Exotic Aspects of Object Orientation

ir. Gerald Kristen MBA

Recently, I was invited to India to present a series of seminars on the KISS methodology for Object Orientation. The visit to India, and the title of this magazine, inspired me to place a number of general object-oriented concepts and the KISS activities in particular in a may be very unusual, exotic perspective.

Expectations of India

Before leaving the Netherlands, I talked about my visit to India with a number of friends who already visited India before. Some visited India for only a short period and others were employed in India for a longer period of time, but they all had the same remark: "India is a country that is different from what you know and in order to get to know what the difference is, you have to experience India". At the time of listening to my friends, I started building images in my mind of what I could expect. Looking back, I can say that most of those images were completely biased by my knowledge and experience. I placed everything in the available framework and related the things I heard to it.

When I arrived at Delhi airport in the middle of the night, I soon knew that my images had to be changed dramatically. The first thing which had to be changed was the enormous pressing heat of the monsoon period in India. The pressing heat is

Figure 1: The Taj Mahal temple in Agra, India (Photo: Gerald Kristen)

even present in the middle of the night. The next experience is that driving in a taxi from the airport to the hotel is best described as a 'low speed kamikaze ride'. The taxi driver switches from the left to the right of the road in trying to avoid hitting unlighted trucks, three-wheel taxis (also called tjuk-tjuks), holy cows in the middle of the road, bicycles riding on the highway, and last but not least the holes in the road. With every passing and crossing, the horn was used abundantly in order to signal to other not yet remarked road users and slow cars. A relative large number of cars with problems were standing in the middle of the road repairing broken axes, flat tyres, or just something related to their age. Taking a left or a right turn mostly happens without using any signalling lamps. In the best case the horn is used or a hand signal comes out of the car, truck, or bus. In the Indian traffic, there are nearly no rules but the right of the strongest. On the other hand, European traffic is highly regulated, standardized, and facilitated and gives most rights to the weakest user of the road. With the first traffic experiences, I had to think back to the saying of my friends that you have to experience India in order to get any idea or understanding of it.

Experiencing Object Orientation

In the past years, I have given many seminars and presentations on the KISS method for object orientation [1]. The audience varied from experienced users of other object oriented methodologies to managers who are inexperienced to any methodology at all. When introducing the basic concepts of the KISS method and stating that the KISS method is fundamentally different from other object-oriented methodologies, I very often remark that persons who are more knowledgeable to those other object-oriented methodologies do not believe that there are really fundamental differences in the KISS method. They start placing the KISS concepts within the framework of object orientation they already have, accepting all things they recognise and rejecting all things they are not familiar with from the other methodologies. This approach

is very difficult if you have to start to adopt a complete new foundation in order to get a proper understanding of the KISS method. As a result, I saw that some of the persons who were knowledgeable to other methodologies blocked, while those without any reference framework could absorb the new concepts without too many problems.

In order to improve the acceptance and learning process, I have developed the KISS DOMINO$^{(R)}$ game. It gives the users of the KISS method the possibility to experience the KISS method. KISS DOMINO is a game with all the different symbols of the KISS method printed on domino pieces in seven different colours. KISS DOMINO is a very physical and interactive way for explaining the KISS concepts to anybody. Besides training purposes we at the KISS organization always use the KISS DOMINO game for defining the specifications of real-life projects. Looking at my experiences in India, I draw an analogy with the KISS method. I advise to persons interested in the KISS method to start experiencing it by using it in real projects. Reference frameworks of existing methodologies can give a false picture of the KISS method. The similarity of my understanding of India at the end of my first visit with the understanding of the KISS method at the end of a seminar is striking to me.

KISS method

The KISS method for the specification of objects is based on a number of concepts that are closely related to each other. The most important concept is 'object type'. The object type is the generic description of the life of objects. The life is described by so called KISS models. In a KISS model the life of objects is described by placing the actions that an object can execute in a structure with 'Iterations', 'Selections', and 'Sequences'. The first action type in the KISS model under the symbol of the object type is the instantiating action. It creates new object instances of the object type. After the object is instantiated, an object can execute all actions according to the

KISS model of that object type, until the life of the object is terminated by the last action type of the KISS model. The diagram of a KISS model consists of one rectangle representing the object type, several diamonds representing the action types, and control symbols representing the iterations and selections. A very simple KISS model which describes the life of customers who are instantiated by the action type 'to register', and who can iteratively order articles and be terminated by the action type 'to leave' is represented in Figure 2.
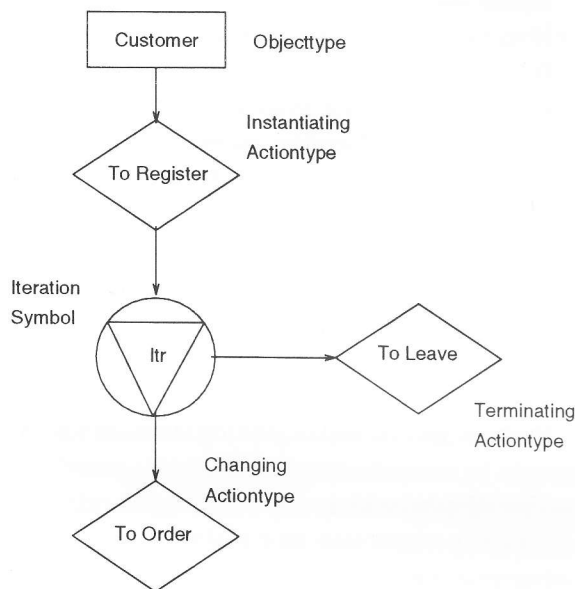

Figure 2: KISS model for the life of customers.

The KISS model describes the life of objects in independent models. The objects that are instantiated by the first action type of the KISS models always form a group of objects. In the KISS method a group of objects is called an *object class*. Two special cases of the object class are the *specialization* and the *category*. The specialization and the category classify objects to an object class based on the execution of action types. The object type itself has no object instances. It only coordinates the way in which object instances may execute actions. The object instances themselves can thus be found back in the object classes to which they are classified. The specification of the object classes is based on classification criteria which themselves can be dependent upon action types. As such, we can conclude that the KISS method makes a distinction between three levels at which objects are represented.

1. Object instance level.

2. Object type level for the life of objects.

3. Object class level for classification of objects.

All three levels are completely orthogonal to each other.

Hindu Gods and Exotic Objects

A first visit to India always includes a sightseeing tour with several temples on the program. India is blessed with several of the most beautiful temples of the world. We only have to think of the Taj Mahal at Agra and the many temples of the Hindus dedicated to one or more gods. In India Hinduism is practised by approximately 80% of the population of over 500 million people. The interesting thing about the Hindu religion is that it has a pantheon of gods, each directed to specific events in the real world. In fact all these different gods can be seen as physical representations of one omnipresent god. The three most important physical representations of god in the Hindu religion are Brahma, Vishnu and Shiva. Brahma is the *Creator*, Vishnu is the *Preserver* and Shiva is the *Destroyer*. All three gods are usually represented by four arms to represent their above human abilities, but Brahma has the advantage of four heads to represent his all-seeing ability. Each god can be represented by different symbols. The special thing about the gods is that real world activities are attributed to the gods with very limited abstraction in their representation and much abundance in details. When we give a closer look at the Hindu gods, we see that they support the life of living beings. Brahma is the god for creation and instantiation of new life. Brahma represents the generic instantiator of new objects. Brahma can have many different forms and behaviors. Brahma is not just the action 'to create', because

he takes much more into account with the instantiation of objects. Vishnu is the god that takes care of all changing actions that take place in the life of persons. Vishnu can also have many different semantic representations which makes Vishnu completely different from 'to modify'. Finally, Shiva is actually the god that fights all kinds of destruction. Looking at the three different main gods of the Hindu religion, we see that they are dedicated to support the life of objects. As such, they are comparable to the action types in a KISS model. Either a god, an action type, or an object type are the real world things. They are all abstractions of the real world for describing the life of objects. As such, the one omnipresent god in the Hindu religion can be seen as an *exotic object type* with many physical representations.

Figure 3: Hindu Gods (Photo: Gerald Kristen)

Exotic Classification

Basically the Hindu religion postulates that we all go through a series of rebirths or reincarnations, that will eventually lead to the spiritual salvation which frees one from the cycle of rebirths. With each rebirth, you can move closer to or further from eventual spiritual salvation. The deciding factor is the *karma*. Bad actions during life result in a bad karma, which ends in a lower reincarnation. Good actions will increase the karma and one will be reincarnated on a higher level and be a step closer to eventual freedom from rebirth. The level of reincarnation is closely related to the caste system of the society. The Hindu religion has four main castes: the Brahmin, or priest caste; the Kshatriyas, or soldiers and governors; the Vaisyas, or tradespeople and farmers; and the Sudras or menial workers and craftspeople. These basic castes are then subdivided into a great number of lesser divisions. Beneath all the castes are the Harijans, or untouchables, the lowest cast-less class for whom the most menial and degrading tasks are reserved.

The Hindu society is based upon classification of persons into four main caste classes and one cast-less class. The main classes themselves are further classified into many subclasses. It is not possible for an individual to change class during his life. So, in order to be ranked in a higher class one has to live the existing life well in order to get a high karma. When dead, one has to be cremated and then one can reincarnate into a higher class. This approach is also followed by object-oriented programming languages. Objects are always instances of one class and will therefore belong to that class. When applying inheritance with inheriting the generic characteristics of the superclass to subclasses, one can see a number of problems. This is the case where object instances of a subclass cannot go to another subclass without getting terminated first and then be reincarnated in the other subclass. These limitations of object-oriented programming languages cause several problems. In order to offer solutions for this problem, one sometimes

apply techniques like multiple inheritance of the subclasses to a new subclass, which in itself will cause further problems. In general, application of inheritance in class structures creates rigidity and less adaptability to new situations in the real world. The classification principles and class concepts of object-oriented languages are equal to those of the caste system of the Hindu religion. In the Hindu society, the application of the caste system has resulted in very rigid social structures. The advantage was that it gave very stable social structures, but the disadvantage is that the society itself is unable to change as long as the caste structures remain.

In the KISS method, classification of objects is not determined by the above class concept of object-oriented programming languages. The class of programming languages both instantiates and classifies the objects with one and the same concept. In the KISS method, the instantiation of objects is determined by the object type. The classification of objects to object classes is in the KISS method independent of the instantion of objects. As such, the classification of objects into subclasses does not create the rigidity we have seen in the situation above. The more exotic classification of objects based upon the implemented classes in an object-oriented programming language has therefore to be seen as different from the classification of objects into object classes were the classification will remain independent from the instantiation of objects.

Conclusions

Visiting and experiencing India, I found is worth more than a thousand stories. In analogy with this, I can say that applying and experiencing the KISS method is worth more than thousand comparisons. The basic concepts of the KISS model for describing the life of objects are also covered by the three main gods of the Hindu religion: Brahma, Vishnu, and Shiva. The classification principles of object-oriented languages create caste-like structures in programs. Objects have to be killed and reincarnated in order to clas-

sify them in another subclass. The caste system of the Hindu religion is based on the same paradigm. The classification principles of the KISS method are independent of the instantiation of objects. And so a much more flexible framework is created than with class-instantiation. Further elaboration of other exotic topics may clarify other topics of interest to object orientation, like for example reuse and standardization. These topics are left to future articles. □

References

[1] Gerald J.H.M. Kristen,
     Object Orientation, the KISS method. From In-
     formation Architecture to Information System,
     Addison-Wesley, 1994

Ir. Gerald J.H.M. Kristen MBA studied Industrial Engineering at Eindhoven University of Techno-logy and Master of Business Administration at the Catholic University of Leuven, Belgium. He is director of Kristen Information & Software Ser-vices (KISS) b.v., which he founded in 1990. Kristen has authored a book and several papers on object-orientedness. KISS b.v. employs an OOTI graduate.