single object type case only, i.e.

```
<Snippet type="UniversalQuantifier" sign="positive">
<![CDATA[<span class="quantifier">每一</span>{0}]]></Snippet>

<Snippet type="IdentityReferenceQuantifier">
<![CDATA[<span class="quantifier">同一</span>{0}]]></Snippet>
```

Here are the other relevant snippets:

```
<Snippet type="UniversalQuantifier" sign="negative">
<![CDATA[<span class="quantifier">某一</span>{0}]]></Snippet>
<Snippet type="AtMostOneQuantifier">
<![CDATA[<span class="quantifier">最多一</span>{0}]]></Snippet>
<Snippet type="ForEachCompactQuantifier">
<![CDATA[<span class="quantifier">对于每一</span>{0},{1}]]></Snippet>
<Snippet type="ForEachNegatableCompactQuantifier" sign="positive">
<![CDATA[<span class="quantifier">对于每一</span{0},{1}]]></Snippet>
<Snippet type="ForEachNegatableCompactQuantifier" sign="negative">
<![CDATA[<span class="quantifier">对于某一</span>{0},{1}]]></Snippet>
<Snippet type="ForEachIndentedQuantifier">
<![CDATA[<span class="quantifier">对于每一</span>{0},<br/><span
class="smallIndent">{1}</span>]]></Snippet>
<Snippet type="ForEachNegatableIndentedQuantifier">
<![CDATA[<span class="quantifier">对于每一</span> {0},<br/><span
class="smallIndent">{1}</span>]]></Snippet>
<Snippet type="ForEachNegatableIndentedQuantifier" sign="negative">
<![CDATA[<span class="quantifier">对于某一</span>{0},<br/><span
class="smallIndent">{1}</span>]]></Snippet>
```

<Snippet type="ExistentialQuantifier" sign="positive">
<![CDATA[某一{0}]]></Snippet>
<Snippet type="ExistentialQuantifier" sign="negative">
<![CDATA[无任何一{0}]]></Snippet>
<Snippet type="ExactlyOneQuantifier">
<![CDATA[仅有一{0}]]></Snippet>
<Snippet type="MoreThanOneQuantifier">
<![CDATA[多过一{0}]]></Snippet>
<Snippet type="OneQuantifier" sign="positive">
<![CDATA[最多一 {0}]]></Snippet>
<Snippet type="OneQuantifier" sign="negative">
<![CDATA[多过一 {0}]]></Snippet>
<Snippet type="FullSubtypeDerivation">
<![CDATA[*每一{0}是{1}]]></Snippet>
<Snippet type="NegativeReadingForUnaryOnlyDisjunctiveMandatory"
sign="negative">
<![CDATA[某一 {0} 无参与任何下列一项:{1}]]></Snippet>
<Snippet type="SubtypeMetaReading" sign="positive">
<![CDATA[每一{0}在<a
class="predicateText" href="elementid:{2}">的类型中是{1}种实
例]]></Snippet>
<Snippet type="SubtypeMetaReading" sign="negative">
<![CDATA[某一{0}在<a
class="predicateText" href="elementid:{2}">的类型中不是{1}种实
例]]></Snippet>
<Snippet type="ExclusionBinaryLeadReading">
<![CDATA[无任何一 {0}同一{1}]]></Snippet>
<Snippet type="FrequencyPopulation">
<![CDATA[每一{0}在“0}ip 的总数据里出现
{2}]]></Snippet>
<Snippet type="FrequencyRangeExact">
<![CDATA[仅有{0}]]></Snippet>
<Snippet type="FrequencyRangeMinAndMax">
<![CDATA[至少{0}和最多{1}]]></Snippet>
<Snippet type="FrequencyRangeMaxUnbounded">
<![CDATA[至少{0}]]></Snippet>
<Snippet type="FrequencyRangeMinUnbounded">
<![CDATA[最多{1}]]></Snippet>
<Snippet type="ExactlyOneTypedOccurrence">
<![CDATA[仅有一 {0}]]></Snippet>
<Snippet type="AtMostOneTypedOccurrence">

<![CDATA[最多一 {0}]]></Snippet>
<Snippet type="Acyclicity">
<![CDATA[无任何一{0} 可以通过更多的穿越循环回到自己当{1}]]></Snippet>
<Snippet type="AcyclicityWithRoleNumbers">
<![CDATA[无任何一{0}可以通过一或更多的角色组合穿越循环回到自
己:
roles
{2}和{3}的{1}]]></Snippet>
<Snippet type="AggregateParameterSimpleAggregationContext">
<![CDATA[{0} 那一]]></Snippet>
<Snippet type="AggregateBagProjection">
<![CDATA[每一 {0}]]></Snippet>

4.4.2 Snippets to support noun classifiers in Bahasa Melayu

<Snippet type="UniversalQuantifier" sign="negative">
<![CDATA[sese {0}]]></Snippet>
<Snippet type="UniversalQuantifier" sign="negative">
<![CDATA[sese {0}]]></Snippet>
<Snippet type="AtMostOneQuantifier">
<![CDATA[paling banyak satu
{0}]]></Snippet>
<Snippet type="ForEachNegatableCompactQuantifier" sign="negative">
<![CDATA[bagi sese {0},
<Snippet type="ForEachNegatableIndentedQuantifier" sign="negative">
<![CDATA[bagi sese {0},
{1}]]></Snippet>
<Snippet type="ExistentialQuantifier" sign="positive">
<![CDATA[sese {0}]]></Snippet>
<Snippet type="ExactlyOneQuantifier">
<![CDATA[betul-betul satu {0}]]></Snippet>
<Snippet type="OneQuantifier" sign="positive">
<![CDATA[paling banyak satu
{0}]]></Snippet>
<Snippet type="OneQuantifier" sign="negative">
<![CDATA[lebih daripada satu
{0}]]></Snippet>
<Snippet type="NegativeReadingForUnaryOnlyDisjunctiveMandatory"
sign="negative">
<![CDATA[sese {0} tiada terlibat mana-mana satu di bawah
berikut:{1}]]></Snippet>
<Snippet type="FrequencyRangeExact">

<![CDATA[hanya{0}]]></Snippet>
<Snippet type="FrequencyRangeMinAndMax">
<![CDATA[sekurang-kurangnya {0} dan
paling banyak {1}]]></Snippet>
<Snippet type="FrequencyRangeMaxUnbounded">
<![CDATA[sekurang-kurangnya {0}]]></Snippet>
<Snippet type="FrequencyRangeMinUnbounded">
<![CDATA[paling banyak {1}]]></Snippet>
<Snippet type="ExactlyOneTypedOccurrence">
<![CDATA[hanya satu {0}]]></Snippet>
<Snippet type="AtMostOneTypedOccurrence">
<![CDATA[paling banyak satu
{0}]]></Snippet>

5 Conclusion

5.1 Introduction

This chapter concludes by summarizing our work toward automating the ORM models verbalization in both Bahasa Malaysia (BM) and Mandarin, noting the main empirical and theoretical results, and suggesting some possibilities for future enhancements.

This research is mainly focused on verbalizing conceptual data model requirements in BM and Mandarin. We have implemented a way to enter graphical models in the Object-Role Modeling (ORM) notation in BM and Mandarin text as well as to automatically generate verbalization of models, including fact types and constraints in these languages. In this regard, we specified the relevant verbalization patterns, and programmed the required transformations in our autonomous prototype, as well as adding some extensions to the current NORMA verbalization framework.

In the specification stage, we classified all the varieties of ORM constraints and fact types that are supported in the ORM graphical notation in both BM and Mandarin in order to convert them into the relevant logical and linguistic verbalization patterns that conform to the underlying grammars of these two languages. After implementing the transformations, we conducted empirical evaluations to confirm the unambiguous and correct rendering of our verbalization output.

5.2 Procedure and Empirical Findings

Initially, we outlined our main objective and purpose concerning automatic verbalization of ORM models in BM and Mandarin. Following that, we researched the literature realted to our research topic to determine what had already been done in this regard, and to ensure that our work would make a significant contribution to the state of the art in automating verbalization of ORM models in the two chosen languages.

During the design stage, we constructed formal patterns and metamodels for capturing the relevant structures in logical and linguistic form in BM and Mandarin. We first transformed the ORM constraints to an underlying logical form, and then transformed the logical form to linguistic forms in the two chosen natural languages. We identified and compared the critical differences between English, BM and Mandarin regarding their major verbalization patterns incorporating logical elements (e.g modal operators and quantifiers) to cover all categories of ORM constraints (internal and external uniqueness constraints, simple and disjunctive mandatory role constraints, frequency constraints, subset, equality and exclusion constraints, object types and role cardinaility constraints, ring constraints, subtyping etc.).

To cater for differences of BM and Mandarin grammar compared with English, special attention was made to properly deal with the treatment of various issues (e.g. pronouns, singular vs plural, prepositions, and active vs passive sentences) as well as to cater for different kinds of noun classifiers used for different kinds of noun. To facilitate the choice of appropriate noun classifiers, we designed a user interface for noun classifier pickup in both BM and Mandarin.

Based on our specficiations, we implemented (mainbly by programming in C#) a prototype to support entry of ORM models, and automatic verbalization of these models (covering all ORM constraint categories) in BM and Mandarin. Additionally, we discussed extensions to the current verbalization framework provided by the NORMA tool which is coded in C#, XML and XSLT.

In our prototype, we included a language picker to assist with entering text in Mandarin and BM, adopting some entirely new ways to enter ORM fact types and constraints, connecting each object type to a noun classifier window which supports general and advanced search for NounPhase, NounType, and NounPhrase Description and Classifier as well as a speech narrator to automate the vocal rendition of the verbalization result.

The verbalization output for a wide variety of ORM constraints was checked with native speakers of BM and Mandarin who found our verbalizations to be both unambiguous and correct. In addition, comparisons with BM and Mandarin verbalizations generated by the Google Translate tool when input with the English

verbalizations generated by the NORMA tool indicated that verbalizations were typically preferred bynative speakers to those generated by Google Translate.

5.3 Contributions and Theoretical Findings

In this section we note some inadequacies in earlier efforts relevant to our research topic, and indicate some ways in which our work extends the state of the art. To our knowledge, no other researchers have yet tried to automate the verbalization of ORM models in Bahasa Melayu.

Some very basic work was done by Anna Choi and others to specify verbalization of ORM constraints in Mandarin, but this covered only a fraction of ORM consraints and not all of these were specified correctly, and the work was not always implemented. Moreover, no formalization of the transformations was provided (e.g. no translation to logical form). In contrast, our work provided a formalized, complete and correct coverage as well a full implementation.

Additionally, we catered in detail for specific features of Mandarin and BM such as noun classifiers and special use of pronouns. To cater for noun classifiers, we first developed a metamodel, then used that to design a user interface to support and simplify the choice of appropprriet noun classifiers.

5.4 Limitations of the Study

While our independently developed prototype supports entry of ORM models and full coverage of automated verbalization of ORM constraints in BM and Mandarin, there are some shortcomings in the second system we built as an extension to the NORMA tool. This second system reached only the first phase to identify the snippet files used for nounc claiisifer plugin and define files to store for user selected classifier. As the current NORMA verbalization framework will be replaced next year by a tree-based system, there was little motivation to provide a complete version of this second system.

Secondly, for the Chinese verbalization in our protoype, currently we support only the simplified Chinese version and not the traditional Chinese version.

Thirdly, the speech narrator in our prototype is restricted to Mandarin only, as the BM language Text-To-Speech (TTS) voice media file is currently not available online to download to plug in into my code.

5.5 Future Enhancements

Once NORMA's new verbalization framework becomeavailoable, we would like like to implement our approach via language extensions to the NORMA tool to fully covering all of the many ORM graphical constraint varieties in both BM and Mandarin languages.

The Chinese verbalization we currently adopted in our system is in Simplified Chinese characters which are commonly used by mainland China, Malaysia and Singapore. In the future, we hope to also have a version with Traditional Chinese (commonly used by Taiwan, Hong Kong, Macau etc) in the next generation of our verbalization tool to further broaden the application of ORM modeling among Chinese communities from all over the world.

Moreover, for the noun classifier properties window that we built into our prototype system, in future when a user attempts to access any object type (noun phase) we would like to straightaway auto-recommend the potential classifier to be use as the most convenient choice. Additionally, we would like to better categorize the noun classifier subclasses (nountype and classifier).

We also hope to fully functionalize our little assistant tools (ORM Robotnik 'Little helper') to also connect the user online to access to real-time potential resources regarding ORM verbalization and notation/constraint guidelines, question and answer topics, and from time to time prompt the user with the most recent updated on related modeling websites.

Lastly, the speeh narrator that we built into our protope system currently supports Mandarin only. In a futire version, we would like to extend this text-tospeech facility to also support auto vocalization in Bahasa Melayu.

References

1. Attempto project (Attempto Controlled English) Website: http://attempto.ifi.uzh.ch/site/.
2. Bakema, G., Zwart, J., van der Lek, H.: *Fully Communication Oriented Information Modelling*. Ten Hagen Stam (2000)
3. Bloesch, A., Halpin, T.: Conceptual queries using ConQuer-II. In: *Proc. ER'97: 16th Int. Conf. on conceptual modeling*, LNCS, vol. 1331, pp. 113-126. Springer, Heidelberg (1997)
4. Chen, P. P.: The entity-relationship model—towards a unified view of data. *ACM Transactions on Database Systems* 1(1), 9–36 (1976), http://csc.lsu.edu/news/erd.pdf.
5. Choi, A.: Extending Object Role Modelling Language to cater for Chinese Language, Technical Report, Department of Computer Science, University of Queenland (1994)
6. Curland, M., Halpin, T.: The NORMA Software Tool for ORM 2. In: Soffer, P., Proper, E. (eds.), *CAiSE Forum 2010*, LNBIP 72, pp. 190-204, Springer-Verlag Berlin Heidelberg (2010)
7. Curland, M., Halpin, T.: Enhanced Verbalization of ORM Models. In: Herrero, P. et al. (eds.), *On the Move to Meaningful Internet Systems 2012: OTM 2012 Workshops*, LNCS, vol. 7567, pp. 399-408. Springer, Heidelberg (2012)
8. Halpin, T.: A Logical Analysis of Information Systems: static aspects of the data-oriented perspective. Doctoral dissertation, University of Queensland (1989)
9. Halpin, T.: Business Rule Verbalization. In: Doroshenko, A. et al. (eds.), *Information Systems Technology and its Applications: Proc. ISTA-2004*, Lec. Notes in Informatics, vol. P-48, pp. 39-52 (2004)
10. Halpin, T.: ORM 2. In: Meersman, R., Tari, Z. et al. (eds.) *OTM 2005 Workshops*, LNCS, vol. 3762, pp. 676-87. Springer, Heidelberg (2005)
11. Halpin, T.: Modality of Business Rules. In: Siau, K. (ed.) *Research Issues in Systems Analysis and Design, Databases and Software Development*, pp. 206-226. IGI Publishing, Hershey (2007)
12. Halpin, T.: Object-Role Modeling: Principles and Benefits. International *Journal of Information Systems Modeling and Design* 1(1), 32-54 (2010)
13. Halpin, T.: Fact-Orientation and Conceptual Logic. *Proc. 15th International EDOC Conference*, IEEE Computer Society, Helsinki, pp. 14-19 (2011)
14. Halpin, T.: Formalization of ORM Revisited. In: P. Herrero et al. (eds), *OTM 2012 Workshops*, LNCS vol. 7567, pp. 348-357. Springer, Heidelberg (2012)
15. Halpin, T.: *Object-Role Modeling Fundamentals*, Technics Publications, New Jersey (2015).
16. Halpin, T., Curland, M.: Automated Verbalization for ORM 2. In: Meersman, R., Tari, Z., Herrero, P. et al. (eds.) *OTM 2006 Workshops*. LNCS, vol. 4278, pp. 1181-1190. Springer, Heidelberg (2006)

17. Halpin, T., Curland, M.: Enriched Support for Ring Constraints. In: Meersman, R., Dillon, T., Herrero, P. (eds.) *OTM 2011 Workshops*. LNCS, vol. 7046, pp. 309-318, Springer, Heidelberg (2011)
18. Halpin, T., Morgan, T.: *Information Modeling and Relational Databases, 2nd edn.*, Morgan Kaufmann, San Francisco (2008)
19. Halpin, T., Wijbenga, J.: FORML 2. In: Bider, I. et al. (eds), *Enterprise, Business-Process and Information Systems Modeling*, LNBIP, vol. 50, pp. 247–260. Springer, Heidelberg (2010)
20. Heath, C.: The Constellation Query Language. In: Meersman, R. et al. (eds.), *OTM 2009 Workshops*, LNCS, vol. 5872, pp. 682-69. Springer, Heidelberg (2009)
21. ter Hofstede, A., Proper, H., van der Weide, T.: Formal definition of a conceptual language for the description and manipulation of information models. *Information Systems*, 18(7), 489-523 (1993)
22. Jarrar, M., Keet, C. M., Dogilli, P.: Multilingual verbalization of ORM conceptual models and axiomatized ontologies, Technical Report, Vrije Universiteit Brussel, Brussels (2006)
23. Kuhn, T.: A Survey and Classification of Controlled Natural Languages, *Computational Linguistics*, MIT Press (to appear).
24. Lim, S. & Halpin, T.: Verbalizing ORM Models in Malay and Mandarin. In: Demey, Y. & Panetto, F. (eds.), *On the Move to Meaningful Internet Systems 2013: OTM 2013 Workshops*, LNCS, vol. 8186, pp. 545-554. Springer, Heidelebrg (2013)
25. Meersman, R.: The RIDL Conceptual Language. Int. Centre for Information Analysis Services, Control Data Belgium (1982)
26. Nijssen, M., Lemmens, I.: Verbalization for Business rules and Two Flavors of Verbalization for Fact Examples. In: Meersman, R. et al. (eds.), *OTM 2008 Workshops*, LNCS, vol. 5333, pp. 760-769. Springer, Heidelberg (2008)
27. Object Management Group: OMG Unified Modeling Language (OMG UML), version 2.5 FTF Beta 1, http://www.omg.org/spec/UML/2.5 (2012)
28. Pool, J. 2006, 'Can Controlled Languages Scale to the Web?', *Proc. CLAW2006: 5th International Workshop on Controlled Language Applications*. URL: http://utilika.org/pubs/etc/ambigcl/clweb.html.
29. Schwitter, R. 2007, 'PENG (Processable English)'. URL: http://web.science.mq.edu.au/~rolfs/peng/.
30. Schwitter, R.: Controlled Natural Languages Website: http://sites.google.com/site/controllednaturallanguage/.
31. Sowa, J.: Common Logic Controlled English. URL: http://www.jfsowa.com/clce/specs.htm (2004)
32. Warmer, J. & Kleppe, A.: The Object Constraint Language, Second Edition. Addison-Wesley (2003)
33. 中文量词大全 http://xh.5156edu.com/page/z7949m2560j18586.html

34. Kata Bilangan dan Penjodoh Bilangan
http://mbsskl.edu.my/panitia_bm/files/2009/02/mengenal-kata.pdf
35. Translating “who” / “that” into *Chinese* - East Asia Student (2011). Retrieved November 11, 2011 from http://eastasiastudent.net/china/mandarin/who-that
36. What Are *Relative Pronouns*? (grammar lesson)
http://www.grammar-monster.com/glossary/relative_pronouns.htm
http://en.wikipedia.org/wiki/That
37. Key Differences Between Chinese and English: Plural Usage (2014). Retrieved February 25, 2014.
https://www.aje.com/en/education/other-resources/articles/key-differences-between-chinese-and-english-plural-usage
38. Vowel Sounds - When to use *A and An*, Learn English (2013). Retrieved August 2, 2013, from http://news.ninemsn.com.au/health/story_13178.asp
39. Imbuhan Kata Kerja (Bahasa Melayu) (2012, Retrieved July 6, 2012, from
40. http://ms.wikipedia.org/wiki/Imbuhan_kata_kerja_(bahasa_Melayu)
41. Bab 4 Penjodoh Bilangan Bahasa Melayu http://studentsrepo.um.edu.my/343/7/BAB4.pdf
42. Apakah Maksud: Kata Nama Am Konkrit dan Kata Nama Am
http://5perenggan.blogspot.com/2014/05/apakah-maksud-kata-nama-konkrit-dan.html
43. Penjodoh Bilangan http://mrafain.blogspot.com/p/sistem-bahasa_23.html