Harmonization Document based on Reference Ontology

Ontology: Object-Oriented Code Ontology (OOC-O)

1. Introduction

This document presents a harmonization to the elements of selected OO programming languages, applying equivalence relations with concepts of OOC-O. The document is organized as follows: Section 2 contains the harmonization purpose and its intended uses; Section 3 presents the matches of the programming languages related to OOC-O.

2. Purpose and Intended Uses of the Harmonization

The harmonization between OO programming languages and Object-Oriented Code Ontology (OOC-O)[1] aims to represent the equivalence between semantic of concepts. Each language adopts different syntax and semantics for their constructs, resulting in different levels in which OO principles are addressed. In this context, OOC-O can be used to support interoperability between them.

Considering the range of existing languages, we selected languages that provide constructs for the basic OO principles discussed above in order to form the baseline of our research, namely: Smalltalk, Eiel, C++, Java and Python. The selection took into account the first two OO programming languages ever proposed and the three currently most popular OO languages according to the TIOBE[2], IEEE Spectrum[3] and Redmonk[4] indexes.

3. Harmonization

This section introduces the Harmonization of the OO programming languages using the Object-Oriented Code Ontology (OOC-O). The subsequent sections present the harmonization between OO Programming Languages and OOC-O (Section 3.1) and the harmonization between OOC-O and OO Programming Languages (Section 3.2).

3.1. OO Programming Languages and OOC-O

The subsections presents the harmonization for each programming language related to OOC-O.

[1] https://nemo.inf.ufes.br/projects/sfwon
[2] tiobe.com, January 2019.
[3] spectrum.ieee.org/at-work/innovation/the-2018-top-programming-languages, July 2018.
[4] redmonk.com/sogrady/2019/03/20/language-rankings-1-19/, January 2019.

3.1.1. Smalltalk Language

Table 1 shows the matches between the Smalltalk language and the OOC-O.

Table 1 Equivalence between Smalltalk Language and OOC-O.

Language Concept	**OOC- concept**
Abstract Class	Abstract Class
Abstract Method	Abstract Method
Accessor Method	Accessor Method
Access	Element Visibility
Argument	Parameter Variable
Class	Concrete Class & Extendable Class
Class Method	Class Method
Category	Logical Module
Instance Method	Instance Method
Instance Variable	Instance Variable
Method	Instance Method & Overridable Method
Name	Name
Namespace	Logical Module
Root Class	Root Class
Shared Variable	Class Variable
Subclass	Subclass
Superclass	Superclass
Template	Generic Class & Generic Method
Temporary Variable	Local Variable
Type	Type
Type Parameter	Type Parameter
Variable	Variable

3.1.2. Eiffel Language

Table 2 shows the matches between the Eiffel language and the OOC-O.

Table 2 Equivalence between Eiffel Language and OOC-O.

Language Concept	**OOC- concept**
Accessor Routine	Instance Method
Access	Element Visibility
Argument	Parameter Variable
Attribute	Instance Variable
Block	Block
Class	Concrete Class & Extendable Class
Cluster	Logical Module
Constante Attribute	Mutability
Creation Procedure	Constructor Method
Deferred Class	Abstract Class
Deferred Routine	Abstract Method
Frozen Class	Non-Extendable Class
Generic Class	Generic Class
Generic Routine	Generic Method
Generic Parameter	Type Parameter
Inheritance Access Identifier	Inheritance Visibility
Local Variable	Local Variable
Name	Name
Return Type	Return Type
Root Class	Root Class
Routine	Instance Method & Non-Overridable Method
Routine Redefinition	Overridable Method
Subclass	Subclass
Superclass	Superclass
Type	Type
Universe	Physical Module
Value Type	Value Type

3.1.3. C++ Language

Table 3 shows the matches between the C++ language and the OOC-O.

Table 3 Equivalence between C++ Language and OOC-O.

Language Concept	**OOC- concept**
Abstract Class	Abstract Class
Accessor Function	Instance Method
Access Modifier	Element Visibility
Block	Block
Class	Concrete Class & Extendable Class
Constant Variable	Mutability
Constructor Function	Constructor Method
Data Member	Instance Variable
Destructor Function	Destructor Method
Final Class	Non-Extendable Class
Final Function	Non-Overridable Method
Inheritance Access Identifier	Inheritance Visibility
Local Member	Local Variable
Member Function	Instance Method
Member Function Return	Return Type
Name	Name
Namespace	Logical Module
Nested Class	Nested Class
Parameter	Parameter Variable
Predefined Type	Primitive Type
Pure Virtual Function	Abstract Method
Static Data Member	Class Variable
Static Function	Class Method
Subclass	Subclass
Superclass	Superclass
Template	Generic Class & Generic Method
Template Variable	Type Parameter
Type	Type

Value Type	Value Type
Virtual Function	Overridable Method

3.1.4. Java Language

Table 4 shows the matches between the Java language and the OOC-O.

Table 4 Equivalence between Java Language and OOC-O.

Language Concept	**OOC- concept**
Abstract Class	Abstract Class
Abstract Method	Abstract Method
Accessor Method	Instance Method
Access Modifier	Element Visibility
Block	Block
Class	Concrete Class & Extendable Class
Constructor Method	Constructor Method
Final Class	Non-Extendable Class
Final Method	Non-Overridable Method
Final Variable	Mutability
Finalize Method	Destructor Method
Generic Class	Generic Class
Generic Method	Generic Method
Instance Variable	Instance Variable
Local Variable	Local Variable
Method	Instance Method & Overridable Method
Method Result	Return Type
Module	Logical Module
Name	Name
Nested Class	Nested Class
Package	Physical Module
Parameter Variable	Parameter Variable
Primitive Data Type	Primitive Type
Root Class	Root Class
Static Method	Class Method
Static Variable	Class Variable

Subclass	Subclass
Superclass	Superclass
Type	Type
Type Parameter	Type Parameter
Value Type	Value Type

3.1.5. Python Language

Table 5 shows the matches between the Python language and the OOC-O.

Table 5 Equivalence between Python Language and OOC-O.

Language Concept	**OOC- concept**
Abstract Class	Abstract Class
Abstract Method	Abstract Method
Accessor Method	Instance Method
Argument	Parameter Variable
Block	Block
Class	Concrete Class & Extendable Class
Class Attribute	Class Variable
Class Method	Class Method
Data Attribute	Instance Variable
Delete Method	Destructor Method
Generic Method	Generic Method
Initializer	Constructor Method
Inner Class	Nested Class
Local Variable	Local Variable
Method	Overridable Method
Name	Name
Namespace	Logical Module
Package	Physical Module
Root Class	Root Class
Subclass	Subclass
Superclass	Superclass
Template	Generic Class
Type	Type

Type Parameter	Type Parameter

3.2. OOC-O and OO Programming Languages

The subsections presents the harmonization of the sub-ontologies related to each programming language.

3.2.1. Object-Oriented Ontology - Core Module

Table 6 shows the matches between the OOC-O Core Module and the selected languages (Smalltalk, Eiffel, C++, Java and Python).

Table 6 Equivalence between OOC-O Core Module and selected languages.

Concept	Programming Languages				
	Smalltalk	**Eiffel**	**C++**	**Java**	**Python**
Attribute (Member Variable)	Shared Variable & Instance Variable & Temporary Variable & Parameter Variable	Attribute & Local Variable & Argument	Data Member & Static Data Member & Local Member & Parameter	Static Variable & Instance Variable & Local Variable & Parameter Variable	Class Attribute & Data Attribute & Local Attribute & Argument
Class	Class & Abstract Class & Template	Class & Deferred Class & Frozen Class & Generic Class	Class & Abstract Class & Final Class & Template	Class & Abstract Class & Final Class & Generic Class	Class & Abstract Class & Template
Element Visibility	Access	Access	Access Identifier	Access Modifier	No Related - The language defines all elements as public visibility.
Logical Module	Namespace & Category	Cluster	Namespace	Module	Namespace
Member	Method & Abstract Method & Class Method & Template & Shared Variable & Instance Variable & Temporary Variable & Parameter Variable	Routine & Deferred Routine & Routine Redefinition & Generic Method & Attribute & Local Variable & Argument	Member Function & Pure VIrtual Function & Final Function & Virtual Function & Static Function & Template & Static Data Member & Data Member & Local Variable & Parameter	Method & Abstract Method & Final Method & Static Method & Generic Method & Static Variable & Instance Variable & Local Variable & Parameter Variable	Method & Abstract Method & Class Method & Generic Method & Class Attribute & Data Attribute & Local Attribute & Argument
Method (Member Function)	Method & Abstract Method & Class Method & Template	Routine & Deferred Routine & Routine Redefinition & Generic Method	Member Function & Pure VIrtual Function & Final Function & Virtual Function & Static Function & Template	Method & Abstract Method & Final Method & Static Method & Generic Method	Method & Abstract Method & Class Method & Generic Method
Module	Namespace & Category & Directory	Cluster & Universe	Namespace & Directory	Module & Package	Namespace & Module
Mutability	No Related - The language has no mechanisms to ensure that a value is not changed.	Constant Attribute	Constant Member	Final Variable	No Related - The language has no mechanisms to ensure that a value is not changed.

Name	Name	Name	Name	Name	Name
Named Element	Class & Abstract Class & Template & Method & Abstract Method & Class Method & Template & Shared Variable & Instance Variable & Temporary Variable & Parameter Variable	Class & Deferred Class & Frozen Class & Generic Class & Routine & Deferred Routine & Routine Redefinition & Generic Method & Attribute & Local Variable & Argument	Class & Abstract Class & Final Class & Template & Member Function & Pure VIrtual Function & Final Function & Virtual Function & Static Function & Static Data Member & Data Member & Local Member & Parameter	Class & Abstract Class & Final Class & Generic Class & Method & Abstract Method & Final Method & Static Method & Generic Method & Static Variable & Instance Variable & Local Variable & Parameter Variable	Class & Abstract Class & Template & Method & Abstract Method & Class Method & Generic Method & Class Attribute & Data Attribute & Local Attribute & Argument
Physical Module	Directory	Universe	Directory	Package	Package
Primitive Type	No Related - The language supports only the class type	No Related - The language supports only the class type	Predefined Type	Primitive Data Type	No Related - The language supports only the class type
Return Type	No Related - The language adopts a unique type, object.	Result Type	Member Function Return	Method Result	No Related - The language does not define the return type in the method declaration. It associates the type of the returned value to the method in runtime.
Type	Type	Type	Type	Type	Type
Variable	Shared Variable & Instance Variable & Temporary Variable & Parameter Variable	Attribute & Local Variable & Argument	Static Data Member & Data Member & Local Variable & Parameter	Static Variable & Instance Variable & Local Variable & Parameter Variable	Class Attribute & Data Attribute & Local Attribute & Argument
Value Type	No Related - The language adopts a unique type, object.	Value Type	Value Type	Value Type	Variable Definition

3.2.2. Object-Oriented Ontology - Class Module

Table 7 shows the matches between the OOC-O Class Module and the selected languages (Smalltalk, Eiffel, C++, Java and Python).

Table 7 Equivalence between OOC-O Class Module and selected languages.

Concept	**Programming Languages**				
	Smalltalk	**Eiffel**	**C++**	**Java**	**Python**
Abstract Class	Abstract Class	Deferred Class	Abstract Class	Abstract Class	Abstract Class
Concrete Class	Class	Class	Class	Class	Class
Extendable Class	Class	Class	Class	Class	Class
Generic Class	Template	Generic Class	Template	Generic Class	Generic Class

Inheritance	Subclass & Superclass	Subclass & Superclass	Subclass & Superclass	Subclass & Superclass	Subclass & Superclass
Inheritance Visibility	No Related - The language has no mechanisms to control the visibility of elements inherited by the subclass.	Inheritance Access Identifier	Inheritance Access Identifier	No Related - The language has no mechanisms to control the visibility of elements inherited by the subclass.	No Related - The language has no mechanisms to control the visibility of elements inherited by the subclass.
Nested Class	No Related - The language does not support	No Related - The language does not support	Nested Class	Nested Class	Inner Class
Nesting	No Related - The language does not support	No Related - The language does not support	Nested Class	Nested Class	Inner Class
Non-Extendable Class	No Related - The language has no mechanisms to ensure that the class is extended.	Frozen Class	Final Class	Final Class	No Related - The language has no mechanisms to ensure that the class is extended.
Root Class	Root Class	Root Class	No Related - The language has no root class inherited by all other classes.	Root Class	Root Class
Subclass	Subclass	Subclass	Derived Class	Subclass	Subclass
Superclass	Superclass	Superclass	Base Class	Superclass	Superclass
Type Parameter	Template Variable	Generic Parameter	Template Parameter	Type Parameter	Type Parameter

3.2.3. Object-Oriented Ontology - Class Member Module

Table 8 shows the matches between the OOC-O Class Member Module and the selected languages (Smalltalk, Eiffel, C++, Java and Python).

Table 8 Equivalence between OOC-O Class Member Module and selected languages.

Concept	**Programming Languages**				
	Smalltalk	**Eiffel**	**C++**	**Java**	**Python**
Abstract Method	Abstract Method	Deferred Routine	Pure Virtual Function	Abstract Method	Abstract Method
Accessor Method	Accessor Method	No Related - The language has no mechanisms to ensure that the attribute is accessed only by the access method.	No Related - The language has no mechanisms to ensure that the attribute is accessed only by the access method.	No Related - The language has no mechanisms to ensure that the attribute is accessed only by the access method.	No Related - The language has no mechanisms to ensure that the attribute is accessed only by the access method.
Block	No Related	Block	Block	Block	Block
Class Method	Class Method	No Related - The language only supports instance method	Static Function	Static Method	Class Method
Class Variable	Shared Variable	No Related - The language only	Static Data Member	Static Variable	Class Attribute

		supports instance variable.			
Constructor Method	No Related - The language does not need to use a constructor since the class itself is an object as an metaclass instance.	Creation Routine	Constructor	Constructor	Initializer
Concrete Method	Method	Routine	Member Function	Method	Method
Destructor Method	No Related - The language only uses garbage collector	No Related - The language only uses garbage collector	Destructor Function	Finalize Method	Delete Method
Generic Method	Template	Generic Method	Template	Generic Method	Generic Method
Instance Method	Method	Routine	Member Function	Method	Method
Instance Variable	Instance Variable	Attribute	Data Member	Instance Variable	Data Attribute
Local Variable	Temporary Variable	Local Variable	Local Member	Local Variable	Local Attribute
Overridable Method	Method	Redefinition Routine	Virtual Function	Method	Method
Non-Overridable Method	No Related - Language has no mechanisms to prevent a method from being overwritten.	Routine	Final Function	Final Method	No Related - Language has no mechanisms to prevent a method from being overwritten.
Parameter Variable	Parameter Variable	Argument	Parameter	Parameter Variable	Argument
Return Type	No Related - The language adopts a unique type, object.	Result Type	Member Function Return	Method Result	No Related - The language does not define the return type in the method declaration. It associates the type of the returned value to the method.
Type Parameter	Template Variable	Generic Parameter	Template Parameter	Type Parameter	Type Parameter