The Ether

–

A philosophical, cosmic

Structure

Version 1.2

Lupius Mohnschein, December 2022

I dedicate this book to my family, who supported me for years and gave me a space to learn what I do now. Also to F., who made me realise again that I enjoy programming. And of course my friends, for everything.

„Have the courage to use your own reason!“ - Immanuel Kant

The Ether is a concept in the tradition of the enlightenment and should only be used in this sense.

Contents

Overview

This chapter gives an overview of the description of the Ether. First, it gives a basic understanding of the Ether and the sources on which it is based on. Then a system is explained, which divides the structures and inner workings of the Ether in sensible units, and which is followed in the description of the Ether in this book. It will be from now on be referred to as the „ether-teaching“. It is divided into six pillars, each of which is divided into three layers. Alternative systems would certainly be possible but are not discussed in this book.

Basic Understanding of the Ether

The basic understanding of the Ether that is adopted throughout the book is the following: The Ether is a cosmic entity with which we, as living beings, can mentally interact with; this interaction enables us to physically manipulate real-world-things without the exercise of any physical force, and to retrieve information about them directly from the things themself. This tries to put one in the state of mind that is necessary to understand, interact with and alter the state of what we call "the digital world".

Hypothetically, in a world in which the Ether exists, the interaction with it will commonly be referred to as magic or the use of the gift of a higher power. However, this should not distract from the fact that it is a philosophical construct and should always be understood as such. It was created with the primary purpose to better understand the digital world and the interaction with it; it may be used secondary for literature or education; but never for any religious or totalitarian purposes. When adapted for literature, it may be embedded in a form of mythology, which, however, this book has tried to avoid as best as possible, in order to focus on the theoretical constructs at its core and keep it as universal as possible. Fictional mythologizations that came to the author's mind during the writing of this book have therefore been clearly separated from the descriptions of the Ether through the use of quote-blocks (these sections were in a later editing completely removed from the document to emphasize the philosophical nature of the structure). The structure of the Ether has also been attempted to be formulated as culturally independent as possible, even though they have been partly inspired, both conceptually and in terms of their designations, from Celtic, Hindu and Indo-European mythological concepts. In general, however, the Ether still claims to be independent of culture and religion, which is why further generalisation should be considered in the future.

The Six Pillars

The ether-teaching is divided into six pillars, which can be imagined fictionally-mythologised as stone pillars, as seen in figure 1. Each pillar introduces new concepts and at the same time extends the introduced concepts of previous pillars. The pillars thus build on each other and offer a ever deeper insight into the Ether and its possibilities.

While the first five pillars convey content, the sixth pillar is merely a placeholder and symbolises the discovery of the Ether on one's own. With that, the content part of the

Figure 1: Fictional mythologization of the six pillars of the ether-teaching

ether-teaching actually consists of only five pillars.

The Three Layers

Each of the pillars in turn is divided into three layers: the cosmic, the medial and the worldly (see figure 2). The layers are different ways of looking at the concepts of the Ether and help to better understand the concepts introduced in each pillar: starting with what they are, in the cosmic layer, about how to access them, in the medial layer and finally how to use them, in the worldly layer.

The cosmic layer describes the inner structures of the Ether and how they work. It is the most theoretical layer and probably the most difficult to understand, as it has no practical reference. However, in this book, all pillar chapters start with the cosmic layer, as the other layers build on the concepts introduced in it.

Next comes the medial layer. While the cosmic layer is rather theoretical, the medial layer uses the concepts of the cosmic layer practically for interacting with things in the Ether. It describes how we can call things in the Ether and interact with them, in a way that it also influences things in the physical world, and for example, make them fly in the air. Therefore, the medial layer has a much higher practical relevance than the cosmic layer.

The worldly layer, finally, concentrates itself completely on the practical application of the Ether. It builds on the learned possibilities of the medial layer to enable more complex interactions with the Ether through dances, so-called „ether-dances“. These

Figure 2: Overview of the three layers of the ether-teaching

are a mixture of movement and spoken words, and can be used to formulate long chains of commands for the interaction with things in the Ether and have them afterwards executed in a row.

The Ether-Teaching

The six pillars together with the three layers form the ether-teaching, which can be seen in figure 3.

If one wants to learn Ether, one starts with the first pillar, internalise its concepts for all three layers and then goes to the next pillar until one reaches pillar 6 and thus has understood the Ether in its entirety.

The circles mark the focal points of the respective pillar. One can see that the pillars shift their focus more and more from the worldly to the cosmic layer as the pillar number increases. This could lead one to think that it is sufficient to understand only the first three pillars in order to be able to do ether-dances and then discontinue the studies. However, the last pillars contain valuable knowledge about the Ether, only with

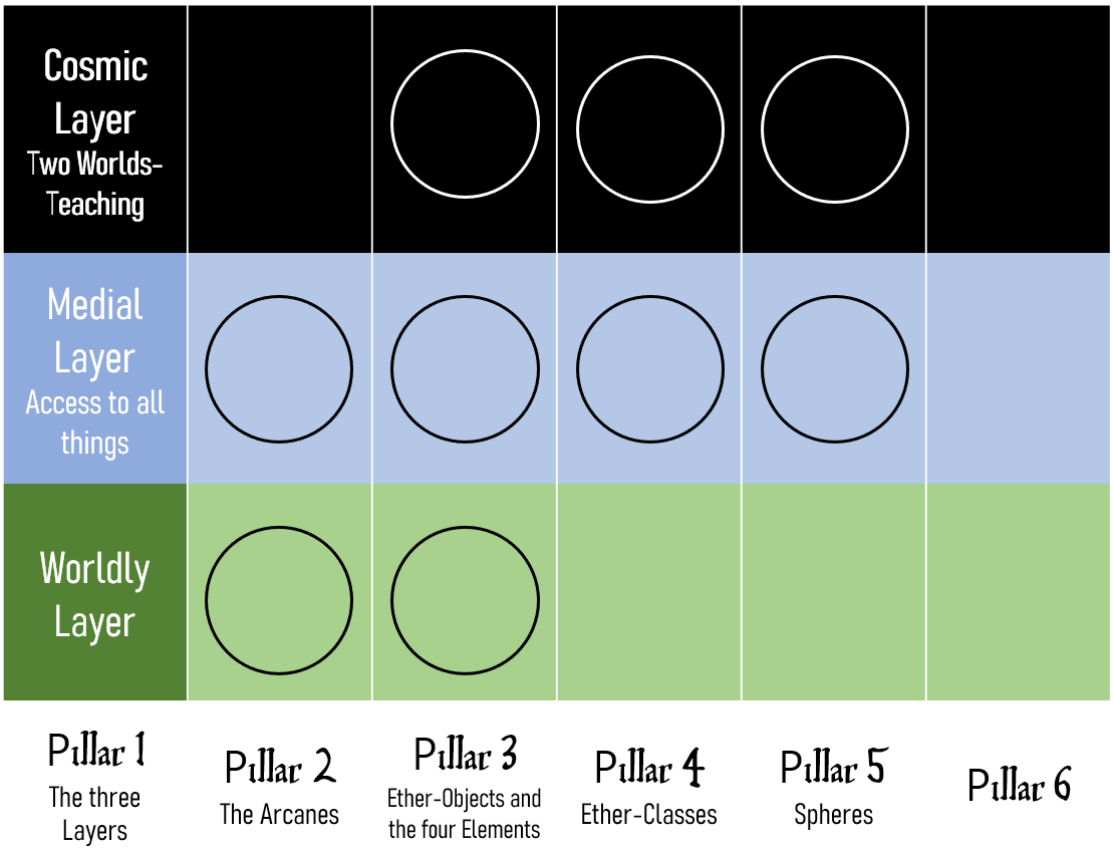

Figure 3: Ether-Teaching overview (marked are the focal points of the respective pillar)

which ether-dances can come into their full form. Namely, they have two central teachings that are imparted in them and whose understanding is important for the practical performance of ether-dances (even if their dance moves and language constructs have already been understood); these teachings are: the teaching of the two worlds, and in particular: the access to all things.

In the cosmic layer, the two worlds-teaching is explained right at the beginning, but its contents are expanded with each pillar. It states that there are two worlds: the physical world, which contains all the things we can see and touch, and the world of the Ether, which also contains all the things in the universe, but which we cannot perceive with our senses. The only way to experience/interact with the Ether is through words and thoughts. So, we can think of the Ether as a twin of our visible world, with which we can communicate in order to manipulate things in the „real“ world without actually physically interacting with them. It thus makes sense to understand the higher pillars as well, since with each additional structure in the Ether that is learned about, access to new physical representations becomes possible as well.

The medial layer is based first of all on the understanding that all living beings in the universe have the ability to be able to call the representations of other things in the Ether and thereby interact with them. This is called the teaching of calling. Unlike the two worlds-teaching, however, this one has a concrete goal. The medial layer follows the idea that it must be possible to call all representations of things in the universe via the Ether and thus (at least theoretically) be able to interact with every thing in the universe. This goal is called the „access to all things“ and is not achieved until the fifth pillar. For the attainment of the goal, the knowledge of the cosmic layers is essential.

Both layers thus contain important concepts that are only taught in the last pillars, but are necessary in order to realise the full potential of ether-dances. The back pillars should therefore not be left out either, even though the basic concepts and constructs of ether-dances are already completed in the first three pillars.

1 First Pillar: The Three Layers

The first pillar provides a basic understanding of the Ether and thus lays the foundation for all other pillars. One learns what the Ether is, as well as its three layers. Some of these topics have already been covered in chapter and are continued in this chapter.

1.1 Cosmic Layer: Structure of the Ether

The cosmic layer describes the inner structure of the Ether. It is based on the two worlds-teaching. The concepts of the Ether introduced in the various pillars complement each other to form increasingly complex and complete models of the Ether, but never describe it in its entirety.

1.1.1 The Two World-Teaching

The cosmic layer describes the structure and behaviour of the Ether without reference to the world we can perceive with our senses. It is thus exclusively theoretical in nature.

It is based on the theory that there are two worlds: the physical world (our world) and the Ether and that they are in constant interaction with each other (illustrated in Figure 1.1). The interaction between the two worlds means that as soon as something changes in the other world, this change also shows up instantly in the other world. This theory is called the two worlds-teaching and has already been partly covered in the last chapter but is now further explained. While the physical world contains everything that exists in the universe, all things also exist in the Ether, but not physically, but „etherically“, that is, in the form of information.

That means, each thing in the physical world has a twin in the Ether, that constantly update its state in relation to the „real“ one, and vice versa. In Figure 1.2 you see a wolf and a tree in their different representations in the physical world and in the Ether.

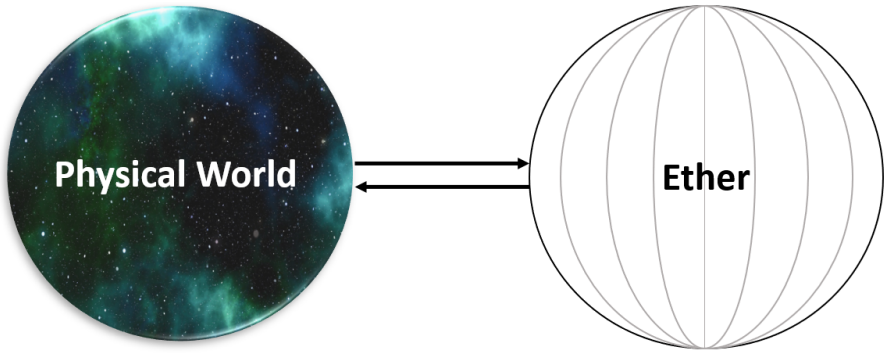

Figure 1.1: Illustration of the two worlds-teaching

As can be seen, the thing in the Ether is represented exclusively by information. In fact, the corresponding object in the Ether contains all the information of the real object that can possibly exist (of course, this could only be implied in the illustration). All this information is constantly updated according to the status of the thing in the physical world. This means that if the wolf becomes a year older, this will also be reflected in the Ether without any time delay. This also works the other way round: if a value changes in the Ether, this is immediately reflected in the thing in the physical world.

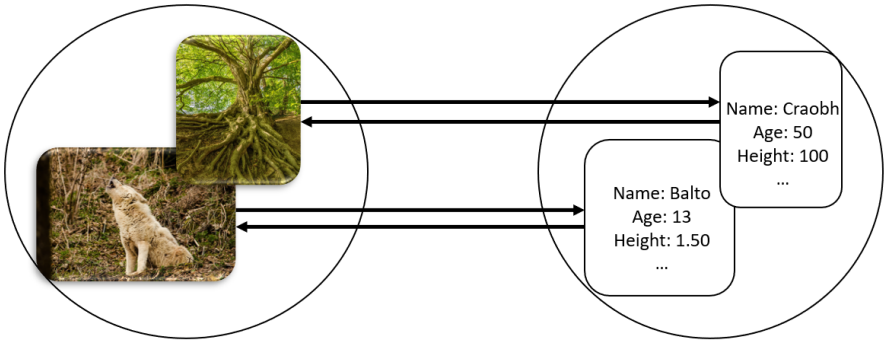

Figure 1.2: Different representations of a wolf and a tree in the physical world and in the Ether

The exact nature of things in the Ether, their inner workings and the structures they form are explained in higher pillars.

1.1.2 Ether-Models

In order to describe the structures in the Ether more precisely and comprehensively, a model of the Ether is defined in each pillar of the cosmic layer. These models are called ether-models. Each ether-model of the current pillar builds on the model of the previous pillar and expands it with new concepts. The ether-model of a pillar is also used in the other layers and forms the basis for discovering new possibilities with the Ether in the

respective pillar. With each pillar, the ether-model is thus expanded and becomes more complete. At the beginning, it is still quite simple, but it is expanded with each pillar by more and more elements in order to be able to describe things more and more precisely and thus also to have more and more possibilities in interacting with them. This will become apparent especially in the medial layer. However, since it is only a model, it never represents a complete description of the Ether, even in the fifth pillar.

1.1.3 Relationship to other Layers: Foundation of the other Layers

The cosmic layer thus provides the basis for all other layers. It is also the most theoretical layer. While the worldly and medial layers are close to our world and take the human point of view to ask how the Ether appears in their lives and can be used by humans, the cosmic layer describes the Ether as a cosmological phenomenon without going into its practical use.

1.2 Medial Layer: Calling

The medial layer revolves around calling and interacting with the constructs of the Ether to influence things in the physical world. Usually, based on the constructs introduced in the cosmic layer, new possibilities arise in these areas.

The medial layer is based on the idea that theoretically all objects in the Ether can be called. The term for achieving this ability is called the „access to all things“ and is the highest goal of this layer (illustrated in Figure 1.3).

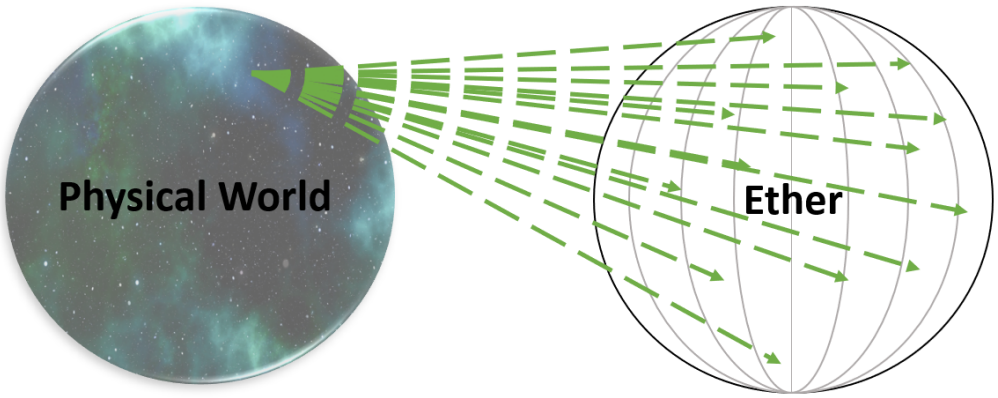

Figure 1.3: Illustration of the access to all things-goal

1.2.1 Overview

Cosmic and worldly layers are easy to understand: one concerns theory, the other the practical application of that theory. The medial layer is usually the most difficult layer to grasp; however, it can be simply understood as: I have a thing here, for example a

tree, and I want to interact with it through the Ether. How does that work? The answer to this question is: by calling it in the Ether.

Calling in this case means a mental form of calling. It means that one has a mental connection to the thing with which one wants to interact, similar to calling a thing of the physical world with a word, for example calling a dog named Larko with the word „Larko“. One would immediately, even if one would not see the dog at the moment, have an image of it in one's mind. But of course it is not quite that simple; how calling exactly works and how one can theoretically be able to call arbitrary things in the universe is therefore explained step by step in the medial layers of the different pillars, until in the fifth pillar one is in a position to have mastered or at least theoretically understood the concept of the access to all things.

The communication with the Ether for all this happens through a medium, which is called the ether-channel, and can simply be used with the words and movements that one utters/does into the air around one. If they correspond to a certain structure, these words and moves can be used to interact with an object in the Ether. The process of interaction consists of two steps: calling a thing and the actual interaction.

After calling, one is able to interact with the called thing. There are two forms of interaction:

- Information Retrieval (passive): Retrieval of information e.g. size, width, colour, etc.
- Active Manipulation (active): Thing is actively manipulated, e.g. exerting force, heating, levitating, etc.

Word-Sequences that call things in the Ether or interact with them are called commands or a spells. Spells for both forms of interaction are learned in the following pillars.

1.2.2 Executing Ether-Commands

In order to execute ether-commands in the Ether, one must first establish a communication with the Ether. This is very simple: the medium to communicate with the Ether is called the ether-channel and can be accessed simply trough the air around us. All the words one says, all the movements one does and all the thoughts one haves can be transmitted through the ether-channel to the Ether. A broad and immediate access to the Ether is thus possible for every thinking living being.

However, not every word we utter nor every move we do ends up in the Ether, because the ether-channel can be opened and closed mentally. By concentrating on the Ether, one can open it; if one withdraws the concentration, the ether-channel closes again. In order to perform the opening and closing of the ether-channel more reliably, mental efforts are often linked to certain movements, e.g. an opening movement to open the channel and a closing movement to close the channel.

Furthermore, not every word we speak into the Ether has an effect. Most of the words we can send to the Ether have no practical effect. In order to actively communicate

with the Ether, there are certain sequences of words called commands, or spells, that are executed in the Ether after they are spoken into the ether-channel. These are very similar to natural language, but follow certain rules so that they can be understood by the Ether. The various commands for interacting with the Ether are another main topic of this layer, along with the commands for calling things.

There is the possibility to execute commands individually, but also to speak them out in groups and then have them executed collectively in the Ether, and with the ether-dances also longer chains of commands become possible, which allow complex interactions with the Ether (see section 1.2.5).

It should be noted that, without going into too much detail, that the ether-channel can be opened and closed as widely as desired. It can therefore be thought of as a mental body opening which we can open, allowing our thoughts to flow into the Ether and receive things from it, or close it and thus cut off communication to the Ether. This also means that there are actually many subcategories between an open and closed ether-channel. For example, one can also open the ether-channel a little in order to do small mathematical computations in the Ether. Low communication with the Ether often happens unconsciously, for example, through thinking or mental calculations. This means that for active communication with the Ether, which is described in the pillars of this book, a certain layer of opening must be exceeded, which from now on will be called „conscious communication with the ether“.

1.2.3 Commands to Call Things

As a rule, things in Ether that you want to interact with have to be called. For example, if you want to interact with an apple, e.g. to let it fall from the tree, you must first call it.

In the course of the six pillars, two different forms of calling are introduced (see figure 1.4):

- Direct Calling: Calling a thing while touching it at the same time (introduce in pillar 3)
- Spherical Calling: Calling a thing through the spherical connections that link all things in the universe (introduced in pillar 5)

Direct calling is the simplest form of calling, where you have to touch a thing when calling it. It is much easier to learn and is already dealt with in the third pillar, but it quickly reaches its limits.

Spherical calling is more complicated, but it makes it possible to call things without touching them and thus to call any other thing from any place in the universe. Spherical calling is only dealt with in the fifth pillar and concludes the teaching of calling with the attainment of the access to all things-goal.

There is also a third way of calling things in the Ether, called retrieving or recalling, which is covered in the second pillar.

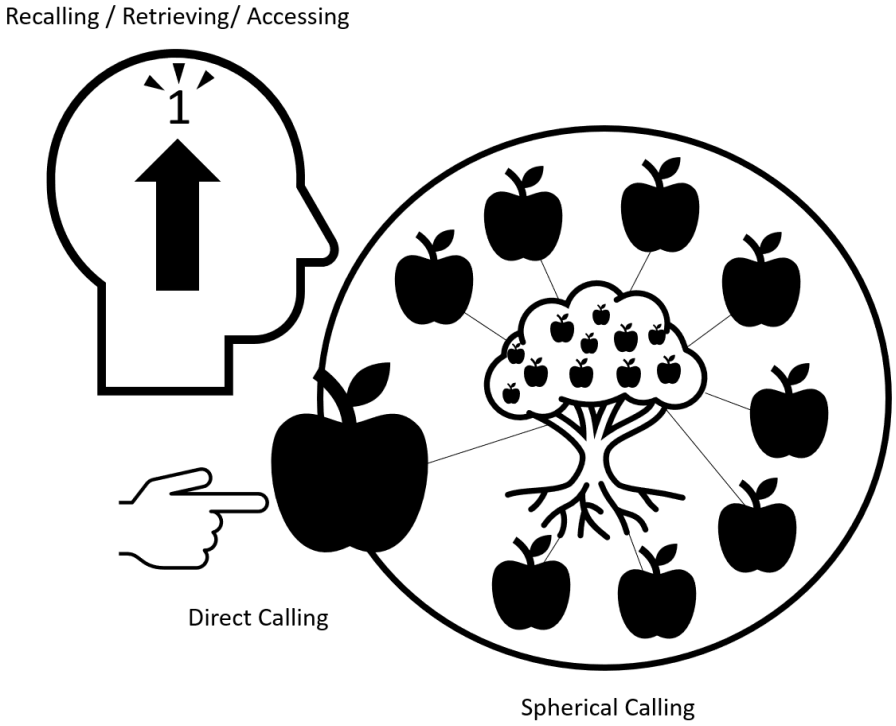

Figure 1.4: Different ways of calling

Teaching of the Names of All Things vs. Teaching of Calling all Things

At this point, one thing should be made clear that is crucial for understanding the communication with the Ether: Things do not have a unique name, that is, there is no „true name“ with which one can simply call things as is the case in some other cosmic constructs. In a world with the Ether, it is not so simple. As will become clear in higher pillars, objects in the Ether are uniquely identified solely by their position in the structures of the Ether. Since one is relatively from this position on another position in the structure of the Ether, this name (if one wants to call it that) is always different depending on where one currently is in structure of the Ether. Therefore, at least from the point of view of a living being, there can be no talk of the existence of unique names for the things in the Ether. From a theoretical, cosmic point of view, the location of each thing can be described unambiguously, which is why the term name could be used in this context, but from the author's point of view this only leads to confusion, which is why this term and suggestions in this direction are avoided in this book. For this reason, the term „naming“, which is used as a synonym for calling in the context of other cosmic constructs, is also not used in this book, as well as the use of the concept of the existence of the „names of all things“ and resulting „teaching of the names of all things“, because, even if valid in some theories, it could lead to false associations in the process of understanding the Ether.

It should be noted that this has nothing to do with the fact that in ether-dances sometimes things are given names, for example, when calling a dog by the name Balto, the name Balto is used to call it. This is done merely to make it easier to address the representations of a thing in the Ether, it is a mental strategy of humans to simplify calling, but has no practical significance beyond that. Whether one calls the dog „Balto“ or simply „Dog“ when calling it, does not matter in principle as long as one is mentally aware of calling the dog one wants to call. Names thus have no higher meaning in the Ether, and they have a special meaning for humans, mainly because they use them to simplify the calling of things and because they have certain social purposes.

1.2.4 Commands to Interact with Things

Once you have called a thing in Ether, you can interact with it. This is the second main part of the medial layer and enables first practical uses of the Ether.

Two types of interaction are distinguished:

- Retrieval of information
- Active manipulation

The two types are presented in more detail below.

Information Retrieval

By calling, the speaker of the command is briefly connected to the called element and can access it as if it would be a part of the speaker themself, similar to the connection that living beings naturally have through their own body and its representation in the Ether. On the one hand, this connection enables the retrieval of information, and on the other hand, the active manipulation of things. For example, by simply touching things, one can access their ether-representation and thus obtain information, for example, as seen in Figure 1.5, information about a dog and a tree by touching them and speaking to them aloud or mentally. For example, in the example, one could find out that the local people call the tree „Craobh“, or that the dog Balto is 13 years old.

Active Manipulation

In addition to retrieving information, things in the Ether can also be actively manipulated. Because of the interrelation between the two worlds, such a manipulation has also effects on the state of the physical thing. That means that through active manipulation, one is effectively able to manipulate physical things through the Ether as seen in Figure 1.6. We could, for example, burn the tree, whose ether-representation we called via touch in the last section, by lighting its ether-representation with the fire element, as seen in Figure 1.7. One would have to pull the hand away quickly, because the tree would burn almost instantly.

Figure 1.5: Accessing the ether-representation of a tree and a dog from the physical world

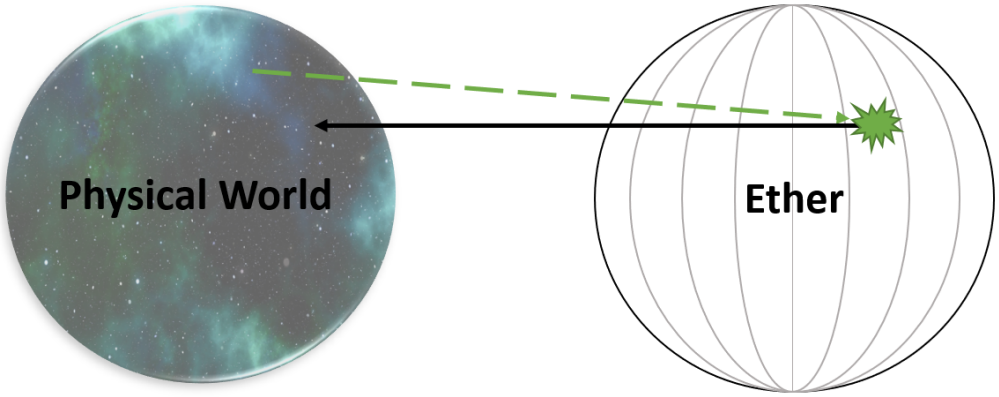

Figure 1.6: Active access and manipulation of the Ether with subsequent propagation of the change into the physical world

Figure 1.7: Access to the ether-representation of a tree and subsequent manipulation

How does this work exactly? Until now, one probably imagined the things in the Ether simply as information containers, whose content is determined by the physical thing. From this point of view, the physical things are active, while the ether-things are passive and just react. For example, a tree falls and crashes into a house, thereby changing the ether-information of the tree itself and the house it crashes into. However, it is also possible to flip this world-view and see all forces in the physical world originating in the Ether. This explains why, through changing the thing in the Ether first, one is able to bring about change in the physical world. In the given example, force would be exerted on the ether-representation of the tree so that it falls over and crashes onto the ether-representation of the house (all of this, of course, happening in the form of information). This, in turn, would influence the physical representations of the tree and the house, meaning that the crash would happen almost at the same time in the physical world.

1.2.5 Executing Groups of Ether-Commands

The medial layer only deals with the execution of single commands to call and manipulate things in the Ether, but there is also the possibility of executing ether-commands in groups. A total of three types of ether-communication are distinguished:

- (Single) Ether-Spell/-Command: single ether-command
- Spell-/Command-batch: group of (single) ether-commands
- Ether-Dance: group of ether-commands linked to more complex speech constructs and dance movements

The possibility of command-batches and ether-dances is the main subject of the worldly layer and it is therefore only described here how group-commands can be carried

out, as this belongs to the subject area of the medial layer, but the potentials that arise from this are left to the worldly layer. Apart from that, the medial layer only deals with the communication with the Ether through single commands.

In Figure 1.8 all types of communication with the Ether are presented in an overview. The example in this section is about building a stone tower. The individual forms are explained using this example in the following sections.

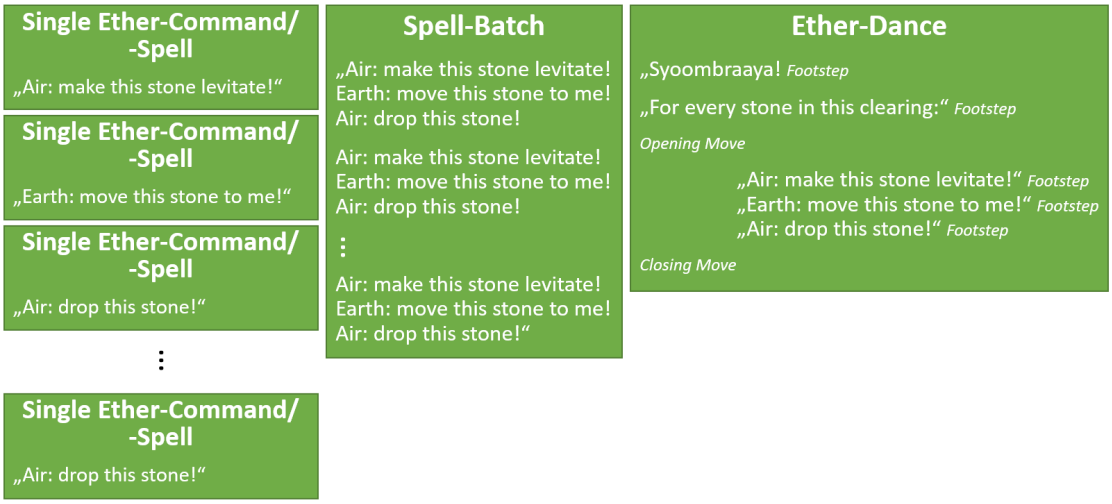

Figure 1.8: Example of direct ether-commands, a command batch and an ether-dance

Executing Single Ether-Commands

Building a stone tower with single ether-commands is rather tedious and lengthy. As can be seen in the illustration, several single commands are required to perform the individual actions with the stones.

Because commands can also be combined with each other and thus represent one command again (which contains several commands), at least somewhat effective interactions with single ether-commands are possible. For example, „Fire Element: Burn Bush!“ contains the commands to call the fire element and a command to apply the burn power to the bush, both commands together also represent a command again, as they are pronounced together. However, this still doesn't change the fact that even with nested commands it is tedious to build a stone tower. This shows the need for being able to perform groups of commands and have them executed in the Ether.

Executing Groups of Ether-Commands (Ether-Batches)

As already covered in the last pillar, the ether-channel can be opened and closed. Instead of one ether-command, it is also possible to say several single commands between opening and closing the ether-channel. Such a group of ether-commands is called an ether-batch. The separation of commands is mental, but is usually accompanied by a simple

movement, such as stepping on the foot. The spoken commands are meanwhile picked up by the Ether and executed one by one when the ether-channel is closed. That is, they are not executed directly, but are executed as a group when the communication is closed, one after the other, in the order in which they were spoken into the Ether. This means that even a single ether-command is not executed until the ether-channel is closed. However, since this usually happens directly after the command has been spoken, it is often assumed on the medial layer that it is executed quasi immediately.

For the construction of the stone tower, one could now pronounce all the commands that in the last section one had to pronounce individually as a group and thereby possibly save oneself work. This would require planning in advance what chain of commands would be necessary for the construction of the tower. The formulation of ether-batches usually also goes in hand with a greater amount of consideration.

Ether-Dances

As seen in the last sections, it is obviously not practical to say a command for every movement the stones should make and stack them in this way. It is already easier to say them in one go (if one has already thought about the required moves beforehand). However, it would be even more convenient if one could also include complex constructs in the commands with which the stones would be arranged all by themselves. For this, ether-dances were invented.

Ether-dances are a more complex, but also far more powerful form of ether-batches. Instead of independent chains of commands, ether-dances introduce additional language constructs that allow for more complex command constructs. Ether-Commands are no longer executed strictly from first to last, but can skip commands and even repeat whole sub-groups of commands.

The opening of the ether-channel typically also marks the beginning of an ether-dance, often this is accompanied by specific words or movements that may vary depending on the type of ether-dance. Once the dance has begun, one speaks the commands or verses and makes dance movements to them if necessary. Finally, the closing of the ether-channel marks the end of the ether-dance (again, there are different accompanying movements and/or words from dance to dance). The spells are then (and only then) executed one by one in the Ether and worked through verse by verse until the end of the spoken verses has been reached. The spoken collection of verses and the corresponding dance moves, which is executed in the Ether after the completion of the ether-dance, is also called an ethergram. Because jumps within verses are possible due to the language constructs of ether-dances, it is not so easy to tell when the end of an ethergram has been reached and it has thus been completed. It can even happen that ethergrams, due to the use of infinite loops, are theoretically never finished.

In Syoombraaya, the schematic ether-dance presented in this book, a dance begins with the utterance of the word „Syoombraaya“ and the simultaneous concentration on the connection to the Ether, or the concentration on wanting to perform an ether-dance. With the syoombraaya dance seen in figure 1.8, one could build a stone tower in an instant. This involves moving the stones to the point where the speaker of the

Syoombraaya dance is located, moving them and then dropping them. The dancer can step aside after saying the dance and watch as stones are piled up in the place where them was standing before (it could also happen that one accidentally stones oneself; so the dance should not necessarily be imitated). The language construct used for repetition in this process is called a loop and can be used to execute certain commands over and over again, e.g. to do things for all the things in the clearing where you are. Loops can also be used to create infinite loops that result in ethergrams that don't end.

The exact function of each language construct is the subject of the worldly layers of the following pillars. However, one can already see that ether-dances, while more complex to understand, also offer many more possibilities in interacting with the Ether and, as can be seen in the example, they often even manage with far fewer commands than corresponding solutions with single commands or command-batches. The ether-dance, on the other hand, can use the language constructs of repetition to get tasks done with only few commands, while at the same time improving the comprehensibility of what the ether-dance does, even though the exact operations may be harder to understand and the design of corresponding ether-dances needs more thought.

1.2.6 Energy

Now, one might think that it is far too powerful to be able to access and manipulate arbitrary things in the universe, and this objection would be quite justified. For if that were the case, one could simply access the moon and manipulate it into crashing down on Earth. While this scenario is theoretically possible, it will turn out that there are also limitations with regard to calling and manipulating things in the Ether that make just such things as deliberately crashing the moon into the earth practically impossible. This limitation is commonly referred to as energy, which is also discussed in the medial layer. It means that it takes different amounts of energy to call and interact with different things in the Ether, depending on were they are and how big they are; and all this dependent on the fact, how much energy one has available in the surrounding area. It will be seen that the concept of being able to access all things in the universe is only a theoretical construct, which in practice, due to the limitations imposed by the concept of energy, will be severely limited and it will take some tricks to actually be able to access the majority of things in the universe effectively, that is, so that you can manipulate them effectively with the energy that is available to one.

1.2.7 Relationship to other Layers: Bridge between the Layers

In contrast to the cosmic layer, the Ether in the medial layer is already viewed from the point of view of a living being of the physical world and thus has far more practical relevance. With the knowledge of the medial layer, one can already carry out interactions with the structures of the Ether introduced in the cosmic layer of the pillar.

In contrast to the worldly layer, however, the accesses are still very simple and limited to single ether-interactions. Executing commands as groups is only treated as a possibility, but is not carried out further. In the worldly layer, this possibility is then taken

up and the full potential that arises with group-commands is opened up, while the used ways of calling are only explained as far as it is necessary for the ether-dances.

In summary, the medial layer thus builds on the constructs of the cosmic layer and shows ways in which they can be used practically; this in turn is taken up and further developed in the worldly layer. The medial layer thus represents a bridge between the cosmic and worldly layers.

1.3 Worldly Layer: Ether-Dances

In the medial layer, the Ether is only interacted with through single, independent spells. The worldly layer in contrast is based on the realisation that one can also utter groups of ether-commands, which are then executed one by one by the Ether. In their most complex form, these group-commands are called ether-dances, which enable powerful and flexible interactions with the Ether. Their possibilities and application are the central topic of this layer.

1.3.1 Overview

Ether-batches can already be used to execute groups of commands but ultimately they are just a more convenient way to use single commands. Ether-dances on the other hand actually open up new territory. They extend ether-batches with additional language constructs that can be inserted between commands and thus influence the course of the ethergram e.g. to repeat certain sections or, depending on a condition, to pass through or skip them. This allows complex interactions with the Ether, as seen in Figure 1.9, and can be used to do a variety of things, for example to build a stone tower as in Figure 1.10.

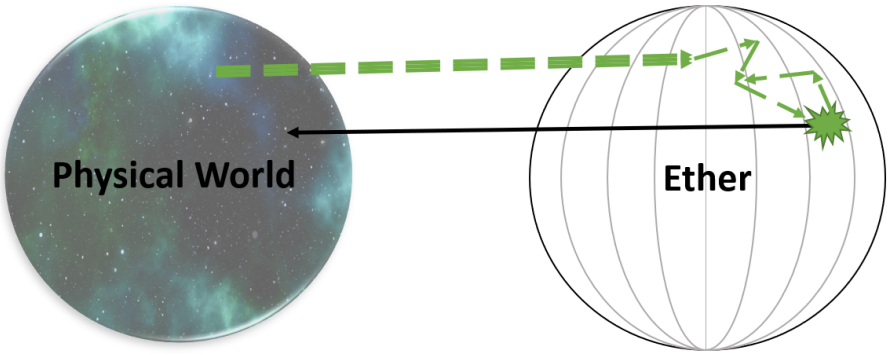

Figure 1.9: Complex interaction with the Ether through ether-dances

Figure 1.10: Application of an ether-dance for the artful arrangement of stones

With that, ether-dances are far more complex than ether-batches. An ether-dance is a sequence of movements and the simultaneous pronunciation of words, which in their entirety are referred to as ether-dance (see figure 1.11). Typically, an ether-dance begins with a particular word, which can also be thought. In Syoombraaya this is „Syoombraaya“. While reciting the individual spells, more or less predetermined movements are performed to structure the spells and as an aid to certain language constructs. These movements are called ether-dance-moves.

1.3.2 Ether-Dance Types

There are different types of ether-dances, all with their own language constructs and dance steps. Most of them have different areas of application, but usually, they share certain types of language constructs.

This book presents the conceptual ether-dance Syoombraaya[1], which contains all the main language constructs of a typical ether-dance and can serve as a first prototype for other ether-dances.

[1] corresponding to the first word of the fictional language „Loxian“, which was created in 2005 by Roma Ryan and used by the singer Enya in several songs

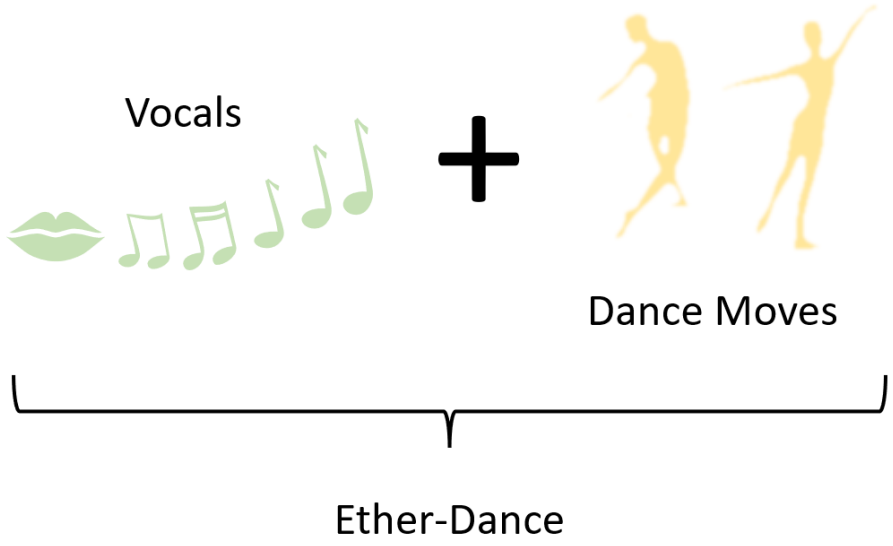

Figure 1.11: Ether-Dance structure

1.3.3 Ether-Dances: Basic Structure and Terminology

To better plan an ether-dance before performing it, the lyrics and dance-moves of an ether-dance can also be described in writing. Such a description of an ether-dance is called an „ether-dance protocol“, „ether-dance rite“, „ether-dance ritual“ or „sourcery text“. Figure 1.12 shows an exemplary ether-dance-protocol of a Syoombraaya dance. It also shows the typical components of an ether-dance.

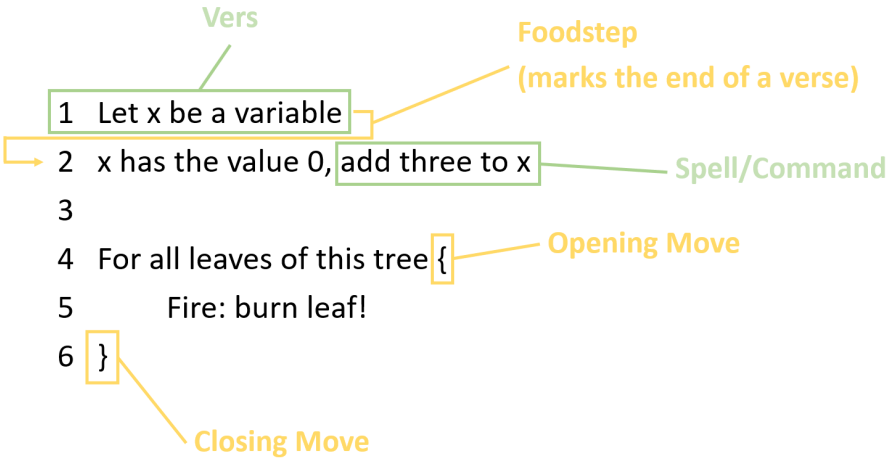

Figure 1.12: Protocol of a Syoombraaya Ether Dance, with its components annotated

An ether-dance consists mainly of the following components:

- Verse
- Spells/Commands
- Dance-Moves

First of all, an ether-dance consists of a series of verses of any length. Each verse in turn can consist of a mixture of spells and dance movements (as already described in the medial layer, a command can also consist of several nested commands). A verse is usually concluded with a specific dance movement, for example with a step.

Dance movements are another innovation introduced with ether-dances. They give structure to an ether-dance, for example by marking the end of a verse, and they are also needed for certain language constructs. Ether-batches can also already contain dance steps; in these, commands that are not to be further nested are mentally separated from each other, which is usually accompanied by a simple movement such as a step, however, these are not mandatory as they are in some types of ether-dances. In Figure 1.13 the dance steps of the Syoombraaya dance are shown and how they are symbolized in ether-dance-protocols.

Dance Moves	Protocol-Symbol	Meaning in Syoombraaya
Footstep	(new Line)	New Verse
Opening Move	{	Open new section
Closing move	}	Close opened section

Figure 1.13: Dance steps of the Syoombraaya ether-dance

1.3.4 Relationship to other Layers: Result/Application of the other Layers

The concepts of the Ether that are introduced in higher layers are treated in the worldly layer only as far as they are necessary for the understanding of the spells and their effects. The cosmic structures only play a secondary role and the rules for calling things are used without dealing with their inner workings. In principle, the other layers are thus only considered through the commands that are made possible by them. At the same time, it implements and practically applies the theoretical potentials of the other layers. It is thus in a certain way the result of the theoretical efforts of the other layers. In this sense, the worldly layer can also be regarded as the actual foundation of the ether-teaching, at least as far as its practical significance and standing in a world with the Ether is concerned. Without the wordly layer, the other layers and the Ether a whole would receive much less attention than with it.

2 Second Pillar: The Arcanes

The second pillar deals with the arcanes, which are the smallest elements of the Ether from which all other ether-structures are built. It thus deals with the Ether at its smallest level. The knowledge about them will allow to perform the first ether-dances, which will already be able to do simple tasks.

2.1 Cosmic Layer

This section describes the ether-model of the second pillar; it is a very simplified view on the Ether and contains only the arcanes as components.

2.1.1 The Arcanes

The smallest parts of the Ether are called „arcanes“. They are unchangeable and indivisible. The complete Ether is composed of them.

Among the arcanes, a distinction is made between three categories:

- Numbers
- Words
- The Truth (true oder not true)

A number can be any number, a word any word and the truth can be either true or not true. So number arcanes include 0, for example, or 0.5 or 500, a word arcane could be „bear“ or „fish“ but also made up words like „arbetweqqqq“. While there are an infinite number of numbers and words, there are only two of the truth-arcane: true and not true.

2.1.2 The Arcane Ether-Model

The ether-model of the first pillar consists only of a huge, disordered collection of arcanes (to be seen in Figure 2.1). It is not the case that each value of each arcane occurs exactly once, i.e. once the 0, once the 1, etc., but all values occur infinitely often. This means that an unlimited number of all values of the arcanes are available.

2.2 Medial Layer

The medial layer of the second pillar deals with the calling and practical use of the arcanes.

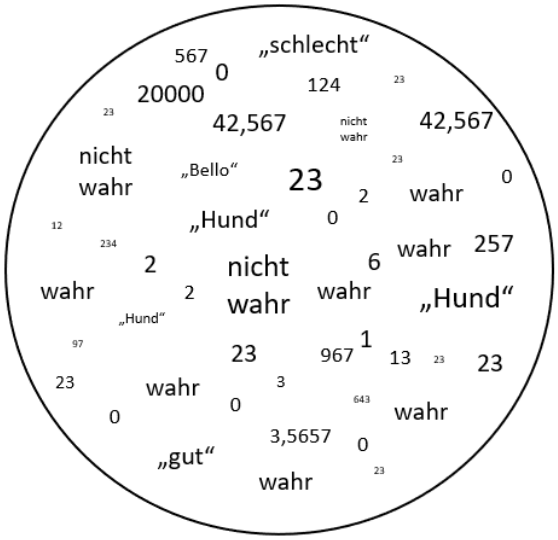

Figure 2.1: Pillar 1 ether-model, called „arcane ether-model“

2.2.1 Access to the Arcanes

In addition to the two concepts of calling, this pillar introduces a new concept that can be used to interact with things in the Ether and that, strictly speaking, cannot be counted as part of the concept of calling. It is called retrieving, recalling or simply: accessing, and refers to the mental retrieval/access of things, usually, arcane values, without conscious use of the Ether.

That means that arcanes do not need to be called: they can be simply recalled by thinking about the desired value. The only thing it takes is an understanding of arcanes, especially words and numbers, which is taught by default at an early age in the schools of most societies. This makes it quite intuitive for most people to recall and use the arcanes without even thinking of interacting with the Ether. For example, one can simply think the number 3 and feel that one does not have to spend any energy on it, and one could also pronounce it without any further obstacles. This makes accessing arcanes very easy and intuitive and usually doesn't even has to be learned, because it already was.

Since all arcanes can be recalled directly, the concept of energy will only be dealt with in the next pillar. It can be assumed that the retrieval of arcane values requires virtually no energy.

2.2.2 Practical use of Arcane Values

Arcane values can be used to do certain operations on them. The tools to do those operations are called „arcanae operators“. There are different operators for different areas of application, for example, for assembling sentences, calculating formulas and setting up statements for calculating truth values. These are not in the Ether as such, but are abstract actions that the Ether makes available to us, but are only present in it

as concepts.

Just as easily as arcane values can be retrieved, arcane operators can also be retrieved. For example, number arcane can be used for calculations: it is very easy to think „6 plus 6“ in your head. It is just as easy to string words together to form sentences of any length, as in this text, for example.

However, unlike the retrieval of arcane values, the evaluation of arcane operators is sometimes complex, which is why not all operators can be performed in the mind. That is, they require a greater amount of energy, which exceeds most people's mental capabilities (mental capabilities are weak accesses to the Ether, where the ether-channel is only slightly open; about like breathing through a blocked nose). In the case of „6 plus 6“, the result is immediately in the mind, but for more complicated calculations such as root calculations, the calculation takes longer, or is not possible at all. Although these abilities can also be trained, at a certain point the limit of mental abilities is reached.

Mental arithmetic is already practised in school, which is the discipline of performing arcane operations using only mental skills, that is, without conscious access to the Ether. While these skills are useful, conscious communication with the Ether allows one far greater possibilities for calculating arcanes. It can be used to calculate even complex operations with virtually no time delay.

To have an arcane operator evaluated in the Ether, one simply executes a command containing the arcane values and the operator to be evaluated. A command is a sequence of words that have a certain structure. You don't necessarily have to understand this structure, in most cases it closely resembles natural language and therefore, it is usually enough to simply say what you want from the Ether.

For example, you say „root of 81“, which would evaluate to 9. However, one would not hear the result 9. That is, one would say „root of 81“ and nothing would happen. Therefore, the following section already introduces a very useful aspect of the air element, which can be used to obtain responses from commands executed in the Ether and produce much more illustrative examples of use.

2.2.3 Receive Responses from Ether-Commands

The arcanes and their operators alone do not allow practical examples of spells and certainly not ether-dances. One could only carry out calculations in one's head, which then would actually be carried out in the Ether, but without any noticable effect to the caller. Because of that, usually, an aspect of the air element is introduced at this point in anticipation, which usually belongs to the introduction of the four elements in the third pillar, that is: the speak-power of the air element. As already described in the first pillar, the air element with the ether-channel has a special relationship to the Ether and is, so to speak, the direct line to it. Through the air travel the words that are processed in the Ether. With the speak-power, the air-element is also able to send back words through the Ether-Channel, which are heard in the mind of the caller. For example, one could say the single command „air: speak out: root of 81“ and would instantly hear „9“ in one's head.

The words that are sent back usually appear in the caller's mind in the form of thoughts, or are perceived as an external voice saying the answer. This can be very confusing and even frightening at first, but is completely harmless.

2.3 Worldly Layer

This section describes the language-constructs of ether-dances and their handling of the arcanes. Language-Constructs will allow to repeat and skip commands, and dynamically change the course of execution of an ether dance. To this end, ways in which ether-dances can give answers and listen to input will first be described, as these will enable better example dances, and then, corresponding language constructs will be presented.

2.3.1 Ether-Dances with Input and Output

This section describes how to create ether-dances with input and output using the speak- and listen-power of the air element.

Speaking Power of the Air Element

As already shown in the last layer, one can receive answers from outspoken ether-commands with the help of the speak-power of the air element. Of course, this power can also be invoked in an ether-dance, which, when processed in the Ether, can send back words to the caller. For example, the result of a number-calculation. This allows for much more practical examples of ether-dances, which is why it is usually anticipated in this pillar rather than in the third one, where the elements are introduced. But the speak-power is not only used in the context of didactic examples but, as will be shown in higher pillars, is the main way for receiving answers from performed ether-dances. It can be quite useful to send messages back to oneself while the ethergram executes, such as the words „hello physical world“ in ether-dance-protocol 2.1. This can be especially helpful when trying to fix errors that occur during an ether-dance.

```
1 Syoombraaya
2
3 Air: say "Hello physical world!"
```

Ether-Dance-Protocol 2.1: Hello-World-Ether-Dance

Listen-Power of the Air Element

In addition to the speak-power of the air element, there is also the possibility, with the help of the listen-power of the air element, to pass arcane values to the ethergram during its execution in order to influence the course of the ethergram. In other words, the listening power of the air element makes it possible to speak to ether-dances during their execution and thereby change their course. Once the ethergram in the Ether reaches

the place where the air element is to listen, the ethergram stops until it hears something from the ether-channel from which it was created. This can be any arcane values, but is usually words or numbers. When the air hears something, it stores it in a variable (these are explained in the next section), and continues. The received input can then be used to dynamically determine the further course of the ethergram. This makes ether-dances much more flexible.

In the ether-dance-protocol 2.2, for example, the ethergram first asks what one's name is. Then it waits until it hears one word and then waits again until it hears a second word, which in this case is supposed to be a person's last name. It then puts the words together and outputs the words „Good day“, followed by the first and last name. This gives you an ethergram that can say hello to any person.

```
1  Syoombraaya
2
3  Air: say "What is your name?".
4
5  Air: say "Please say your first name".
6  Let x be a variable.
7  Air: listen for a word and ban it in x.
8
9  Air: say "Please say your last name".
10 Let y be a variable.
11 Air: listen for a word and ban it in y.
12
13 Air: say "Good day".
14 Air: say x
15 Air: say y
```

Ether-Dance-Protocol 2.2: Good Day Ether-Dance

Even if one does not yet understand some of the language constructs of this example, because they will only be introduced in the next sections, one can already see that the powers of the air element make at least somewhat interesting ether-dances possible. In the next pillars, these will take a less prominent role, because there, with the interaction of ether-objects, more interesting application examples will become possible by themselves; in this pillar, however, much use will be made of them.

2.3.2 Language-Constructs of Ether-Dances

The language constructs of ether-dances are shown using the conceptual ether-dance „Syoombraaya“, which contains all the main language constructs and dance movements of ether-dances. In Figure 2.2 one can see an overview of the these language constructs and, in a mixing of fictional mythologization and concept description accepted at this point for once, their models in nature and mythology through which we might have learned them. These language constructs will be presented in the next sections.

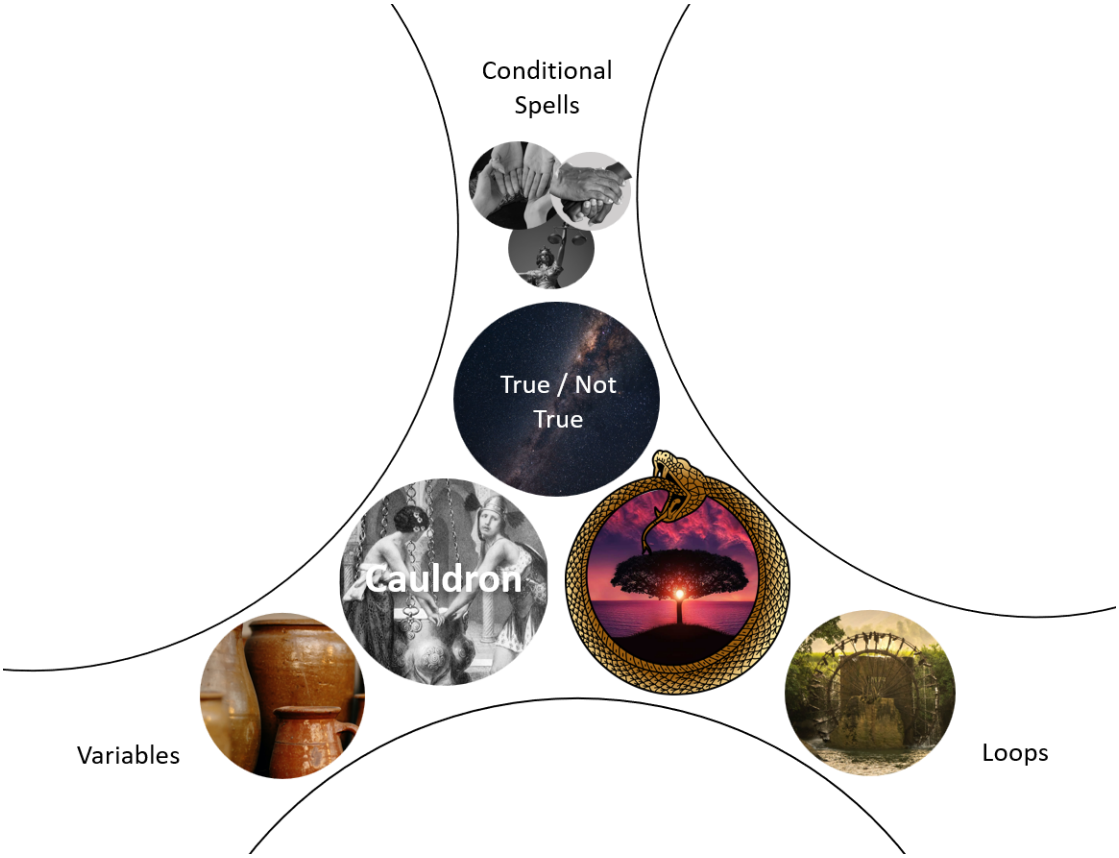

Figure 2.2: Most important Syoombraaya language constructs (with fictional mythologization)

Variables

Variables are like containers into which one can put arcane values. One can take them out just like real containers and replace them with other arcane values. One can also check which value is currently in a certain container.

In the last section on the listen-power of the air element it was already seen that variables can be used to store arcane values obtained via the listen-power of the air element and thus make them usable for the ethergram. Otherwise, the value would be heard by the ethergram, but simply fade away in the Ether, unused. This already makes variables enormously useful, as values can be stored in them dynamically and do not yet have to be fixed when dancing the ether-dance. For example, in ether-dance-protocol 2.3

the number five is added to any number heard. Without variables and the listen-power, you would have to do a different dance for each number you want to add to five. Instead, here you have a dance for any number.

```
Syoombraaya

Let x be a variable.

Air: listen to x.

x plus 5.

Air: speak x.
```

Ether-Dance-Protocol 2.3: Ether-dance adding five to any number heard

Even beyond this use case, variables can be used to temporarily store arcane values in them. In ether-dance-protocol 2.4 you can see the same example without using the listening power. In this case, one would not necessarily need the variable and could also simply pronounce the arcane values in the ether-dance at the required position instead of storing them in a variable beforehand, but this sometimes makes it easier to know what one is using the values for and the ether-dance becomes more comprehensible. For example, instead of the arcane „3,141“ at the very beginning of the ether-dance, you could create a variable called Pi and ban the arcane value in it. At every point where you now wants to use Pi, you would not have to recite the long decimal, but now could simply use the variable Pi and would immediately know that the long decimal is meant.

```
Syoombraaya

Let x be a variable with value 0.

x plus 5.

Air: speak x.
```

Ether-Dance-Protocol 2.4: Ether-dance pronouncing „five“

This example shows a third advantage of variables: they make it possible to give new names to things that are stored in them. Without variables, you only have fixed arcane values that you can use in the ether-dance. With variables, you can name the values as you like and make the dance easier to understand. For example, the „good-day“ example of the last section could be made much more understandable by renaming the variables as seen in ether-dance-protocol 2.5.

```
Syoombraaya

Air: say "What is your name?".
```

```
4
5 Air: say "Please say your first name".
6 Let firstName be a variable.
7 Air: listen for a word and ban it in firstName.
8
9 Air: say "Please say your last name".
10 Let lastName be a variable.
11 Air: listen for a word and ban it in lastName.
12
13 Air: say "Good day".
14 Air: say firstName.
15 Air: say lastName.
```

Ether-Dance-Protocol 2.5: Good day ether-dance

Conditional Statement / Conditional Commands / Branching

If you want to depend the course of an ether-dance on certain conditions, for example regarding the content of a variable, you can use the language construct of branching. In the case of a branch, one says a statement, which is, when the ethergram is executed, checked and, based on this, is decided, which of two paths the ethergram should take. For example, one could check if a variable contains the arcane 2 and then the ethergram would take path A, otherwise it would take path B. In the ether-dance, these two paths can be easily defined. If the statement is true, the ethergram will simply execute the next command. If the statement is false, the ethergram will continue after the „otherwise“ language construct and skip the other commands.

For example, in the ether-dance in protocol 2.6, the branch is used to decide whether the heard number is less than five or not.

```
1 Syoombraaya
2
3 Let x be a variable.
4 Air: hear number and ban in x.
5
6 If x is less than five {
7     Air: say "Number less than five."
8 }
9 Otherwise {
10     Air: say "Number greater than or equal to five."
11 }
```

Ether-Dance-Protocol 2.6: Number Less Than Five?-ether-dance

This example also show the use of parenthesis for the first time, which symbolise the dance movements that are done at the respective points of the ether-dance. In the case of the opening parenthesis, this is an opening movement; in the case of a closing bracket, it is a closing movement. In the language construct of branching, the dance moves serve to mark the two different areas to which the ethergram can jump after evaluating the statement. The meaning of dance moves in general is explained in more detail in section 2.3.3.

Loop

The loop is the most powerful language construct of ether-dances, as it enables the arbitrary repetition of commands. With its help, huge amounts of spells can be described with just a few verses. Before each run, a condition can be checked to decide whether the loop should end or continue.

For example, in ether-dance-protocol 2.7, a loop is used to pronounce all the number arcanes from 1 to 100.

```
1 Syoombraaya
2
3 Let counter be a variable.
4 counter contains the number 1.
5
6 While counter is less than 101 {
7     Air: speak counter.
8     Add 1 to counter.
9 }
```

Ether-Dance-Protocol 2.7: Counting-ether-dance

It is now easy to see that one could just as well create a loop that does not stop at all. Such a loop is called an infinite loop. Thus, with the introduction of loops, it is for the first time possible that something one sends to the Ether will never terminate and could be, theoretically, continuously executed in the Ether without ever stopping (practically, these loops are usually manually stopped). Infinity loops can be created quickly by accident and are sometimes even intentional, but have no immediate negative impact on the person performing the dance. Since the energy is drawn from the environment, one does not have to worry about losing energy from one's own body or anything like that. Moreover, it is easy to abort an ethergram in its execution: whenever an ethergram is started to be executed, an already running ethergram is aborted. There can always only be one ethergram being executed per ether-channel at a time. Thus, there are

no immediate dangers from infinite loops, or at least no more than from the language construct of the loop in general, which is obviously very powerful and should therefore be used with caution.

At this point, one can see that variables also become much more powerful in the context of loops, as they can be used to store the progress of the loop. Based on this progress, it was possible in the example to find out when the loop should be terminated, namely when the counter has been counted up often enough.

Loops will become more important in the course of the following pillars, because, with the introduction of objects, they can be used to execute commands for large sets of things, for example, to throw all the trees in a forest into the air.

2.3.3 Dance Movements and Visibility of Variables

Dance moves have already been introduced along with the language constructs presented in the last section. However, this section will go into more detail about their role and what they are needed for in the first place.

The main role of dance moves is to allow further structuring of commands, which is needed for certain language constructs. For example, in the case of branches, they divide the ether-commands into the two parts according to the condition of the branch, which are visited depending on whether the condition is true or not. They have the same role in loops. Without dance moves, specific words would be needed, for example in the case of branches to separate the separate sections, which could become confusing (although this is also done in some, especially older, ether-dances, which aren't dances in the strict sense of the word, because there lack of actual dance moves, but are still considered as such). In other types of ether-dance, other dance movements exist that enable other language constructs, but in all ether-dances, dance moves (or something similar) are needed to enable language constructs beyond simple commands. An ether-dance without dance moves (or something similar) is not an ether-dance but an ether-batch.

Dance movements also have an influence on the variables that are created in them. Variables exist only within the section of the opening and closing dance movements in which they are created and all sections contained within them. That is, if a variable was created within the dance movements of a branch, it ceases to exist after the section of the branch. If the variable was created before the branch, it will exist within the section of the branch and also afterwards. This behaviour of variables is called visibility or existence of variables and is not continued here further.

2.3.4 Methods

This section deals with the grouping of ether-commands into methods. Methods can be used to bundle chains of commands that are used frequently together, being able to use them with only one method-command.

For example, in ether-dance-protocol 2.8, a method is created which bundles the commands of the „Good Day"-ether-dance into one method. One could now call this method at any point in the dance and when the execution of the ethergram reaches this point,

the execution jumps to the definition of the method and executes the „Good day“-ether-dance and then jumps back to the point where the method was called. In this way, it is very easy to use chains of commands that are used frequently without having to constantly recite all of them.

```
Syoombraaya

Create method GoodDay {
    Air: say "What is your name?".

    Air: say "Please say your first name.".
    Let x be a variable.
    Air: listen to word and ban it into x.

    Air: say "Please say your last name.".
    Let y be a variable.
    Air: hear word and ban it in y.

    Air: say "Good day".
    Air: say x
    Air: say y
}

Call Method GoodDay
Call Method GoodDay
```

Ether-Dance-Protocol 2.8: Method for saying good day

However, methods can do much more than to server as placeholder for chains of commands. They can have inputs and outputs, much like ether-dances. In ether-dance-protocol 2.9, the method can receive two numbers and then calculate their sum. It can be seen that methods can be passed arcanes and, if they store them in variables via listening commands, can use them in the method. Finally, methods can also return an arcane as the result of execution with the speak command. In this case, the result of the method is spoken via the air element, however, the speak- and listening-commands within the methods stand in no relation to the air-element, but are internal key-words of the specific ether-dance, which certainly have been inspired by the powers of the air-element, but were merely chosen by the creator of the ether-dance, because it felt natural. In other ether-dances, the passing and receiving of arcanes to methods might be handled differently, for example, with the use of dance moves.

```
Syoombraaya

Create method Add {
   Hear number 1 and ban in variable number1.
   Hear number 2 and ban in variable number2.

   Say number1 plus number2.
}
```

```
9
10 Air: say call method Add with 5 and 3.
```

Ether-Dance-Protocol 2.9: Method for adding two numbers

2.3.5 Outlook

The language constructs presented in this pillar already represent the essential language constructs of ether-dances. Nevertheless, it is important to understand the other pillars as well in order to realise the full potential of ether-dances. Accessing new constructs through the medial layer requires the introduction of new language constructs for ether-dances in each pillar in order to deal with these new constructs. The handling of more powerful ether-constructs will then also increase the possibilities through ether-dances, thus enabling more complex, but also more interesting dances.

Therefore, all that is needed at this point is a little patience. Already in the next pillar, it will be possible to make the trees of a clearing fly into the air.

3 Third Pillar: Ether-Objects and the Four Elements

In the third pillar, structures emerge from the disordered mass of the ether, which can be called through their physical representations: the ether-objects. The following sections deal with these objects and their nature; they also contain descriptions about the four elements with which these objects can be manipulated and thus also influence the physical world.

The new ether-model will make it possible to query the properties of things, for example the age of a tree, the colour of a sheep's fur or the weight of a stone. This will enable more complex ether-dances, for example, it will be possible to output whether a fruit that one touches is bad or not. With the help of the elements, ether-objects can also be manipulated, for example to make stones float in mid-air.

3.1 Cosmic Layer

This section describes the third pillar's ether-model, in which ether-objects are introduced. It also clarifies whether the four elements appear in this model, and if so, how.

3.1.1 Objects

Ether-objects can be thought of as the representations of all things in the universe through arcanes.

For example, in Figure 3.1 you see a representation of the ether-object of the wolf Nua. On the right side one sees the ether-object and on the left side its representation in the physical world, and in between, the constant interrelation in which they are. One can see that on the left side the she-wolf is present as one perceives her as a human being and on the right side only in the form of arcane values arranged according to certain properties of the she-wolf.

3.1.2 The Object-Oriented Ether-Model

In the ether-model of the second pillar, the arcane values in the Ether are arranged in structures called ether-objects, hence the name: object-oriented ether-model. An ether-object consists of a series of arcane values, each of which has a name. These are the properties of the object. For example, in Figure 3.1, „age“ is the name of a property and the associated value is 8. The object in itself has no name and can only be clearly determined through its connection to its physical representation.

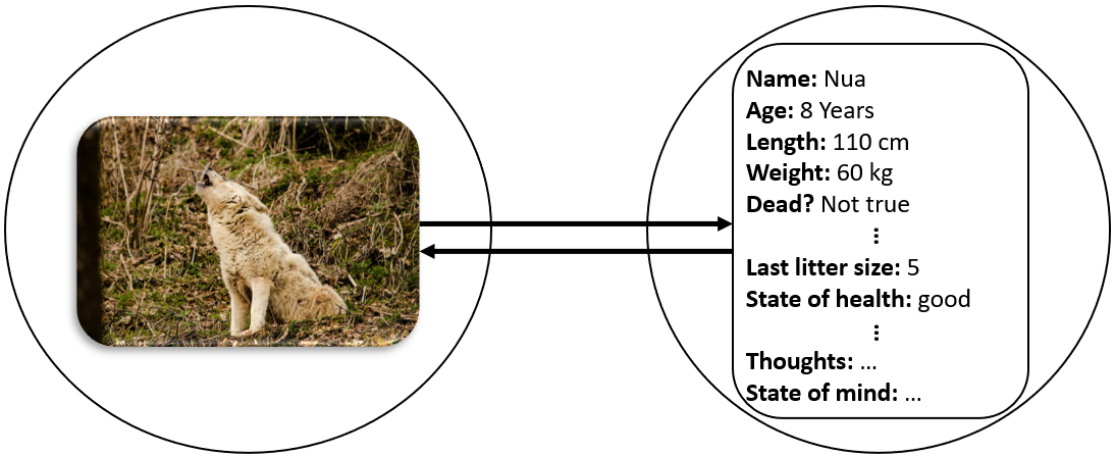

Figure 3.1: Ether-Object-Diagram of the wolf Nua

Every ether-object has such a connection to something in the physical world, through which (and only through which) it can be identified (at least in this and the next pillar), and every thing in the world also has a corresponding ether-object. As seen in Figure 3.3, in a clearing with three trees, each of the trees would have a unique connection to an ether-object, and each of the ether-objects would have exactly one connection to its physical representation. The same would apply to all the other plants in the clearing, as well as the ground, the insects and the observer themself. This means that with none of the information in the ether-object can one clearly distinguish the object from other ether-objects; this is only possible via the connection to the representation of the physical world.

An ether-object thus becomes useless if one does not know its physical representation. If one does not have a picture of the tree in figure 3.3, one could not find out with certainty from the ether-object alone to which physical thing it belongs. The connection to their physical representations will become important in the medial layer for calling ether-objects.

Are Arcanes Ether-Objects?

A few final notes: arcanes are not ether-objects. Ether-objects are structures of arcanes, while arcanes do not structure anything; they just are.

Empty Ether-Objects

It should also be clarified whether there are empty ether-objects, i.e. ether-objects without properties.

To be clear: they do exist, but not naturally. That is, there are no natural empty ether-objects, or at least none have been found yet (to say that arcanes are empty ether-objects would be wrong, they are something else, they are fixed values, ether-objects are

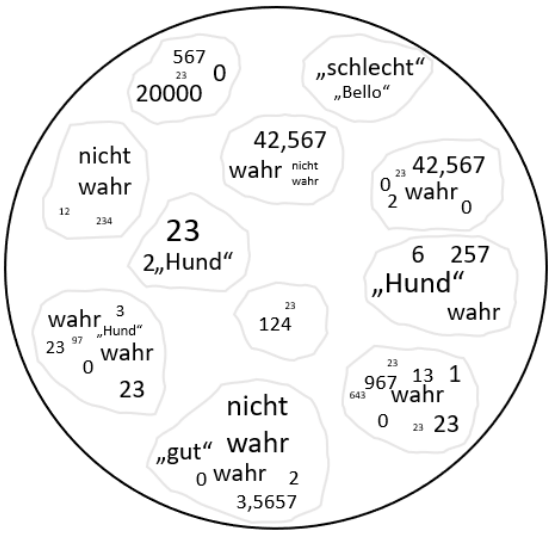

Figure 3.2: Pillar 2 ether-model, „object-oriented Ether-Model“

structures). They would also have to be connected to things in the physical world without properties, which has not yet been found and probably could not be found because the objects would have no information to access. So even if there were empty ether-objects, they would be practically irrelevant. However, as described in section 3.3.3 of this pillar, they can be created to exist temporarily in the Ether in the form of artificial ether-objects, which then can be endowed with properties and even arcane values to become objects that behave and can be accessed just like natural objects.

World-View of the Flat Ether

And one last thing concerning ether-objects: the world view learned in this pillar is often described as „world view of the flat ether“, because at this level of knowledge all ether-objects are on one level and there are no hierarchies, which are only introduced with the next but one pillar.

3.1.3 Object-diagrams

There are different ways to visualize ether-objects. The type of visualization we have seen so far is called an ether-object-diagram. In it, ether-objects are represented with rectangles and their properties are written into these rectangles. An ether-object-diagram can contain a single ether-object, but also of several ether-objects.

In Figure 3.4 an exemplary ether-object-diagram is shown. It could be surprising that the ether-objects in the object-diagram have been given names, although they do not actually have unique names. In fact, however, this is common practice and is done in order to be able to talk better about the objects under consideration. It is easier to talk about objects you want to interact with if they have names, rather than drawing

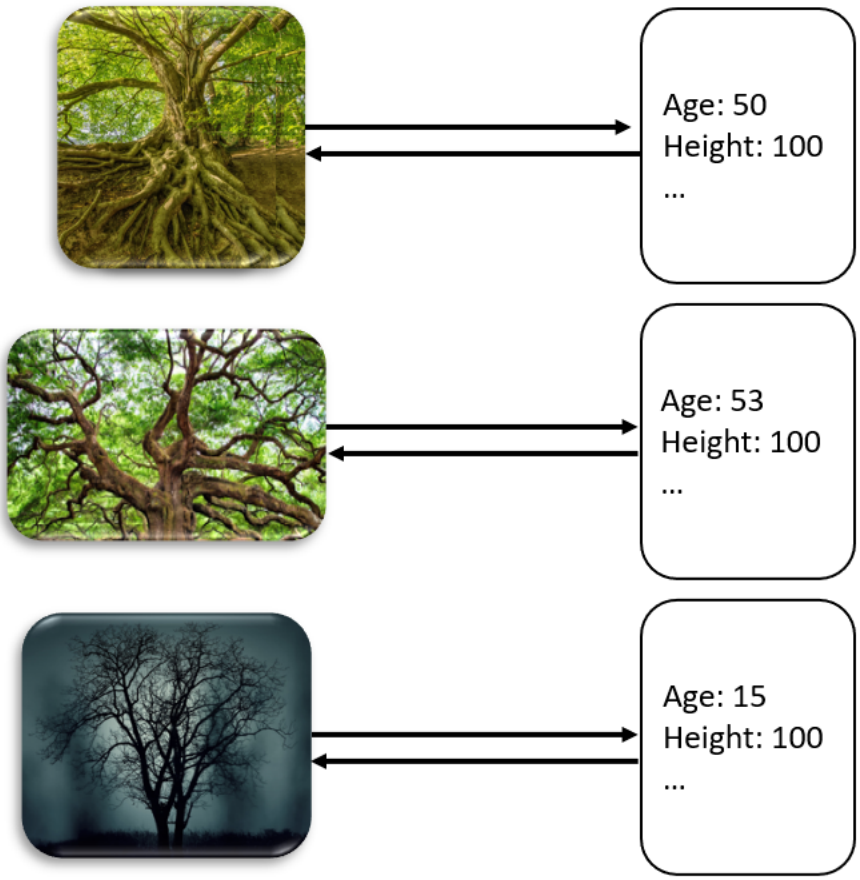

Figure 3.3: Tree ether-objects and their properties

pictures of their physical representations and referring to them when one wants to talk about the object in question, as in previous illustrations. Nevertheless, this should not hide the fact that ether-objects do not have unique names and a connection to their representation in the physical world is needed to identify them. It could, for example, also be the case that an object has several names and for this reason alone cannot be described unambiguously by its name. One should therefore always keep this in mind: It is not possible to call ether-objects directly via these names, names in object-diagrams only serve illustrative purposes. Therefore, it is more or less irrelevant how the objects are named, as long as one knows which physical thing is meant by it.

Representation of the Properties

Since there are potentially an infinite number of properties of an ether-object, typically only those are shown which are relevant in the context for which the diagram is drawn, or those which seem most relevant in general. The latter criterion is also influenced by how much energy they cost, which is described in more detail in section 3.2.6.

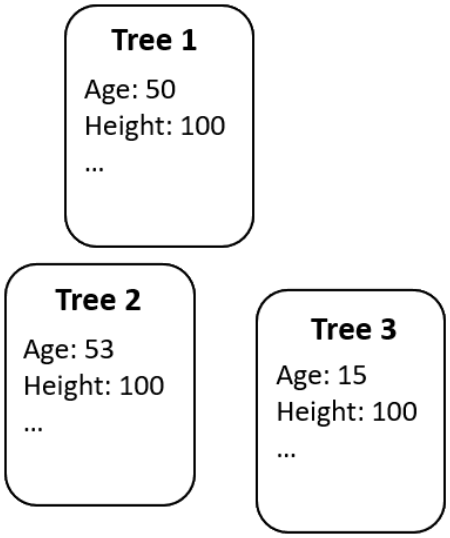

Figure 3.4: Example of an ether-object-diagram

3.1.4 The Four Elements

The four elements are the instances behind every movement in the universe. Every change in the Ether and in the physical world emanates from them. They thus set the course for the path of the universe; since the powers of the four elements can also be used by living beings and chance also plays a role in their work, a world with the Ether is, however, never fatalistic.

The four elements are the central tools for manipulating ether-objects. They are typically referred to as fire, water, earth and air.

The Powers of the Elements

Each element has certain forms of occurrence, through which it acts in the worlds, called powers. They are listed by category in Figure 3.5. It should be added that the force „dissolve“ means to bring something closer to the state of aggregation gaseous. It can also be seen that the hearing and speaking powers of the air element, which were already learned about in the last pillar, are only one of many powers that an element can possess; and that the mentioned powers are so-called special powers to which there is no counterpart or overlap in the other elements and thus can only be influenced/controlled with the respective element. The special powers of the other elements are explained in more detail in the following sections.

	Change of aggregate state	Creational Powers	Change of Weight	Special Powers
	Turn into fire, burn, heat, warm	Extinguish		Shine (emit light), darken (absorb light)
	Turn into water, liquify, cool, ice, turn into ice			Soul / Thoughts / Psyche / Mental State
	Turn into stone, solidify, structurally harden, structurally weaken, disintegrate, turn into earth	Create	Make heavier	Apply force
	Turn into air, dissolve		Make lighter	Listen, speak, levitate, fly

Figure 3.5: Overview of the powers of the elements

Position in the Object-Ether-Model

The four elements have a special position in the structure of the universe, since they belong to both the Ether and the physical world, but are neither ether-objects nor physical things, nor both. That is, they exist in their form simultaneously in the two worlds without any separation. They are the same in both worlds, there are no representations as such, there are only the elements themselves. So, in this pillar, the four elements are considered as being part of the Ether, but not as ether-objects but as the elements themself and thus stand somehow separate from the structures discussed in this pillar.

This does not need to be fully understood in this pillar (it will be further elaborated in pillar 5); what is essential to know is only that the elements can be found in the physical world and in the Ether, but are not everywhere like the arcanes and therefore need to be accessed through their specific occurrences in the Ether/physical world. This will become important in the sections about calling the elements in the medial layer of this pillar.

3.2 Medial Layer

In this pillar, calling of and interacting with ether-objects becomes possible and thus, for the first time, also the indirect manipulation of things in the physical world.

For the purpose of calling the ether-representation of a thing in the physical world, the simplest possibility to do so is described in this pillar: via calling by touch, called „direct calling“. It makes it possible to retrieve information from an ether-object and manipulate the called object. However, the latter also requires access to one of the elements; therefore, calling the elements is described next and then how to manipulate called ether-objects with their help. Afterwards, the concept of energy is discussed, which imposes its natural limits on the concept of calling in general and subsequent interaction.

3.2.1 Calling Ether-Objects via Touch („Direct Calling“)

One cannot access ether-object directly like the arcanes, but there is the concept of calling, with which to create a connection to things one wants to access and interact with. There are two types of calling; one is called „direct calling“ and works by touching the object one wants to interact with. It is the central form of calling in this and the next pillar. The second form of calling, by which all things in the universe can be reached without having to touch their physical representations, is introduced in the fifth pillar.

One might wonder why a touch is needed at all and why one cannot call it „just like that“ by imagining it, like retrieving arcanae values. This is because, as discussed in the last pillar, ether-objects themselves do not have a unique name. They are only uniquely identified by their connection to their physical representation and therefore the object can only be called via this representation (at least in this pillar). Therefore, touching the physical thing is necessary to be able to call its ether-object.

This means in practice that if one touches a thing, one has direct mental access to the representation of that thing in the ether and its associated arcane values. For example, if you want to call one of the trees in the clearing you are in, you have to go to the tree and touch it. If you touch the wrong tree, you will not be able to access the desired tree as shown in Figure 3.6. In principle, it does not matter which word you use to call the tree, it is mainly the mental process that matters, meaning which tree you concentrate on.

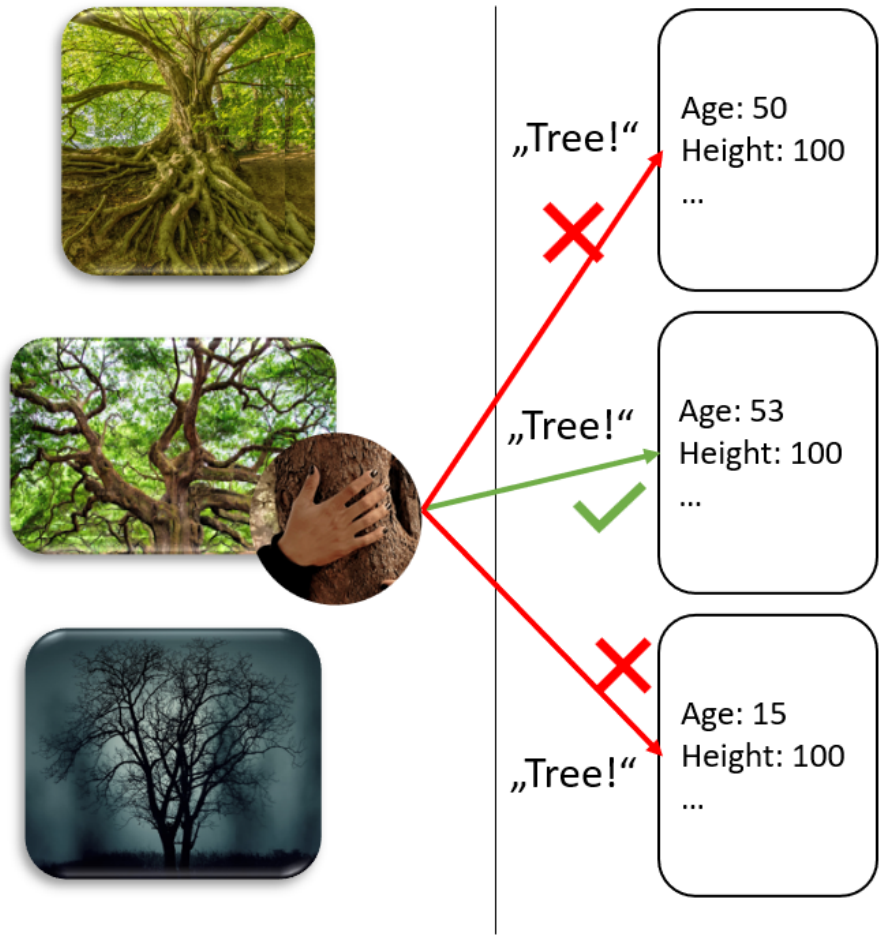

Figure 3.6: Successful and failed attempts to directly call three trees

The Different Calling-Possibilities When Touching A Thing

As simple as this principle may sound, one quickly realizes that the actual practice is a little more complicated. For example, one could ask whether, in the case of a tree, if one touches the trunk, one is calling the trunk or the tree. Although in this example the answer might be unambiguous, in the same sense one could ask whether, when one touches an apple, one is calling the apple or the tree. The answer here is that both are possible, that is, that one could either call the apple or the tree, depending on which one focuses. This means that is not clear per se, which ether-object one is calling when one touches a thing. In fact, there are many possible objects, which can be called from the forest, in which the tree stands, up to the whole universe, or down to the atoms at the spots of the trunk one touches. This complex mechanism is explained in more detail in Pillar 5, as the concept of spheres is needed to understand it; for now its enough to understand that one needs to concentrate on the thing, which one wants to call,

otherwise you will not succeed in calling it.

There are many mental strategies to focus on the „right“ object, meaning the object, which one wants to call. Many people use certain words when calling a thing, e.g. „Bello“ in the case of their dog, or more general names such as „Tree“ if they do not yet know the name of a tree. This has something to do with the said concentration, as this can help one to focus on what one wants to call and not, because one is thinking of something else by mistake, call the wrong thing; however, if one is able to do this, one can also just think it and call the thing exclusively in thought. Thus, this does not change the fact that things in the Ether do not have unique names. The names that people pronounce only help them to call the right things (more on this in the next section). If you are already touching an object, it should not be difficult to focus on it anyway.

The above also does not mean that you can call multiple objects at the same time. The fact still holds that one can only call one object at a time.

However, it is not always pleasant to touch an object, sometimes one cannot stay long in the presence of certain objects and would not have time to perform a spell. Fortunately, there is a trick for such cases: it is also enough to have only a piece of something to call directly, for example a piece of the bark of a tree. However, this connection wears off over time and access becomes increasingly difficult, until at some point it becomes practically useless. More on this in the section on the concept of energy (section 3.2.6) and in the fifth pillar.

Calling things by their Given Names

It has often been written that things have no name. What is meant by this is that they do not have a unique name; it already can happen that in a school class two children have the same name. Things do not have a unique name by which they can be distinguished from all other things in the universe. This kind of name is called a „unique name“ and does not exist in the ether.

Of course things can still have names given by humans, just as humans usually have a name, or even animals. When names are written about in the following, this kind of name is always meant, which refers to the name-property of the object being talked about. Another word for them is „real names“. The question is whether these names can be used to refer to things. Of course they can. Since the name is just one of many, it can be used like any other word to call the thing. It is only important to know that the name is not responsible for calling a thing, but by touching the thing and mentally connecting with it. The fact that, for example, a dog of a family has a name does not matter for the Ether more than any of the other properties of the dog. It may have practical merits, because it makes it easier to concentrate on the dog and thus easier to call it. But you could call the dog just as well by simply saying „dog“ when you touch it and concentrate on it. This would work just as well even if the dog does not have a name. Given names therefore have almost no practical relevance in the context of communicating with the Ether.

Let's go back to the example of the trees in the clearing. The people who live in the area have given the three trees names over the years: Birk, Vi and Gra. Over time,

these names have also been formed as properties in the respective ether-objects. Now one would like to call the tree Birk. One could now say: „tree“, one could also say: „Birk“, if one associates the tree with this name. It doesn't really matter as long as one is clear about which of the trees one wants to call. But this can be easier if an object has a given name.

In Figure 3.7 one can see the same diagram as in the last section, except that now, instead of generic names, the names of the trees are used to call the trees. As described, this way of calling makes no difference to calling in Figure 3.6, except that it may be easier for the person, which does the calling, to focus on the tree to be called.

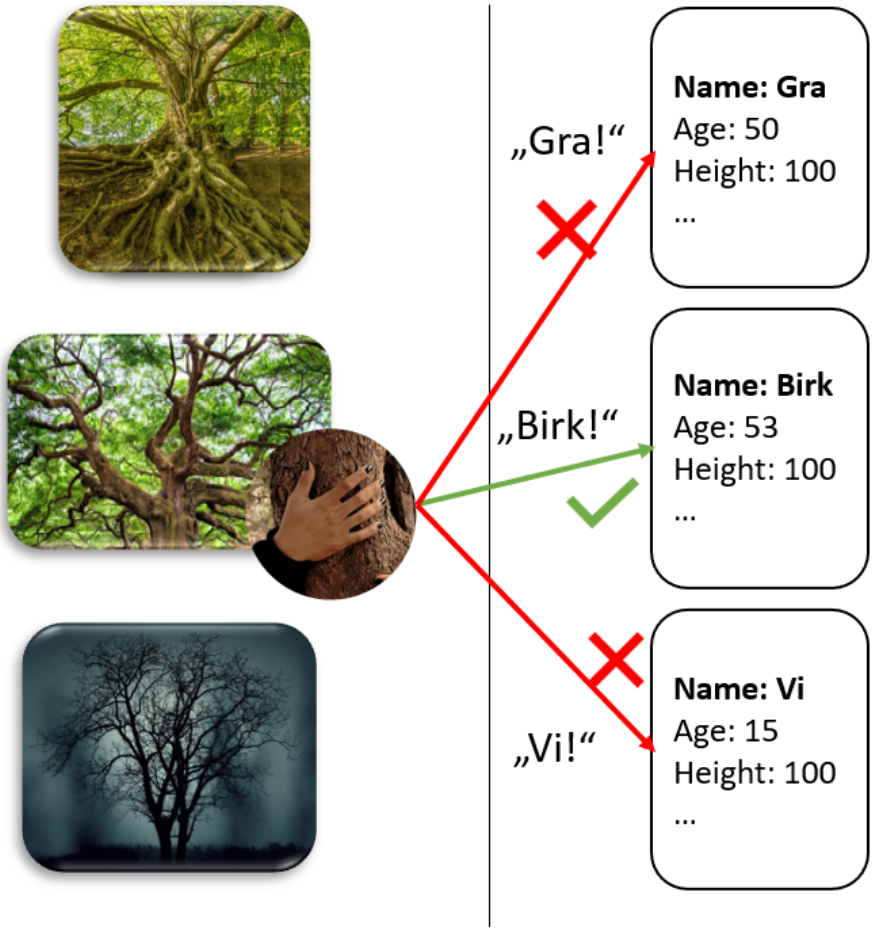

Figure 3.7: Calling trees by their common names

3.2.2 Interaction with Ether-Objects: Access Arcanes (Retrieval of Information)

Once you have established a connection to an ether-object by calling it, you can interact with the object, e.g. to access its arcanes and thereby retrieve information from the object. This is done in exactly the same way as you can access arcanes in memory: you simply think about it. For example in the clearing from Figure 3.6, when touching the second tree, you could simply say „tree“ and call its image into your mind. One could then recall all the property values of the tree when thinking of their names, e.g. the size of the tree. Then one would have the arcane value of the size in mind. This interaction with ether-objects is called information retrieval or information access, and happens passively, meaning that the ether-object from which the information is retrieved is not affected by the interaction.

However, not all information can be retrieved so easily, some requires more concentration and there is even some information that is impossible to reach with human concentration (this has something to do with the concept of energy, which will be discussed in section 3.2.6). It is, for example, very easy to get the information about the name of the tree. Finding out its height already requires more concentration, but this is also relatively easy to do. If you are looking for the way out of a forest, you could find out the highest tree in the clearing in this way, and then climb it to find out the direction leading out of the forest. Getting information about how many atoms it contains, on the other hand, would be almost impossible to retrieve. More on this in section 3.2.6.

3.2.3 Calling the Four Elements and Accessing Their Powers

The four elements are not ether-objects and the means to call them must therefore be discussed separately from calling ether-objects. But as it will turn out, they can still be called easily via direct calling due to their unique characteristics.

Calling the Four Elements Through Direct Calling

The four elements have a special position in the two world-teaching. They are the same in the Ether and in the physical world. This means that if one touches an element one at the same time touches something in the Ether. Through this phenomenon they can be easily called through their physical manifestations.

By touching or being very close to them, one can access them, just as one would do with ether-objects that one wants to call. Each of the four elements can be found in the physical world either in their pure form (like water, air or earth) or in things associated with them (burning things, wet things). The element fire is typically found in burning things. The element earth is found in the earth. The element air is found in the air. The element water is found in the water or in wet things. If one has, for example, one foot in water, one could say „Water!“ to call the element water. Because they are the same in both worlds, there is no difference between calling the physical thing and the ether-representation and therefore calling them requires almost no concentration. Only

when they are being part of or are attached to physical things, is it essential to focus on calling the element and not the thing it is attached to.

As already discussed, each element also has powers. Once one has called an element, one can also directly access their powers. Both commands for calling and accessing the power of an element can, as also already described, be combined into one command. For example: „Water: liquefy!“, to turn an ice wall into water and thus find the way out of an ice cave.

The Ether-Channel and the Air Element

In the cosmic layer and the last sections we have now learned how calling the elements works and what special status they have in the object-oriented ether-model. Furthermore, we already know from the last pillar that hearing and speaking are parts of the powers of the air element. The ether-channel is thus also a part of the air element and it now makes sense that one can communicate with the Ether just like that: because we are speaking directly into the Ether, so to speak. Since one is (usually) surrounded by air, one can communicate with the Ether through it basically everywhere.

Access to the Ether is thus possible for every living being through the air that surrounds our bodies. We speak through it and can move in it. By means of concentration, words as well as movements can be transmitted to the Ether. As long as one is surrounded by air, one has access to the ether-channel. If one is completely surrounded by water, for example, one cannot communicate with the Ether.

3.2.4 Interaction with Ether-Objects: Active Manipulation

The second type of interaction with ether-objects is active manipulation, which can only be done with the help of the elements. It is not possible to change property values of ether-objects directly and, for example, to change the size of a tree. Instead, one uses the power of an element to exert it on an ether-object and thereby indirectly change its values.

Unlike information retrieval, active manipulation thus requires two things: a connection to an ether-object through calling, and the access to the powers of an element. Both points were covered in the last two sections and can now be used together to apply the power of the called element to the called thing. This can be done simply by combining the two commands to call the thing and apply the element's power.

For example, the air element can be used to make an object fly, such as a tree, by saying: „air: make this tree fly!“. If one at the same time touches both things (element and object), this command works. Through the interaction of the worlds, the effects of the power are then immediately transferred to the physical things. Since the roots are torn out for this, it will also have an effect on the values of other objects. The use of elements can thus always be understood as the use of physical force and the consequences of the use of a force are not always exactly predictable.

An example can be seen in Figure 3.8. In this one you are standing in front of a tree with three apples and now you want to drop the fresh apple from the tree without

picking it, as you are too lazy to do so. The idea is that you touch the apple loosely and say an ether-command to drop the apple from the tree into your hand without pulling on it.

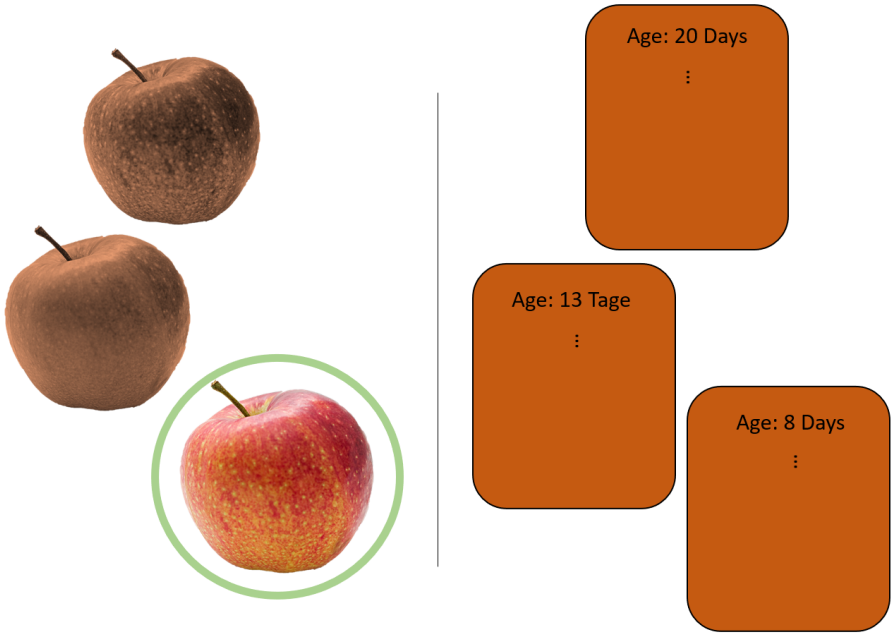

Figure 3.8: Ether-Objects of a tree and several apples

For this, vertical force would have to be exerted on the apple, for which the element earth can be used. Since you stand with your bare feet on the floor, you have direct access to it. So now you can call the apple (by holding the apple in your hand you can call it directly) and then use the earth element to exert vertical force on the apple by saying: „Earth: exert vertical force on this apple!“. Now the apple would come off the branch and fall into the palm of your hand.

If one follows this example, one might notice that the apple falls into one's hand seemingly simultaneously with the completion of the command. In the moment one finishes speaking the command (thus closing the ether-channel) one will feel the apple fall off. This is because the effects that result from a manipulation are always instantly passed on to the physical things. One can virtually assume that the manipulation happens in both worlds at the same time, since the passing on of the changes happens instantaneously and cannot be distinguished from an actual (i.e. physically direct) manipulation, apart from the fact that they are obviously artificially induced.

3.2.5 Practical use of Ether-Object-Diagrams

Ether-object-diagrams were introduced in the cosmic layer without explaining what they are used for. Actually, they are quite useful as they can be used to illustrate ether-objects

with which one wishes to interact. This can make it easier in a given situation to plan the ether-dance or ether-commands to achieve the thing one wants to achieve with the objects. In the last sections and in the following sections, object-diagrams are used again and again to illustrate situations that would be much more difficult to understand without object-diagrams.

3.2.6 Energy

Now, the possibilities discussed would be far too powerful if they could be applied in every conceivable way. For example, you could touch a house and make it fly up in the air to transport it to another place. The reason why such a thing is not possible, or at least not possible in every environment, can be explained by the concept of energy.

The Concept of Energy and its natural Occurrence

The concept of energy states that the calling of things and the interaction with them, as well as the working of powers of elements, costs energy, which is only available in limited amounts.

Energy is located in the environment and is drawn from the Ether to carry out commands or ethergrams. There are places of high energy and places of low energy. Figure 3.9 shows an example of an energy map that can be used to identify places where there is a lot of energy available to communicate with the ether. Villages are marked as circles.

One can see that in places that are heavily inhabited, that is, where humankind has changed a lot, the energy is lower than in places where nature is still untouched, for example in the forest, in the mountains or by the sea. This means that in a city it would probably not be possible to make a building or a large stone take off because it would require too much energy; but in a forest it might be possible. More precisely, it would also be possible in a city, but the command might not be finished then, as will be described in the next section.

As a rule of thumb, the bigger the thing, the more energy it takes and, the farther away you are from a thing, the more energy it takes to interact with it. Therefore, it takes more energy if you only have a proxy of the actual thing, for example the crumb of the wall of a castle. There is a connection between the crumb and the castle, but it diminishes over time and needs a lot of energy in itself. Thus, direct access is always better, but for free dancing, proxy objects are often more comfortable. In these cases, however, a physical proximity to the object should still be maintained.

The fact that the execution of ethergrams is fueled by the energy of the environment is also why it is not necessarily a bad thing for the speaker of an ether-dance if it

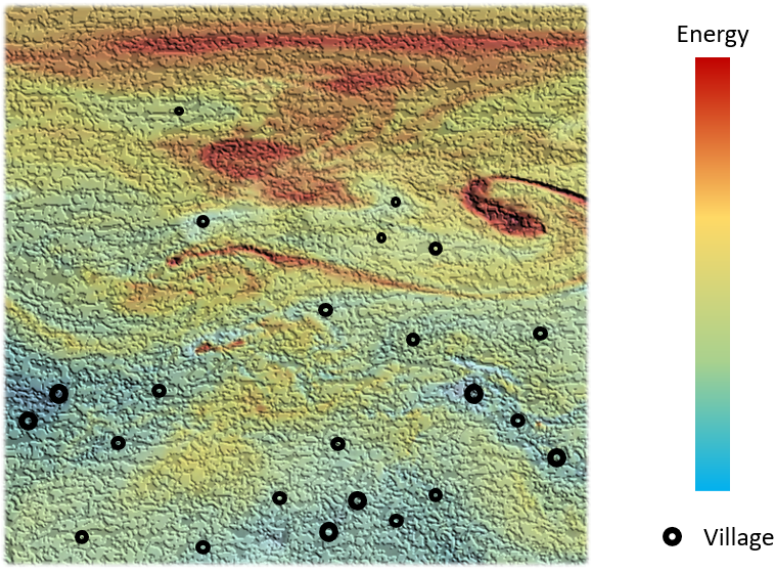

Figure 3.9: Energy-Diagram of the fictitious kingdom of Wurbaterban

accidentally contains an infinity loop. The speaker will wonder why the ethergram does not finish, but in principle the energy is fed from the environment and has no effect on the speaker of the dance. However, in this case, the ethergram will by definition never finish.

What if there is not enough Energy to execute an Ethergram?

Strictly speaking, there is never too little energy to execute an ethergram, but too little time. There is always a little energy in an environment, and at some point, the ethergram would be completed. The point is that the ethergram would potentially take far too long, and in some cases not even a lifetime, to finish and therefore it is commonly said that there is not enough energy to complete the ethergram; more correct would be to say that the ethergram takes too long, or, that there is not enough energy to finish the ethergram in a reasonable amount of time. Because if there is only very little energy, spells are cast very slowly. This can go so far that it would take the time for stones to pass before a wick could be lit by the fire trick in an extremely low-energy area. So in practice this distinction does not matter, but it is good to know that ethergrams usually could finish (that is, if they don't contain infinite loops), if only one had enough time.

Energy and the Retrieval of Information

The fact that one cannot access all the information of an object because it requires too much concentration also has to do with energy. Some properties require more energy to retrieve, others require almost no energy, and some can practically never be retrieved. At this point, the special power of water can also be understood, why thoughts can be accessed more easily through water. Water facilitates access to mental information such as thoughts, psyche and spirit, which normally would not be accessible to humans with real amounts of energy.

In the case of information retrieval, the same applies as in the last section: accessing information that requires a lot of energy is never impossible, it can only take too long relative to the life span and patience of the human being.

Energy- and Object-Diagrams Ether representations thus potentially contain all possible information about a thing, but they vary in difficulty of access, which is why ether-diagrams typically only represent the information that can be easily obtained without much effort (or at least with a reasonable amount). Information that changes quickly and information that have many dependencies tend to be harder to access. The most extensive ether-diagrams also describe information that require more energy to access, such as health status, birthday, thoughts, state of mind, etc. They are usually only used für reference purposes.

Energy and the Four Elements

In order for powers of the four elements to work, energy is needed. There are powers that require more energy than others. Some powers, such as the creational powers, cannot usually be performed in a place, because they require far too much energy. But if such a place exists, any living being could use the creational powers there.

3.2.7 Retrieving Property-Values Directly (Without Calling The Object)

Finally, one possibility of information retrieval that has not yet been addressed in this pillar should be mentioned: the information retrieval of property values that one does not touch. The question of whether this is possible comes naturally to one when one is reflecting upon the Ether and the process of thinking itself. For example, when one thinks about one's own dog, one can certainly recall certain characteristics from it, such as its name, etc. One would therefore think that it should be possible to recall property values at least in some situations. However, this is not the case. For these are only representations of the thing in the mind. One does not access the information of the actual thing, that is lives and not the memory of it in the mind. When one now thinks again even in the example of the dog one isn't actually accessing the dogs properties. For example, the weight of the dog might have changed in the meantime. One is merely accessing a memory one had of the dog. And because this memory is located within oneself, one can directly access it. Since it is thus still a form of retrieving (just not in the

way one would hope) and has no significant practical advantages, it is not followed up in the book and is not mentioned as a third possibility alongside retrieving and calling, but continues to be seen under retrieving.

It should be added, however, that this information about things is called mental representations of things. This kind of retrieval is only possible if the object is very familiar to one, so that an object could collect/form in the memory, respectively if it has been trained for a long time, or, by developing a strong emotional attachment to a thing. As already mentioned above, this is purely a function of memory and not a conscious interaction with the Ether and will therefore not be taken up again in the further course of the book.

The object that is created is also an ether-object, which represents the synaptic currents and chemical processes that form the memory of the object, e.g. the dog. It has however no connection to the ether-object of the real thing.

3.3 Worldly Layer

Information retrieval and active manipulation of ether-objects can of course also be used in ether-dances and combined with the language constructs learned in the first pillar. The possibilities that arise from this alone are already immense, but for variables in particular, this will also open up new, useful possibilities, which are lined out below. Furthermore, the possibilities of creating own ether-objects are described, and tips for designing low-energy ether-dances are given.

3.3.1 Variables and Ether-Objects

From now one, ether-objects can also be stored in variables. This will make it more convenient to call them and use them throughout an ether-dance.

„Storing“ Ether-Objects in Variables

It can be exhausting and annoying to have to focus on a thing every time when calling it. By storing the ether-object of a thing in a variable, this can be made more convenient. For example, you could store a nameless apple in the variable x with „variable x point to this apple!“ or „variable x bind this apple!“ or „variable x ban this apple!“, but this does not mean that now the whole apple would be contained in the variable. In fact, unlike arcanes, the value itself is not stored in the variable, but only a reference to the ether-object as seen in Figure 3.10, in which the ether-object of a tree is banned in a variable. Therefore, in the case of ether-objects, we also speak of the variable „pointing“ to the ether-object.

This means that the object is not stored in the variable like arcane values; instead, a permanent connection is established between the variable and the object (binding/banning does not mean that the object becomes immobile, nothing actually changes in the object itself). Thus, each retrieval of a variable is like calling the ether-object, but the ether-object itself remains in its place in the Ether without being changed. Practically,

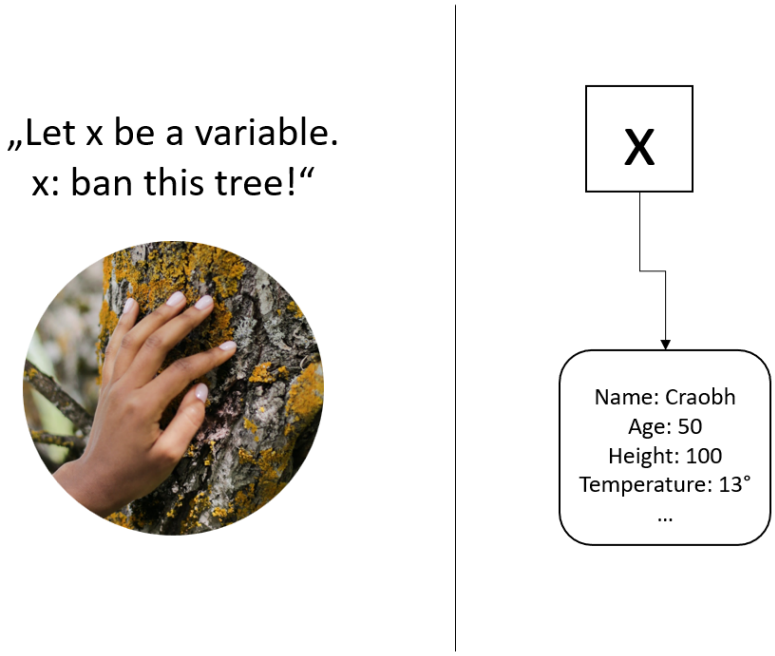

Figure 3.10: Ban ether-object of a tree in a variable

nothing changes; a variable is like a proxy for calling the object and merely makes it easier to focus on the object. The variable can be used in the same way as the object itself, for example to cast the fire element on the object as seen in Figure 3.11. Note that touching the object is still necessary. The variable only makes it easier to concentrate on the thing. Because of this, it is also possible to let multiple variables point to the same object; however, in a variable still only one object can be banned.

Another practical advantage of variables is that you can name them freely and thus also give a name to things without a name in order to deal with them more easily. It certainly doesn't make it better to name the tree x, in fact it would only make the ether-dance harder to understand. But one could, for example, name the tree „thisTree“ and call it as such throughout the ether-dance, which is more pleasant than having to call it all the time and also potentially increases the understandability of the ether-dance.

Using Variables to reduce Energy-Costs of Ether-Dances

In the case of high energy commands, it makes sense to cache them in a variable, as they can then, in the case of arcanes, be easily called up, especially if their access consumes a lot of energy. For example, the word-property mental state could be cached in this way.

With more complex things, that is, objects, variables only simplify access per se, because variables have shorter names and retrieval of the variable per se is easier, but the energy consumed remains the same. It's just that you don't have to concentrate on the thing every time, but can simply use the variable, which is much more convenient.

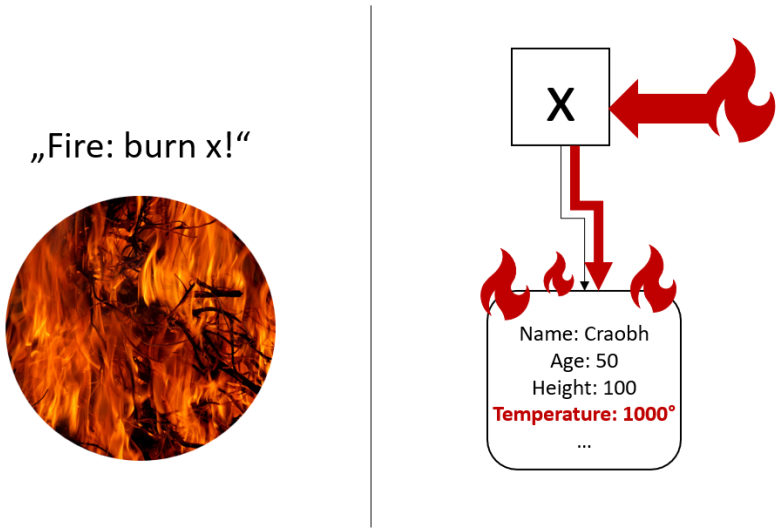

Figure 3.11: Casting the fire element on a tree, by using a variable that's points to the tree

When accessing objects, variables thus have no effect on the energy costs.

Variables and the four Elements

A final remark on whether you can also bind elements in variables: Elements cannot be bound in variables because they are not arcanes and not ether-objects.

3.3.2 Interaction with Ether-Objects in Ether-Dances

The abilities of variables to refer to/ban ether-objects that was learned about in the last section can now be used for ether-dances to make interactions with ether-objects even more powerful. One can retrieve information by calling an object as in Figure 3.12. There, the temperature of the burning tree of the last sections is accessed and spoken out.

Active manipulation is also possible. With the ether-dance in the ether-dance-protocol 3.1 one can touch an apple and check whether it is rotten or not. If not, it falls into one's hand.

```
1 Syoombraaya
2
3 Let MyApple be a variable.
4 MyApple points to this apple.
5
6 If age of MyApple < 5 {
7     Earth: exert force vertically downwards on MyApple.
8 }
9 otherwise {
```

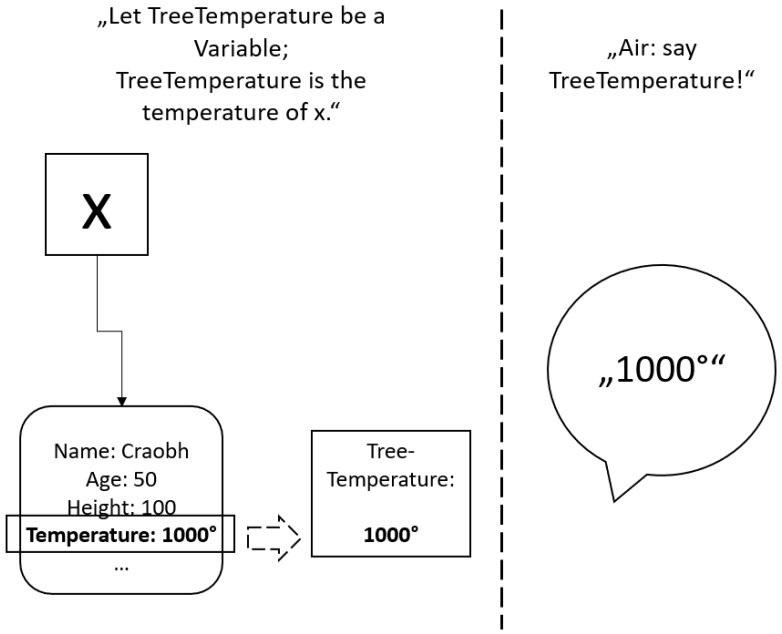

Figure 3.12: Access to property of ether-object and subsequent pronunciation by air element

```
    Air: say "Rotten apple".
}
```

Ether-Dance-Protocol 3.1: Ether-dance to pick an apple or recognise as rotten

One can see in the example that arcane operators can also be applied to the values of properties, since these are also arcanes. An addition or comparison of number properties is thus possible without further ado and enables various mathematical possibilities.

Another example would be to have a stone circle around a tree trunk. If one has the position/coordinates of the objects, one can calculate the distance and then determine the course with the air element.

3.3.3 Creating own Ether-Objects

Besides interacting with natural ether-objects, it is also possible to create one's own ether-objects and interact with them. This can be useful when one wants to do calculations without having to call the actual objects.

Creating artificial Ether-Objects and Interacting with them

For example, in ether-dance-protocol 3.2, a separate object is created to represent the tree „Quadruk“ (of course the object has no actual connection to the physical world and

is therefore no natural ether-object). To do this, the properties of the object must be set and the values of the properties must be set. At first, the ether-object has no arcane-values and no properties and is therefore an empty object. The properties have to be manually set and endowed with arcane values. After that, it can be used in the same way as if it were an ether-object with a connection to the physical world, thus enabling different types of calculations. Of course, this would have no effect on the physical world, as the object only exists in the ether and is called an artificial ether-object.

```
Syoombraaya

Let x be a variable.
Create your own object Quadruk.
Let the object have the property
    "size", a number
    and the property "width", a number.
Be the object in x.

The size of x is 5cm and the width 6cm.

Air: say the size of x!
```

Ether-Dance-Protocol 3.2: Creation of own ether-object

An artificial ether-object can also be interacted with. Information retrieval works the same way as with natural ether-objects, with the difference that all properties can be retrieved without concentration.

Manipulation of an artificial ether-object on the other hand does not work in the usual way, because it cannot be done via the four elements, as these can only be applied to natural ether-objects. Instead, the property values of the artificial ether-objects can be directly changed, as it would never be possible with natural ether-objects. In this way they can be filled with own properties and arcanes to be able to store own values for calculations.

That all ether-objects must have a unique connection to something also applies to these temporary ether-objects, which is why they are held with variables. Only if at least one connection to a variable still exists, does the artificial ether-object also exist.

Artificial Ether-Objects in the Two Worlds-Teaching

Artificial ether-objects have no thing in the physical world, with which they are interrelated, that is, they have no representation in the physical world. They are considered to exist with the other Ether-Objects (see Figure 3.13) but can only be called if they are connected to a variable; else wise they are impossible to call and therefore practically lost. Though they do not actually cease to exist but slowly fade over time, while fading over time means here that they are actually never completely gone but only become at some point practically impossible to interact with, even if one had a connection to them through for example a variable, and are therefore considered non-existent.

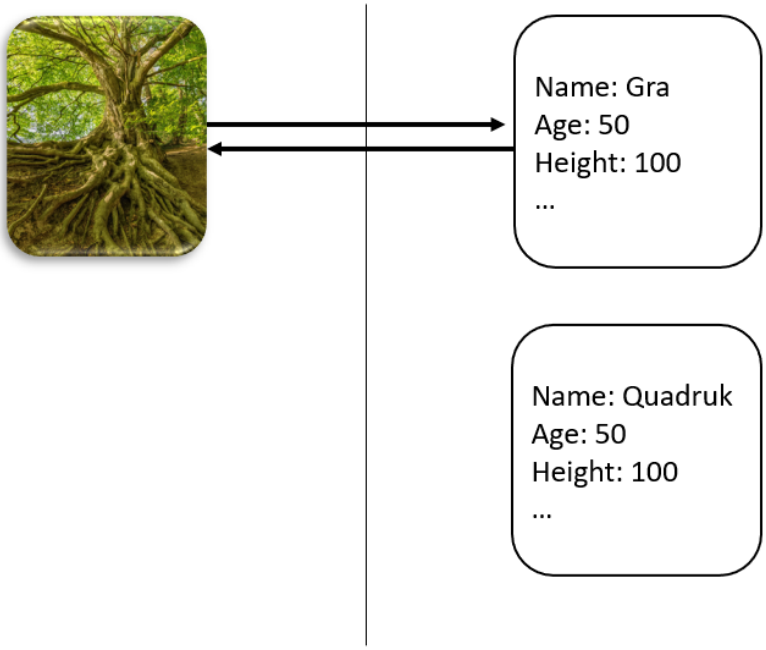

Figure 3.13: Position of natural and artificial ether-object in the two worlds

Transform artificial Ether-Objects into Natural Ones, or: Creational Powers

With the materialisation of artificial ether-objects, the transforming of an artificial ether-object into a natural ether-object, that is, a „real“ thing of this world. With the creational power of the earth, ether-objects could actually be created so that they become natural ether-objects. However, this is usually not possible for humans. How it then works is not further defined here.

Turning Natural Ether-Objects into Artificial Ether-Objects, or: Extinguishing Powers

In the same way, of course, there is also the possibility of erasing „real“ things, that is, erasing their physical representation. In this case, it is only present in the Ether and thus no longer a natural object. Thus it has been practically extinguished as a „real“ thing of our world. It still exists in the Ether for a certain time, but slowly disappears because it no longer has a connection to its physical representation that defines it. If it was held in a variable, it will continue to exist through that connection as long as it exists. Otherwise, if it has not been held by any variable, it is no longer able to be called in the Ether and thus practically, that is, from a worldly/medial point of view, no longer existent (see the explanations in the section before last). Just like creative powers, extinction powers of the fire element cannot normally be performed by humans.

3.3.4 Energy rules of thumb for Ether-Dances

It is not possible to execute all ethergrams just like that. The best ether-dance is of no use if it is not finished. Therefore, one should know a few rules about which interactions with the Ether require how much energy.

For that, here are a few rules of thumb to estimate the energy-costs of an Ether-Dance:

- The larger the ether-object, the more energy it costs to interact with it.
- It is essential to ban hard-to-access properties of ether-objects into variables.
- The further one moves away from a thing, the more it costs to interact with it (variables can't help here)

4 Fourth Pillar: Ether-Classes

In the fourth pillar, structural regularities are found between the ether-objects in relation to their properties. These groups of objects, which all share the same properties, are called „classes“.

Ether-classes make it easier to interact with ether-objects of a certain class, e.g. trees or foxes. If one knows the class of an object, one gets information about its typical properties and can then easily use them in ether-dances without having to know the properties of that particular object.

4.1 Cosmic Layer

This section introduces ether-classes and the resulting „structured“ object-oriented ether-model.

4.1.1 Classes

Ether-classes are certain types of objects that all have the same properties. For example, all fox ether-objects have certain common properties and all tree ether-objects have certain common properties (seen in Figure 4.1). Perhaps they differ from each other in parts, but many of their properties are likely to be the same. Therefore, a collected name has been found for this group of objects: fox in the first case and tree in the second one. In the context of ether, these collective names for objects with the same structure are called classes. There is thus an ether-class for foxes, called the fox class, and another for trees, called the tree class.

4.1.2 The Structured Object-Oriented Ether-Model

In the last pillar, the Ether was a collection of fundamentally different, unique objects. Among these objects, regularities are now forming: some objects, such as all dog objects, have the same properties; they all have the properties of body size, tail length, etc. These apparent templates in the Ether that apply to all objects of certain groups are called ether-classes.

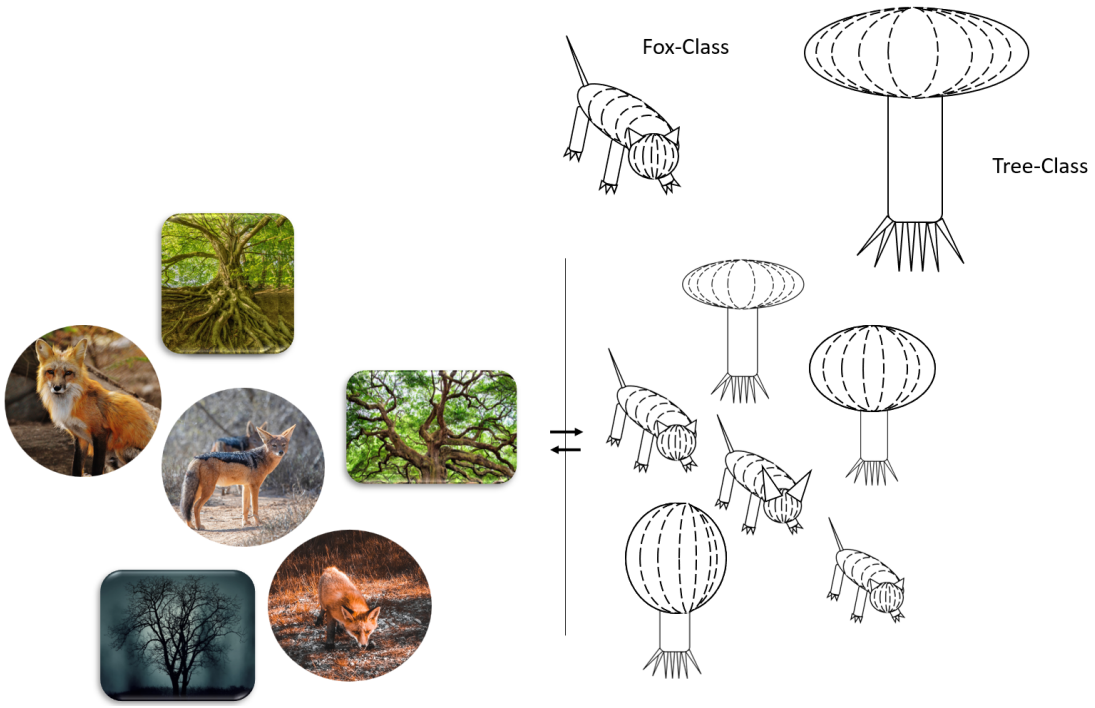

Figure 4.1: Foxes and trees, in the physical world and in the Ether, and their classes

The realisation of the existence of classes results in the ether-model of the sixth pillar: the structured object-oriented ether-model, shown in Figure 4.2. In it, classes are like templates that apply to all things in a group of objects. However, they are not contained in the Ether as separate structures from the ether-objects, but only exist in the Ether as structural regularities among the ether-objects.

For example, there is a tree class, most of which have a certain set of properties, e.g. a size, age, number of leaves, etc. The tree class describes the properties that all trees share (or at least a majority of them) as seen in Figure 4.3. However, the class itself does not exist as such in the Ether, it can only be observed as a regularity/structure among the objects there and should therefore always be drawn clearly separately from ether-objects in a diagram.

Ether-class also do not connect the objects of a class in any way other than that they share a certain set of properties. Ether-classes are no structures in the Ether, they are structural regularities that can be observed by comparing ether-objects with one another.

4.1.3 Class-Hierarchies

An ether-object does not have a specific class, to which it belongs but several. In fact, it potentially even has an infinite number of classes, although in practice only a few relevant classes are used and have been given names by people. For example, Figure 4.4

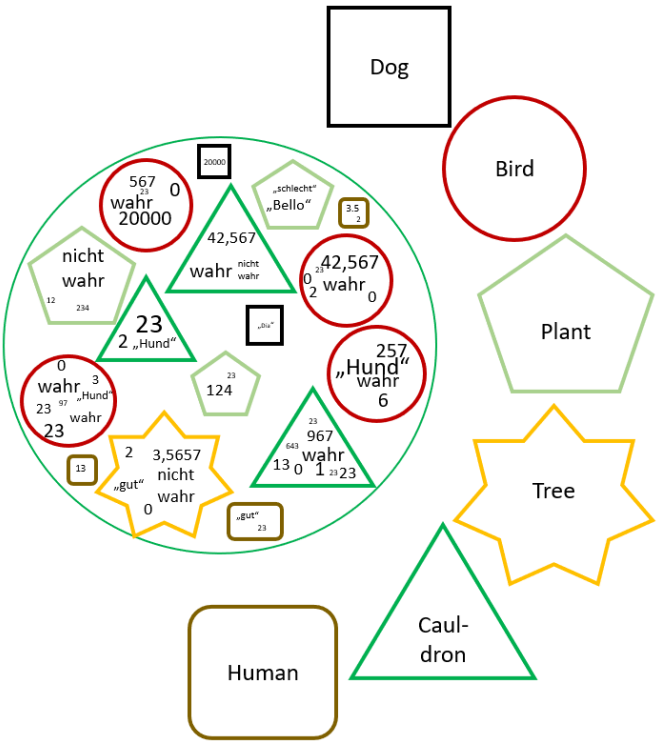

Figure 4.2: Pillar 4 ether-model, „structured object model“

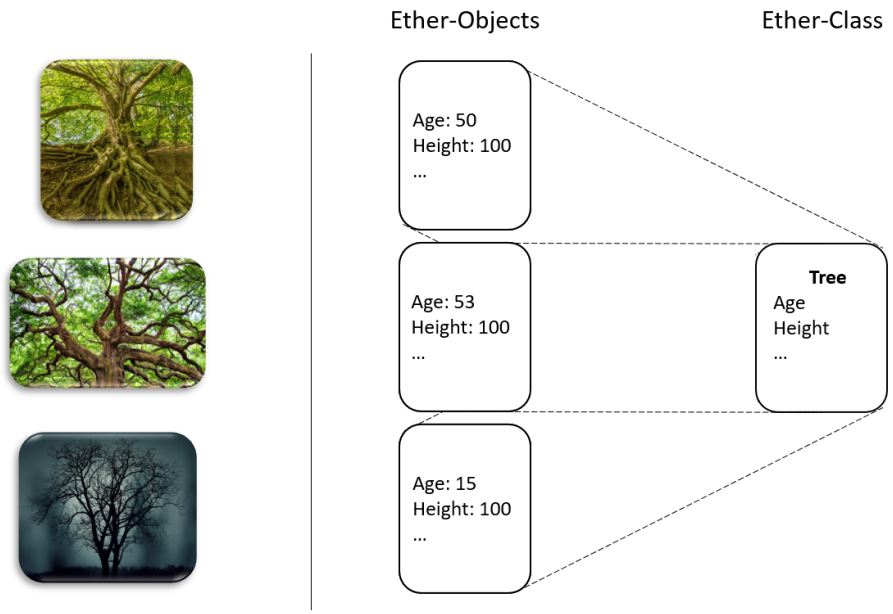

Figure 4.3: Tree ether-class and its properties

Figure 4.4: Some of the classes of a fox and their level of detail

shows the different classes to which a specific fox belongs. There is the living thing-class to which it belongs. It can also be included in the class of mammals and the class of foxes.

With this knowledge, for every thing in the universe a hierarchy of classes could be created, listing all the classes it belongs to. Moving up and down this hierarchy, the thing could then be described more generally or more specifically, depending through which class of the hierarchy one looks on the object.

Different classes thus also reflect different views of one and the same thing. If one considers a fox only as a living being, one cannot say much about it, but if one considers it as a mammal, one can already conclude that it is capable of giving birth to offspring (if it is a female). And if one thinks of it as the fox standing there in front of one, one can infer further information, for example, one may know that the animal has a name, and so on.

For a single thing, the class hierarchy is one long list of classes, which become more and more general upwards. However, one can try to combine a linear hierarchy with the hierarchy of another thing, for example that of a wolf. It is noticeable that fox and wolf both belong to mammals as well as to living beings. That is, one could classify them in the same diagram both under the class mammal, and mammal being of class living being.

In consequence this means that the class-hierarchies of different object are sometimes linked and for objects that are very similar, for example two dogs, they even overlap in major parts. One can thus imagine all classes of all objects of the universe in one hierarchy, which can be seen in parts in Figure 4.5. This class-hierarchy, that contains all the classes of the universe in one big structure, is called the class-hierarchy of all things. Since it contains an infinite number of elements, it is impossible to write down, although

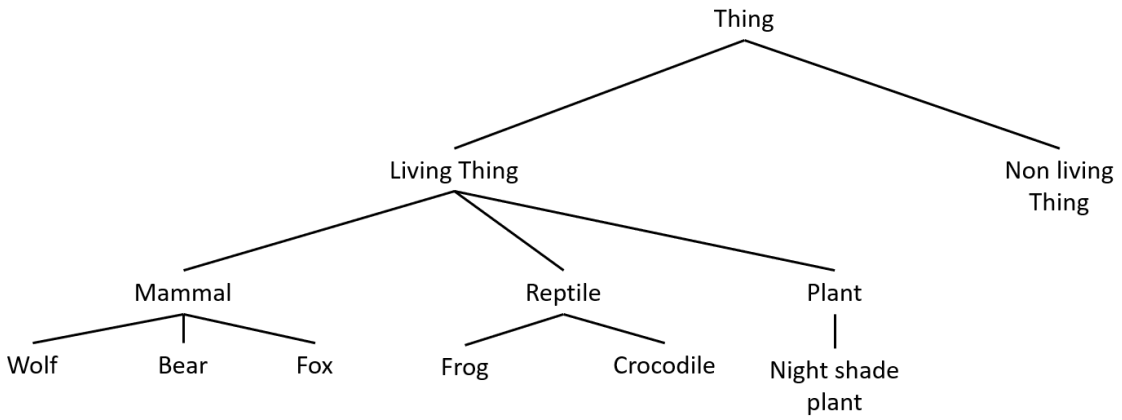

Figure 4.5: Class-Hierarchy of all things (incomplete depiction)

parts of the hierarchy could be printed for example in books to show an overview of important related classes.

Until now the hierarchy of all things has been thought of as being a big tree whose branches keep splitting indefinitely. However, the structure of the class-hierarchy of all things does not actually look like a tree whose branches keep splitting: there are also branches that come together again, that is, connections between former branches. For example, although a bat is an animal with wings, it is also a mammal. Therefore, it belongs to both the class of mammals and the class of creatures with wings, and would have to be represented as seen in Figure 4.6. This means that a completely branched tree structure is not given. Strictly speaking, the structure should therefore be seen as a network rather than a tree. However, usually in the representation of class-hierarchies common classes are to be used and a reasonable selection of the relevant classes is to be chosen, in order to ensure a clear diagram. That is, it can usually still be thought of as a tree, even if it is actually a net.

4.1.4 Ether-Class-Diagrams

Additional to object-diagrams there also exists diagrams to visualize ether-classes. They will important in the medial layer, because they can be used to look up the properties of objects one wants to interact with.

Components of Class-Diagrams

For example, in Figure 4.7 one can see the class-diagram of the tree class. It can be observed that the class is usually represented centrally as the heading of the class.

In contrast to object-diagrams, a class-diagram describes the properties of all objects of this class. This allows one to see what properties a particular object has if one knows that it belongs to a particular class.

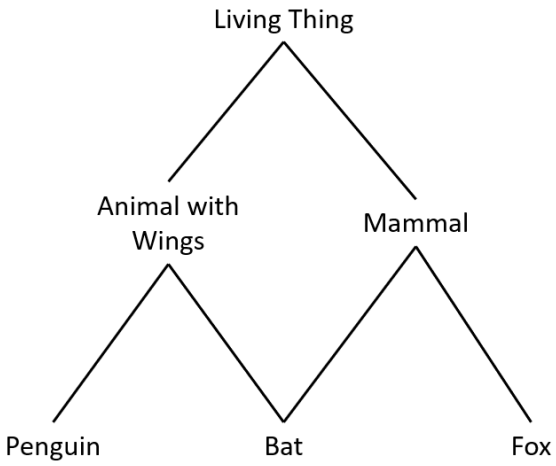

Figure 4.6: Class-Hierarchy with one class with multiple parent classes

Figure 4.7: Class-Diagram of the tree class

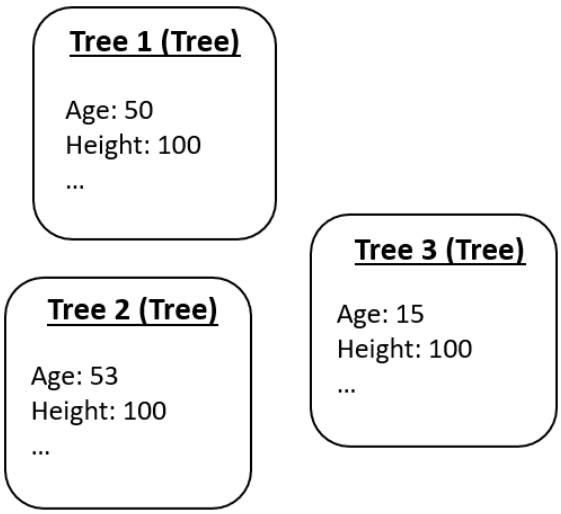

Figure 4.8: Object-Diagram showing the classes of the objects

Adaptation of Ether-Object-Diagrams

One might wonder whether with the introduction of class-diagrams, object-diagrams become obsolete but in fact, quite the opposite is true. Class-Diagrams can even help to create object-diagrams by providing information about which properties to enter for an object. At the same time, object-diagrams are still useful, especially to illustrate examples or to visualise particular situations, in which class-diagrams would be of little use. That means, class-diagrams do not replace object-diagrams, they complement them.

In some situation, however, it might also be useful to show the class to which an object belongs in an object-diagram. This is not possible in the current object-diagrams, because they were created before the incorporation of classes and have no way of handling this information. Because of this, object-diagrams must be slightly modified and adapted to the concept of classes. For example, the situation could arise that when interacting with an object, one wants to indicate that one wants to interact with it as the object of a certain class, for example as a dog or as a plant. This means that one would like to write in an object-diagram with which class the respective objects are called. In these cases, the class of the object can be written in parenthesis above it (or next to the name of the object if it has one). For one, the parenthesis indicate that the class of an object is never fixed and an object can have many classes (as described in the last section). Therefore, several class names can be written there, which can be separated by commas. Additionally, they indicate that one does not mistakenly confuse class names with a unique name of the object (which does not exist). Furthermore, name and class are to be underlined so that one does not confuse object-diagrams with class-diagrams. That's why, from now on, the headings of objects are underlined for every object-diagram from this pillar on, while the classes of class-diagrams are not. Such an adapted object-diagram is shown in Figure 4.8.

It is of course also still possible to use object-diagrams as described in the last pillar without the incorporation of class-diagram-information. It is always possible to add the class to objects of an object-diagram but never necessary. It is even possible to display the classes of some objects and not others, as long as the heading of all objects are underlined.

4.2 Medial Layer

This section describes how classes can be used to more reliably retrieve the information of things, especially when one doesn't know the names of a thing's properties.

4.2.1 Calling Ether-Objects with their Class

In the following is explained, how to call ether-objects with their classes.

Different Views introduced by Calling With Classes

Classes are the common way to call things in the world. In fact, we already used them in the last pillar to call ether-objects, only at that time we were not aware of using classes, for example when we called a special tree as „tree“ and imagined certain properties under it, although its actual ether-object might have far more properties. Classes are very important in the practical use of ether. Only thanks to classes we know what properties certain things have. When we see a dog, we know that it belongs to the dog class and thus has certain typical properties. Classes help us to find our way around the huge number of ether-objects in the Ether without having to know the properties of each object.

As described in section 4.1.3, each ether-object potentially has an infinite number of classes it belongs to (but mostly there are only a handful of common classes), and each ether-object can be called as any one of its classes. Even though certain information are lost in the process, since only the main properties are listed in classes, it's often easier to call something as its class, since one usually only needs the most common class properties anyway.

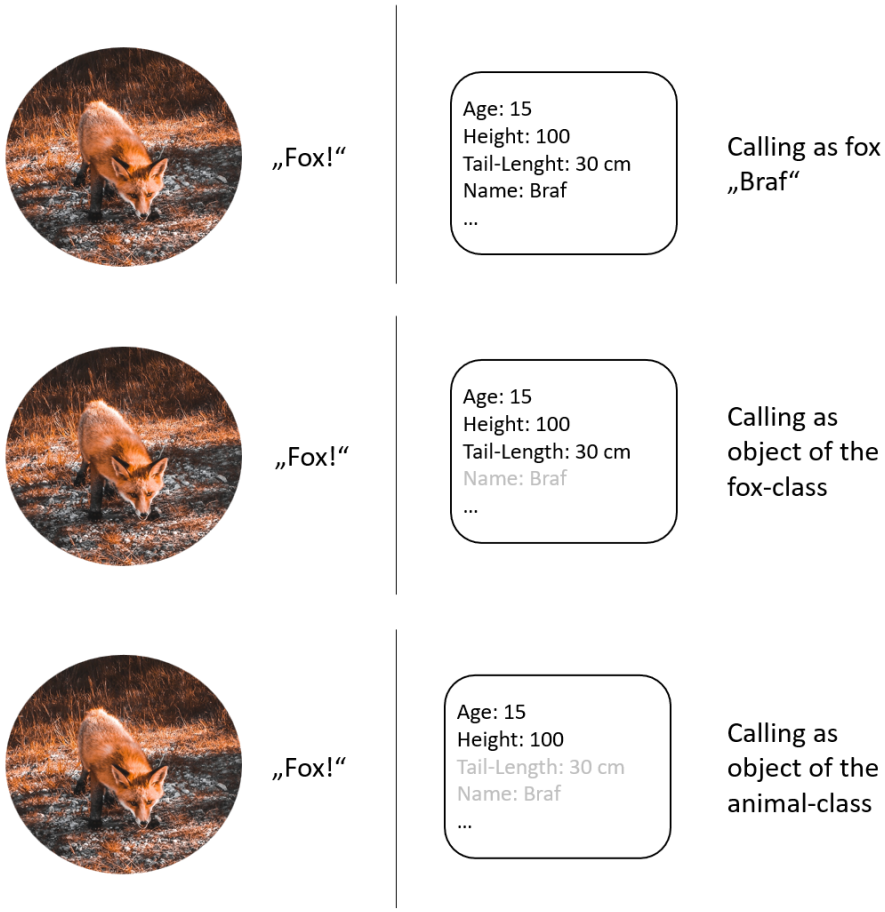

Figure 4.9: Calling fox object with different classes

All of this doesn't change anything about calling classes. Classes can still be used as easily as in the last pillar. The concept of classes are mainly about knowing what properties a certain thing has and what properties it doesn't have. Ether-classes help one to find them out. In fact, the properties we saw in the last pillars are only because we know which classes the objects belongs to.

For example, in Figure 4.9 a fox is called with different classes. We can see that with the mere knowledge that the fox has a name, it can be called with a more precise class.

It should be noted that in principle one can call the fox with any word, so in Figure 4.9 was always used the word „fox“ to show that the choice of class happens mentally; however, it is often helpful to speak out the name of the class with which one wants to call a thing. So, for example, if one want to call the fox as an animal, which would make significantly fewer properties visible, one would call it with the word „animal“.

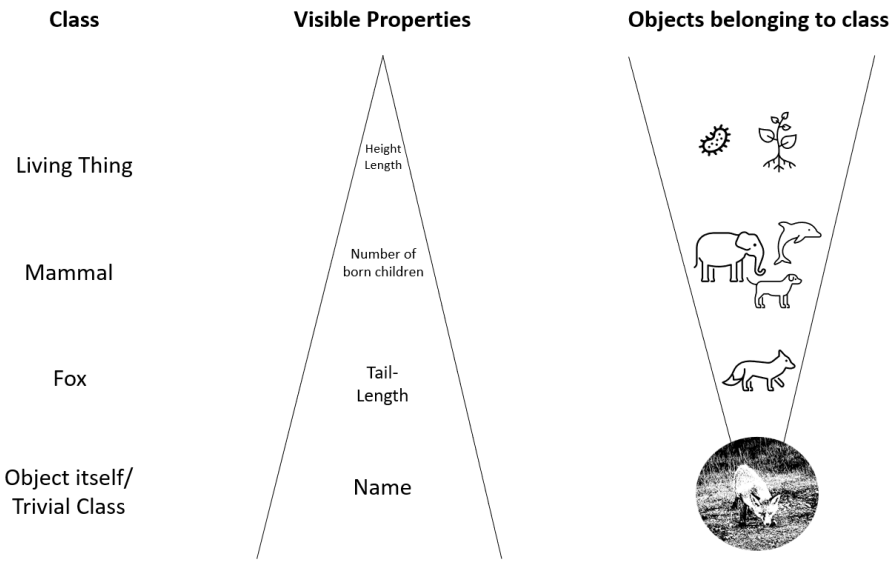

Figure 4.10: Different views on a fox ether-object by different classes

Role of the Class-Hierarchy when Calling Things

As already seen in the last section, one and the same object can be called in different ways with the introduction of classes. Depending on how it is called, more/less properties become visible in the process. For each thing in the universe exists thus a whole set of possible classes, which describe the thing ever more near, until the class has exactly the same characteristics as the thing and applies only to the thing itself. Which class one chooses is thus always related to the level of detail to which one wants to talk about a thing. If one wants to know the size of a fox, it could be sufficient to call it as an animal, since this contains the property size. If, on the other hand, one wants to know fox-specific information such as tail length, one would have to call the fox as fox (see Figure 4.9). On the other hand, if one wants specific information about a particular fox that has properties that other foxes may not have, such as name, one should call it as that particular fox.

One can see that when one calls a thing with a class in which certain properties are not present, the properties of that thing have not disappeared, they are still there; they are just not visible. A class is thus also a kind of view of a thing. Depending on how close one approaches the class to the thing itself, the more information one sees, but the smaller the group of objects with which it shares properties. In Figure 4.10 one can see an excerpt of the class with which one can call a fox. In the lowest level, the fox is called as this fox there on the meadow, i.e. as it could also be called without the use of the class concept (this would be an ether-class, which has the same properties as the ether-object and thus describes it completely, also called trivial ether-class). This possibility of calling remains of course.

As one can see, the higher one goes up the class-hierarchy, the less visible the properties are. The properties of the previous class are also contained in the class below; but not vice versa. For example, the property Size is visible for all classes shown, while Tail-Length is visible only for the trivial class, as well as the class Fox, but not in the other classes. On the other hand, more and more ether-objects are contained in the respective higher classes, making the class more general and allowing it to be applied to more things. For example, the class of mammals has many more objects than the class of foxes.

It is therefore always aware of the fact that there are many classes available to call a thing and that which one to choose depends on how much information one has about a thing and which degree of detail is needed.

4.2.2 Practical use of Class-Diagrams

Class-diagrams can help to retrieve information from objects. In particular, they help to find out the names of the properties of a thing one wants to interact with. Without class-diagrams, one would have to guess properties based on intuition alone. With class-diagrams, one can check one's intuition and support it with facts.

Use of Class-Diagrams when calling Unknown Things

Class-Diagrams can be useful when interacting with an unknown object, because sometimes one might finds oneself in a situation where one doesn't know the objects involved well enough to be able to write down all of their properties for a object-diagram, or simply, because there is not enough time. In these situations class-diagrams, or a basic knowledge of them, can help to estimate properties and their values to quickly estimate how to be able to interact with them.

This is especially apparent in the presence of a monster. If you have a thing in front of you that moves, but you don't know exactly what it is, it becomes rather difficult to call it, let alone access one of it's properties (that is, calling does not become difficult per se, but as what to call it). If you could now use a class that describes the animal to some extent, e.g. as a mammal, you can already retrieve rudimentary information about it. For this you would use the class-diagram of the class „mammal“ and look at its properties, maybe you already know them. Now you could call the corresponding properties.

In some cases, the class of a thing is not immediately obvious. But even here classes can be useful. For example, in Figure 4.11 there is a creature that could not be known yet. It would therefore be very difficult to know what properties it has and impossible to interact with it without this knowledge. However, thanks to classes, one could already

Figure 4.11: Unknown entity to interact with

say with certainty that it is a living being and thus has size and age. Now the properties size and age are relatively easy to see in this case. However, the fact that the properties of ether-classes often follow the gut feeling does not speak against them. An extensive knowledge of ether-classes and their properties is obviously more useful than just listening to gut feelings, especially in situations where time is a valuable commodity. Still, further narrowing down the class would be helpful here. It would be best in this case to use a reference book to find out the possible classes of the being. To do this, one could look for the class-hierarchy of the creature class and see which classes are below that and apply to the creature. You could go down the class-hierarchy in this way and narrow down the being more and more.

But sometimes even that doesn't help anymore, because you don't get further at one level. Another possibility would be to try to narrow down the class further by checking properties as described in the next section. So you could finally learn that it is a talking axolotl and you can just talk to it.

Knowledge of the most important ether-classes and their properties is thus essential for interacting with the Ether, since otherwise one would have to rely entirely on intuition and guess all the properties for certain objects, such as unknown objects.

4.2.3 Accessing Non-Existent Properties, or: the „Nothing"-Value

The following sections deal with the case of accessing a non-existent property of an object.

The Nothing-Value

One might ask oneself, what value one gets when accessing a property of an object that does not exist? The answer to this question is: one does not get a value. One gets nothing.

Since it is difficult to handle and talk about this unavailable value, the value „nothing“ is introduced as a placeholder. It is assumed from now on that it will be returned in these cases. It behaves like an arcane value, but strictly speaking it is not one of the arcane, since by definition it is what is returned when there is no arcane value. It is thus the anti-arcane, the mass in the Ether that is not the arcanes. One can think of it as everything in the Ether that is not the arcanes. Such a place in the Ether is accessed when trying to retrieve information from a property that does not exist and therefore such a place is also returned as information.

So if one accesses a object's property that does not exist, the nothing-value is returned. On this no operations can be carried out meaningfully. If this happens nevertheless, the result is always „nothing“.

Checking Properties

Because of the danger of falsifying calculations that start from the nothing-value, it can sometimes be useful to check whether a property actually exists in an ether-object. To do this, one simply asks the ether-object whether a property exists and receives true or not true as an answer, for example like this: „fox: do you have the property name?“. For example, with the creature in figure 4.11, this would have told one directly whether it can speak or not. However, this is often annoying and tedious. It is easier if one already knows what classes a thing has. Access to the most common Class-diagrams and a good basic knowledge of classes are usually enough, so manual checking is usually not necessary. It can though sometimes be used in combination with the two said skills, to narrow down the right class with the help of checking properties.

Checking the Class of an Ether-Object

Even easier than inferring classes from certain properties is to check whether an object belongs to a class or not. To do this, simply ask the object: „Object Fifi: do you belong to class Manticore“ or „Does Fifi belong to class Manticore?“. Then all properties of the object are compared with those of the class. If they match, „true“ is returned, otherwise

„not true“.

The reliability of this check is related to having a correct picture of the class in mind. If one names as class „horse“ but thinks of a unicorn, the object will be checked for class unicorn and the check will evaluate as not true in most cases. It is therefore advisable to have a written representation of the class one is calling, for example in the form of a class-diagram.

Thus, in the case of the creature in figure 4.11, it would be even easier to find out whether it is a carnivorous Argos monster or not (in this case, it is not). In some cases, however, there is not enough time to go through all possible classes, so an extensive knowledge of classes and the appearance/behavior of their associated physical things remains indispensable.

Calling Things with the Wrong Class

If an object is called with a wrong class, only the properties that are present in the object and in the class are visible. Thus the probability is high that one receives „nothing“ as value with most accesses and the computation is falsified significantly. Calling things with a wrong class should therefore be avoided as best as possible.

4.3 Worldly Layer

Ether-classes are basically only relevant for calling ether-objects or retrieving information and have virtually no meaning in ether-dances. The ether-objects that become available through calling are the same objects as in the last pillar and can be used in exactly the same way, except that some properties are not visible and thus cannot be accessed. The main advantage of classes is that one can better understand the Ether and access its elements more reliably, which is not a concern of the worldly layer. Therefore, no practical examples of the use of classes will be shown in this layer, as they would be little different from those in the last pillar.

The only point worth mentioning for the practical use of ether-dances is the handling of the „nothing-value“. There is also the possibility to create one's own classes, which can be useful for some calculations.

4.3.1 Dealing with the „Nothing“-Value in Eher-Dances

Nothing-values can enormously distort the calculations of ether-dances. If at some point of the calculation, the value „nothing“ is the output one immediately knows that at some point in the ether-dance one has accessed a property incorrectly. However, it can also happen that in the course of a longer ether-dance it is not noticed that non-existent properties are accessed in between. Therefore, it makes sense to check at critical points whether properties are actually present and to offer alternative properties. In general, however, excessive precautions against these edge cases are usually not necessary.

Otherwise, the nothing-value, just like all arcane values can be easily retrieved and used like an arcane value.

4.3.2 Creating own Classes

It is also possible to create one's own classes, which in turn can be used to create objects of that class, as if one were using templates to create objects. It can be easier to create, for example, different dog objects which all have the same class, instead of creating each dog object separately and having to set all the properties each time. A class can save time and avoid errors there.

Creating a class works basically equivalent to creating an object, except instead of saying „create an object", it says „create a class". In ether-dance-protocol 4.1 one can see the creation of a unicorn class as an example. This can now be used to more easily make own objects with it, in this case unicorn objects which already have all the typical properties. This can make the work easier if one wants to create many objects of a class.

```
Syoombraaya

Create own class Unicorn.
Let the class have the property
    "size", a number
    and the property "horn size", a number.

Let lila be a variable.
Ban in lila a new object of the class Unicorn.
The size of lila is 5cm and the horn size is 6cm.

Air: say the horn size of lila!
```

Ether-Dance-Protocol 4.1: Creating the class „unicorn"

Such classes are called artificial ether-classes since they do not apply to any natural objects. Regardless of an object is artificially created from an artifical or natural ether-class, in any case the result is an artificial ether-object.

For the actual creation of the class, that is, the incorporation as a natural ether-class into the Ether, which actually applies to natural objects, which are also created in this course, the same applies as for ether-objects: it needs for it the creational powers of the earth element, for which humans normally do not have enough energy available. Should this nevertheless be the case, objects are created (depending on the available energy), which correspond to the class, both in the physical and in the Ether, these arise directly from the spirit of the person, who exercises the power. With the creation of these objects, the class is automatically natural as well, since it now applies to natural objects in the Ether.

In principle, the creation of natural classes happens quite automatically by finding new terms and ideas for things. The conscious, forced creation of natural classes and their associated objects, however, requires creational powers.

5 Fifth Pillar: Spheres

In the fifth pillar, connections arise between the objects that previously existed independently of each other in the Ether. This creates a large, hierarchical structure that connects all ether-objects.

Because all ether-objects are connected to each other, it will now be possible to call any objects in the universe and thus also to call objects that are not directly touched. This new possibility of calling is called spherical calling.

In this pillar, the main goal of the medial layer, the „access to all things“, is thus achieved. It also completes the understanding of the structure of the Ether that is given in the cosmic layer in this book. One thus has all the available background knowledge for performing ether-dances; the fifth pillar thus concludes the given part of the ether-teaching and leads into the free discovery of the Ether in the sixth pillar.

5.1 Cosmic Layer

In this section, spheres are introduced, which connect all objects of the universe. The basic concept of spheres is first briefly explained in the next section. Based on this, it is then described which Ether-Model results from it and which ways exist to represent its inner structures, as well as which effects the introduction of spheres has on the possibilities of representation of ether-structures known so far.

5.1.1 Spheres

The fifth pillar is based on the realization that objects can contain other objects. An object and all the objects it contains are called a sphere.

For example, a tree obviously contains more than only one object. It has many different components, such as the trunk, leaves, roots, and so on. All of these together form the sphere of a tree as seen in Figure 5.1. Thus, a sphere can be thought of as a container of all the components of something.

However, these objects could also be broken down again into their components. In Figure 5.1 the apple-sphere itself is split into its stem, seeds and pulp. The objects contained in the tree-sphere are thus all spheres themselves, which in turn can be broken down into smaller spheres, and so on. The same thing that has been done with the apple should be done for every child-sphere of the tree, as well as for their child-spheres and so on, if one would want to represent the complete tree-sphere. Every thing in the universe is thus not only an ether-object, but always also a sphere containing other spheres, since every thing in the universe has smaller components into which it can be broken down.

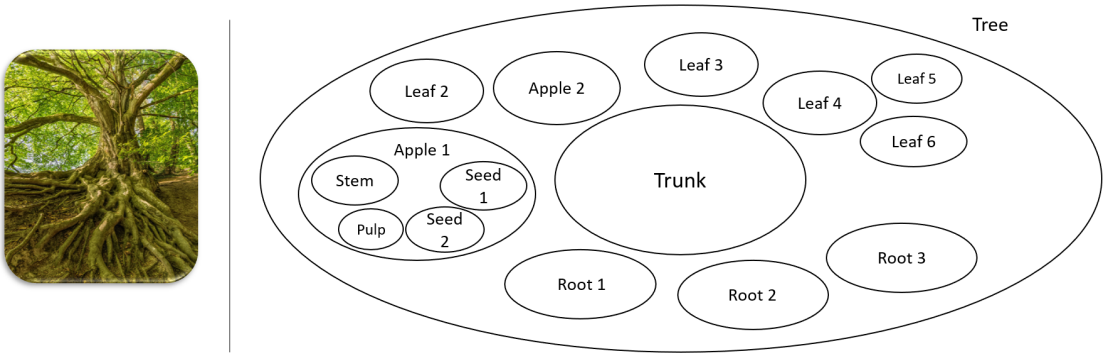

Figure 5.1: Tree sphere (sphere notation)

5.1.2 The All-Sphere, or: the Spherical Ether-Model

The representation of the tree sphere in Figure 5.1 is in fact only a section of the tree sphere, since, for example, the sphere of the apple, which are also contained in the tree sphere, could in turn be split into smaller spheres. In addition, the tree-sphere is in turn contained in other spheres, directly above for example it could be the sphere of the forest, for example, in which the tree is located and above the continent in which the forest is located. That is, one can imagine the Ether in this pillar as a huge nesting of spheres, which, starting with the sphere that contains all the spheres of the universe, branches out infinitely into smaller and smaller spheres.

The sphere that contains all the spheres of the universe is also called the world-sphere; in Figure 5.2 one can see an illustration of it, which of course can only represent a section of its child- and children's-child-spheres, as there are potentially an infinite number of them.

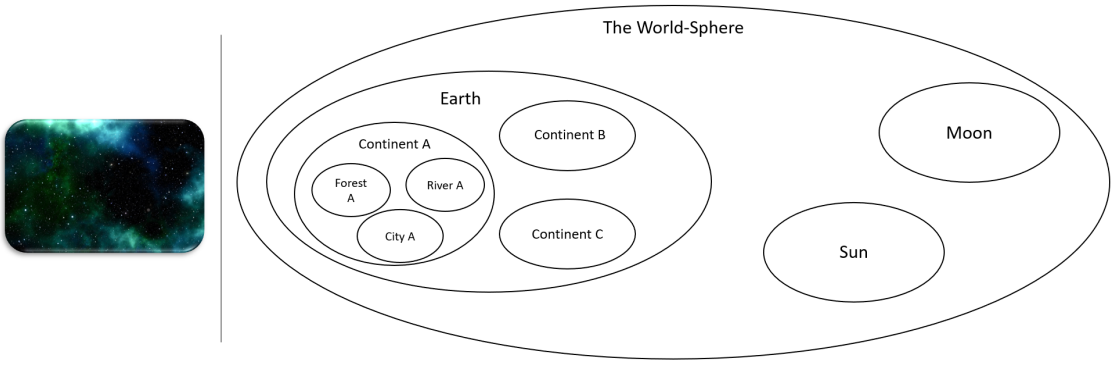

Figure 5.2: The world-sphere and its children and children's children (sphere notation)

This results in the new ether-model, seen in Figure 5.3, in which all objects in the universe are also assigned to a sphere and are contained in other spheres and are thus interconnected in a large hierarchical structure.

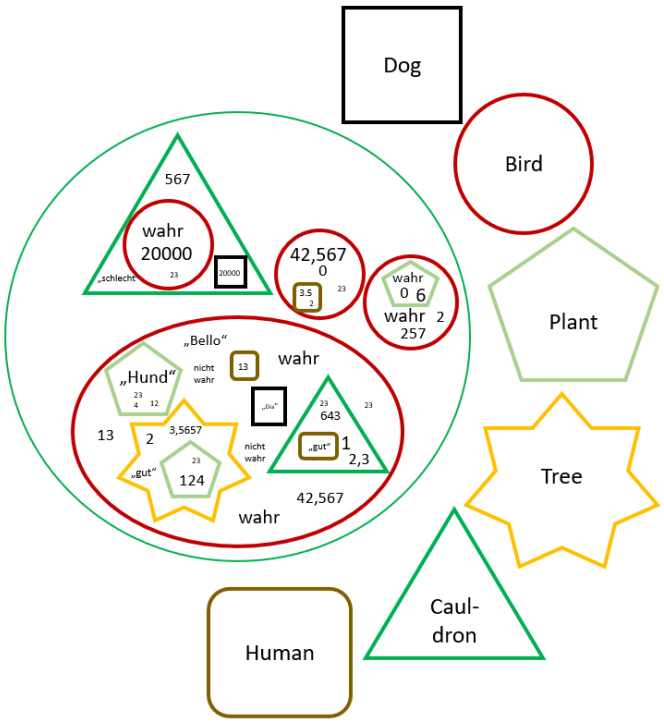

Figure 5.3: Pillar 5 ether-model, „spherical ether-model“

Each sphere is assigned to exactly one ether-object, for whose contained ether-objects it stands, and vice versa; that is, each ether-object is always assigned to exactly one sphere. Nevertheless, as will become clear in later sections, the terms cannot be used synonymously.

5.1.3 Sphere-Diagrams (sphere notation)

The representation with which spheres are shown, for example in Figure 5.2, is called sphere-diagrams. In these, one can see very well how spheres in turn consist of other spheres etc. and thus represent a nesting of theoretically infinitely many nested spheres.

In the notation of sphere-diagrams, which have been shown in the last figures, called sphere notation, spheres are visually represented by circles and can therefore obviously be distinguished from objects and classes. The circle represents the boundaries of the sphere, while the sphere itself contains all spheres within it.

It is true for all sphere-diagrams that they can only ever represent a section of a sphere, since the spheres contained in it can also be broken down into smaller spheres and, in the same way, their children and children's children. This infinite nesting, which theoretically extends down to the atomic level, is impossible to represent in a diagram, which is why usually only the first generations of child-spheres are shown.

5.1.4 The World Tree, or: Sphere-Diagrams (Tree Notation)

In this example, the hierarchical structure in the Ether will be described in more detail, with a special focus on the role ether-objects play in it. In this context, the concept of the world tree is also introduced.

The Tree Notation, or: the Difference between Objects and Spheres

Ether-Objects no longer appear in the sphere-diagrams presented in the last section. It is therefore difficult to understand what role they play in the new ether-model, or whether they exist in it at all. Therefore, a variant of sphere-diagrams is introduced, which makes the ether-objects visible in the new ether-model. Is called the tree notation and with its help, the difference between objects and spheres becomes better understandable.

In Figure 5.4, the alternative representation of the sphere-diagram in Figure 5.1 with the tree notation can be seen. Instead of using circles to represent the spheres, the tree-notation instead shows the objects assigned to the spheres with lines. This representation makes it easier to see the connection between ether-objects through spheres. A line between two objects means that the lower object is contained in the sphere of the other object. This is also called the object's sphere-connection, which will be discussed in more detail in the next section.

The difference between object and sphere finally becomes clear in Figure 5.5, in which a more complete section of the sphere of a tree is shown. A sphere is thus assigned to exactly one ether-object and includes all spheres contained in it, which in turn are assigned to ether-objects. A sphere is thus a set of ether-objects that are arranged in a certain structure. The ether-objects in turn are only individual points in this structure and hold information about this point.

That is, when one speaks about the sphere of a tree, one speaks about the tree as a whole, that is, with all the things contained in it. When one talks about the ether-object of a tree, this simply denotes the information associated with that sphere, e.g.

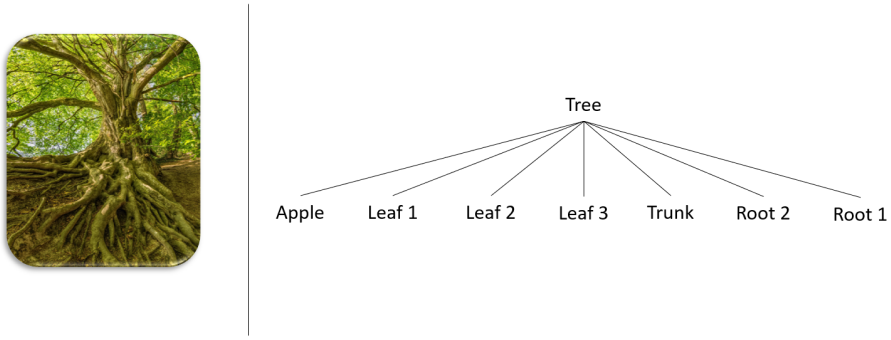

Figure 5.4: Tree sphere (tree notation)

age, height etc.

Both diagram-variants (tree and sphere notation) thus exist in their own right, as they each represent different aspects of spheres and can be used for different purposes. While sphere-diagrams of the sphere notation represent the spheres, sphere-diagrams of the tree notation represent the objects and the connections created by the spheres assigned to them. The tree notation thus has a stronger focus on ether-objects. They are therefore also called sphere-object-diagrams (not to be confused with object-diagrams). It also emphasises the fact that objects and spheres are not the same thing: a sphere is a substructure of the world tree, that is, an object and all its children and children's-children, whereas an object is just a single point in this structure that holds information. Sphere-diagrams of the sphere notation, on the other hand, are better suited to explain the concept of spheres and to be able to imagine them better. They are therefore used primarily for didactic purposes.

Sphere-Connections

With the introduction of the tree notation, one gets a new view of spheres: namely, as connections, as which they are represented in the tree notation. At the same time, it could be observed in the last and previous sections that the child-spheres, i.e. the objects that are directly below an object in a sphere-diagram, are spoken of particularly frequently, for example, which child spheres a tree has, etc. Because the ether-objects directly connected to a sphere-object have such great significance, the connections with them have been given their own name: sphere-connections.

A sphere-connection describes a connection between two ether-objects that occurs because the sphere of one object is a direct child-sphere of the other object. It is important to understand that sphere-connections do not describe objects, but a connection between objects, however, they are most of the time talked about from the point of view of a specific object, that is, the sphere-connections of a given object. The sphere-connections of an ether-object that exist either to the child-spheres or to the parent sphere can be considered. A general distinction is made between two types of sphere-connections:

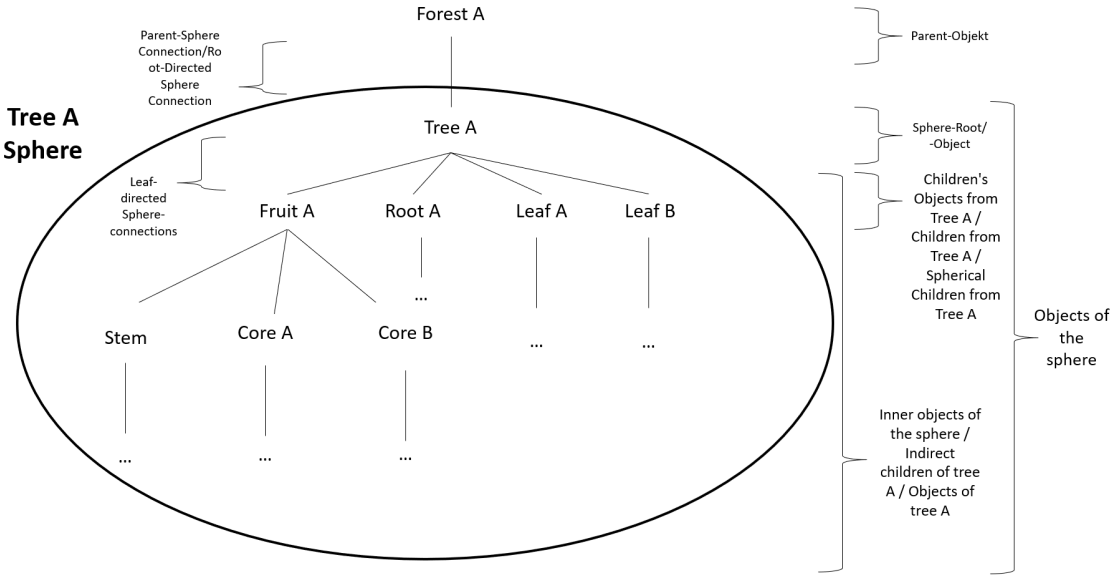

Figure 5.5: Difference object and sphere

- Child-Sphere-Connections/Leaf-Directed-Sphere-Connections: Connections to objects of the child-spheres.
- Parent-Sphere-Connection/Root-Directed-Sphere-Connection: Connection to the object of the parent sphere.

While child-sphere-connections connect an object to its child-spheres, which are in general more than one, parent-sphere-connections are always the connection to only one object, because every sphere has exactly one parent-sphere (only the world-sphere has none).

Since leaf-directed sphere-connections are more often used practically, sphere-connections and leaf-directed sphere-connections are often used as synonyms.

The World-Tree

As already described in the second to last section, the branching of the spheres can be continued as deeply as desired, since all objects are again assigned to a sphere, which in turn contains objects, and so on. The same holds for the other direction, because every ether-object is itself contained in a sphere, which is part of a bigger sphere, etc. (while this direction comes more quickly to an end). From the point of an ether-object, a sphere-diagram of the tree notation thus can be extended potentially infinitely in both directions (meaning from small to large) as well as infinitely extended in width, because every parent-sphere has child-spheres, which could be added to the diagram and also extended in both directions. So theoretically, a sphere-diagram of the tree-notation would be possible that contains all objects in the Ether. This construct is denoted as the

world tree (see Figure 5.6). In the sense of the tree-notation, a sphere is always a sub-piece of the world-tree, which is „hanged“ on an ether-object and branches downwards infinitely deep, that is, branches into smaller and smaller objects.

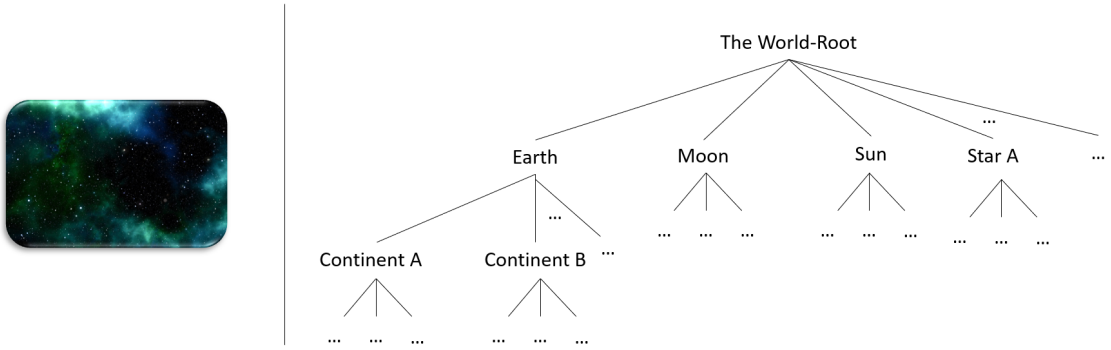

Figure 5.6: The all-sphere and its children and children-children (tree-notation), a.g. „the world tree“

Ultimately, the world tree is only the representation of the world-sphere in a different variant of the sphere-diagram. However, the tree notation and the world tree will play a central role especially in the medial layer, where the position of objects in the world tree will be of great importance. That's why the ether-representation of the universe in its entirety is from now understood as being structured in the form of and often itself called: the world-tree.

Sphere Boundaries

One might ask oneself how the boundaries between spheres are defined. For example: are a person's clothes (pants, top, etc.) part of their sphere or separate spheres?

Which of these two cases applies is not further defined here and must be defined for the specific cases. However, within a world with the Ether, just like the existence of the classes, sphere boundaries also exist without fixed logical justification. They can either be taken as given, or they evolve dynamically based on the words people find for certain

constructs; alternatively, of course, a completely different rationale could be considered. In most cases, they should follow intuition or at least not go completely against it.

It should also be noted that sphere boundaries are generally dynamic, meaning that they can change over time. For example, if someone takes off their pants, they are no longer part of their sphere (in fact, objects can also be in two spheres at the same time (more about this in section 5.2.7). The world tree is thus subject to constant change and sphere-diagrams can strictly speaking only be applied to concrete situations. However, it can be assumed in practical application that sphere-diagrams have been set up in such a generally valid way that they can usually always be used.

Adaptation of Object-Diagrams

In addition to the tree notation of the sphere-diagrams, there still is the diagram form of the third pillar for the representation of ether-objects: the object-diagrams. They are not replaced by the tree notation, but can still be used to represent specific situations. The tree notation is used to represent specific areas of the world tree, while object-diagrams represent selected ether-objects, and possibly their sphere-connections. Object-diagrams thus represent situations, while sphere-diagrams of the tree notation represent a concrete sphere.

As in earlier pillars, this does not make old forms of representation obsolete; the concept of spheres merely supplements them. At the same time, the old forms of representation must be adapted to the new structures and the aspects relevant to them. This is also the case with object-diagrams. For the object-diagrams can also represent aspects of sphere-diagrams: namely the sphere-connections of objects. To do this, they must be extended by elements that make it possible to represent sphere-connections in object-diagrams. An example of what this looks like can be seen in Figure 5.7.

In this way, the sphere-connections that objects have to their parent or child-spheres, can be integrated into the object-diagrams. As seen in Figure 5.7, the sphere-connection to the parent is drawn with dashed lines and the connections to the children are drawn with solid lines so that they can be distinguished from each other.

The resulting object-diagram can be seen in Figure 5.8, it now also contains the object properties and has thus become a regular object diagram in which the spherical connections of the objects are now also shown. Of course, an old use of the object-diagrams is still possible if the spherical information is not needed. The same will apply later in the case of adapting the class-diagrams to the concept of spheres (see section 5.1.5).

5.1.5 Class-Typical-Sphere-Connection-Diagrams (Star Notation)

After the relationship between objects and spheres has been discussed, this section explains the relationship between spheres and classes. The class concept can be applied in particular to the sphere-connections of objects.

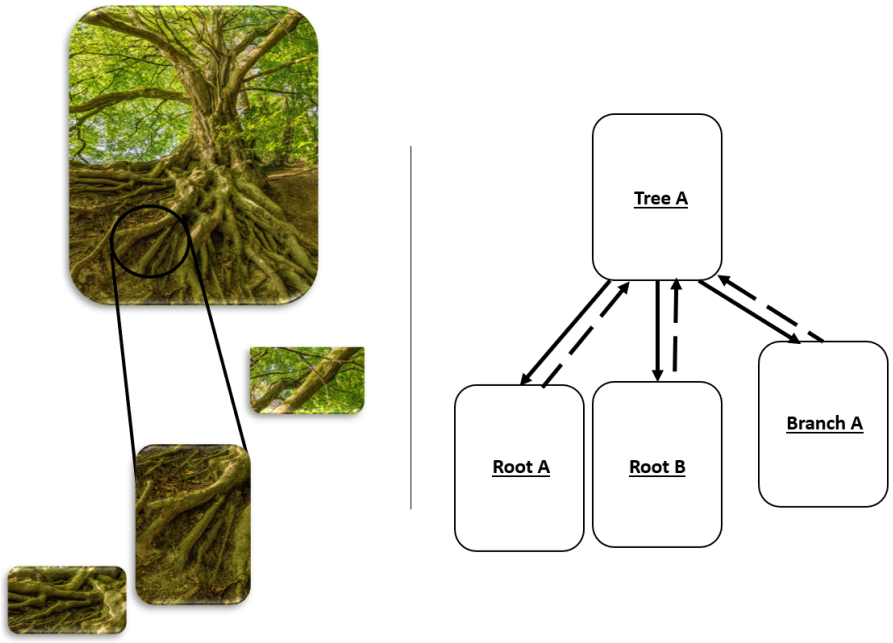

Figure 5.7: Tree ether-objects including their sphere-connections

Classes of Sphere-Connections

If one looks at the figure 5.9, one can quickly see that the tree spheres shown have similar child-spheres. Since some of them all belong to the same class, it is reasonable to assume that this class affiliation also affects the sphere-connections. For the case of trees, it seems that objects of the classes leaf, root and trunk are often child objects of a tree sphere. They seem to be class-typical-sphere-children of the tree class.

In Figure 5.10 one can see a simplified representation of these sphere-connections that are typical for the tree class. Such sphere-connections that are typical for the objects of a class are called class-typical-sphere-connections. They are represented with the so-called class-typical-sphere-connection-diagrams (or in short: sphere-connection-diagrams), which has already been used in Figure 5.10. The variant used is referred to as star notation and will also be retained in the following sections. Alternatively, sphere-connection-diagrams could be displayed with lines, which are easier to draw, however, this variant is not discussed in this book. Class-typical-sphere-connection-diagrams are not to be confused with class-diagrams, which will be dealt with later.

For illustrative purposes, Figure 5.11 shows the typical sphere-connections of villages, and Figure 5.12 those of dogs.

Adaption of Object-Diagrams

With the help of classes, the sphere connections of objects can now be described more precisely and be distinct from one another, that is: they can be classified. This means,

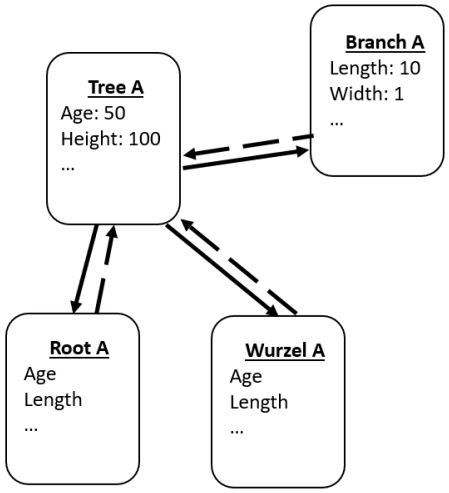

Figure 5.8: Ether-Object-Diagram showing the sphere-connections of the objects

object-diagrams need to be adapted again to the concept of class-typical-sphere-connections. Until now, all sphere-connections in object-diagrams were equal in their properties, that is: they were neutral and indifferent to which objects they connected. Now, as seen in Figure 5.13, the knowledge sphere-connection-diagrams can be used to add to the sphere-connections of object-diagrams the names of the class-typical-sphere-connections they belong to.

This creates the object-diagram shown in Figure 5.14, which contains more detailed information about which classes the objects that are accessible via a sphere-connection belong to and also contains the typical properties for objects. This additional information about the classes of the child objects will become important in the medial layers in order to be able to call child objects of certain classes more easily. As before, object-diagrams can still be used without the incorporation of the concept of class-typical-sphere-connections, as well as adding some but not all of them in the same diagram.

Are there Classes of Spheres?

With the introduction of class-typical sphere-connections, the question arises whether there are also classes of spheres, that is, whether there is a tree-sphere-class, up to the question whether there is a „World-Sphere-Class“ or „World-Tree-Class“, that is, a class for the spheres of universes to which also our universe's sphere would belong.

If one looks at the tree-spheres in Figure 5.15, one quickly sees that the children's spheres are similar. But is this also true of the children of their children? If this would be true, there could be world-tree of the classes of all things. However, this is where it gets complicated, as can be seen in the following example.

If one tries to create a visualization of the class of a sphere solely based on sphere-connection-diagrams, one quickly reaches the limit of this idea of representation as can be seen in Figure 5.16. The first generation of children works rather well, but in the

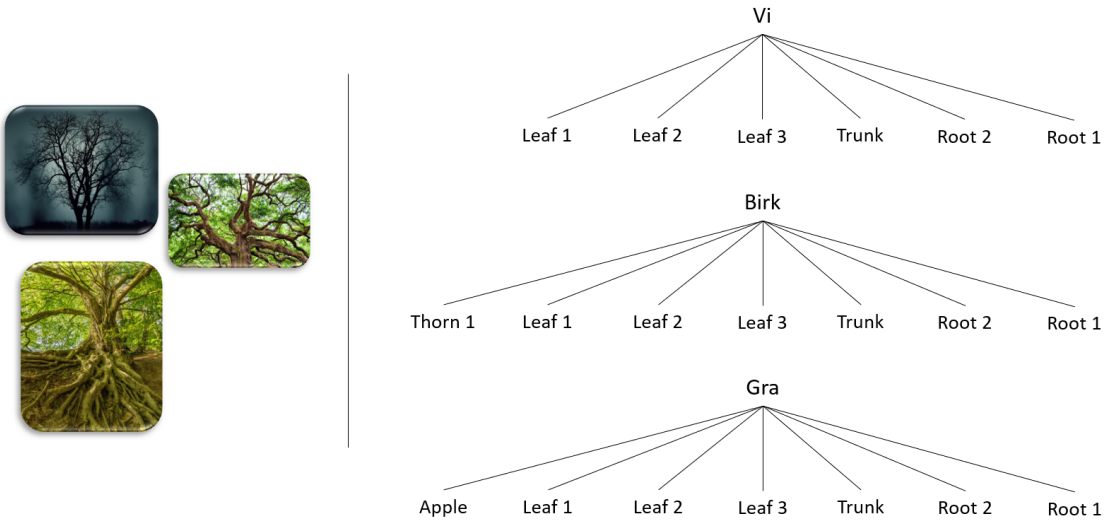

Figure 5.9: Sphere-Diagrams of three trees (tree notation)

case of fruits there could be multiple options depending on which kind of fruits the tree has. In case it is an apple tree, it would be clear that the fruits have stem, seeds and so on, but in the case of other fruits, this might be different. For example a chestnut has spines as a child class, but an apple does not.

This problem can be solved with the help of the class-hierarchy. Only by combining sphere-connection-diagrams with class-hierarchy-diagrams can a sphere-class be visualized. One would need to open a branch after fruits for every fruit-class that trees can have. Then, from these classes, one could continue this way, jumping to and fro between class-typical-sphere-connection-diagrams and class-hierarchy-diagrams. While this would be possible, it is hard to understand, hard to draw and doesn't have much practical application why it is not described here further. But it is useful to know that sphere-class can be visualized by a alternating combination of class-hierarchy and sphere-connection-diagrams.

Now, one could also ask: if there is a class-hierarchy of classes, is there one for spheres classes as well? Of course there is. There is also a sphere-class-hierarchy of all spheres, which as the root has the world-tree-class. However, this will not be discussed here either.

Adaptation of the Pillar 4 Class-Diagrams

Of course, class-typical sphere-connections can be integrated into class-diagrams as seen in Figure 5.17.

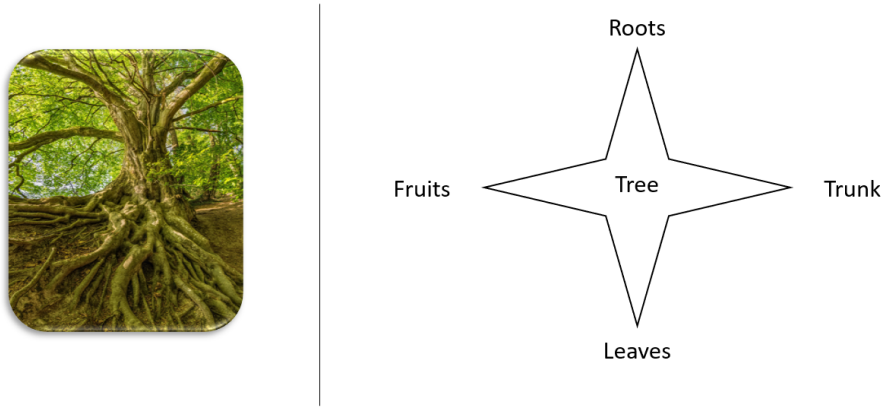

Figure 5.10: Tree sphere (star notation)

Figure 5.11: Village sphere (star notation)

Important Differences between Spheres and Classes, and their Concepts

Confusion can quickly arise, especially between spheres and classes, about the difference between the two concepts. These are described and clarified below.

Spheres and Classes It can happen that due to a fundamental misunderstanding about classes, the difference between spheres and classes is not understood and they are considered to be one and the same. This misunderstanding stems from the assumption that classes connect the objects that belong to them. This means that all trees of the class tree are connected to each other. However, this is wrong. All trees are independent of each other and, if they are, they are connected via spheres. The fact that they belong to the same class only means that they share similar properties and are thus related in nature. The class is an abstract pattern that can be recognised in the properties of ether-objects, while spheres are pattern in the constellation of ether-objects. In this regard, spheres are much closer to the concept of objects then they are to classes, because both are actual structures in the Ether, while classes are structures found when comparing

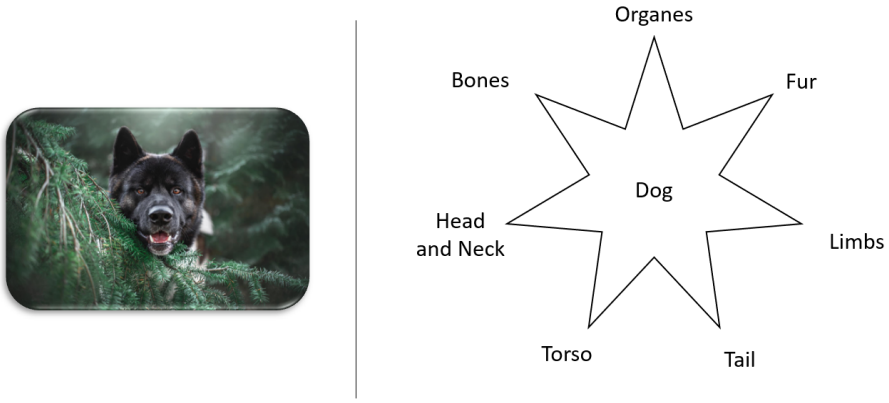

Figure 5.12: Dog sphere (star notation)

these structures, and therefore more like structural regularities.

Tree-Diagrams and Class-Hierarchy Both diagrams, the world tree as well as the class-hierarchy of all things can both be represented as a large tree. However, both are something fundamentally different. The difference consists, on the one hand, of the elements from which the diagrams are constructed, which were already described in the last section: the world tree consists of objects and their sphere-connections, the class-hierarchy of all things consists of classes and their relationships to each other. This also explains the difference between the two types of diagrams: sphere diagrams or the world tree describe the objects in the ether, while the class-hierarchy orders a collection of abstract patterns that can be found in the objects of the Ether and presents them in a hierarchy. Both diagrams are human-made and at the same time proven regularities of the universe; while the class-hierarchy describes the structure of things that do not actually occur in the ether, but only as patterns; and the world tree describes the structure of things in the ether. The diagrams thus basically do the same thing only on other element sets, or, on a different level of abstraction. Nevertheless, the class-hierarchy must not be confused with the world-tree.

Class-Hierarchy and Sphere-Classes-Hierarchy The class-hierarchy and Sphere-Classes-Hierarchy are doing the same thing but for different sets of elements. In the case of the class-hierarchy for object-classes and the case of the sphere-class-hierarchy for sphere-classes.

5.1.6 The Four Elements in the World-Tree

In pillar 3 the question, whether the four Element are part of the Ether was answered positive. While they were no ether-objects, they were somehow part of the Ether. But with the introduction of spheres and the world-tree, the question about their exact

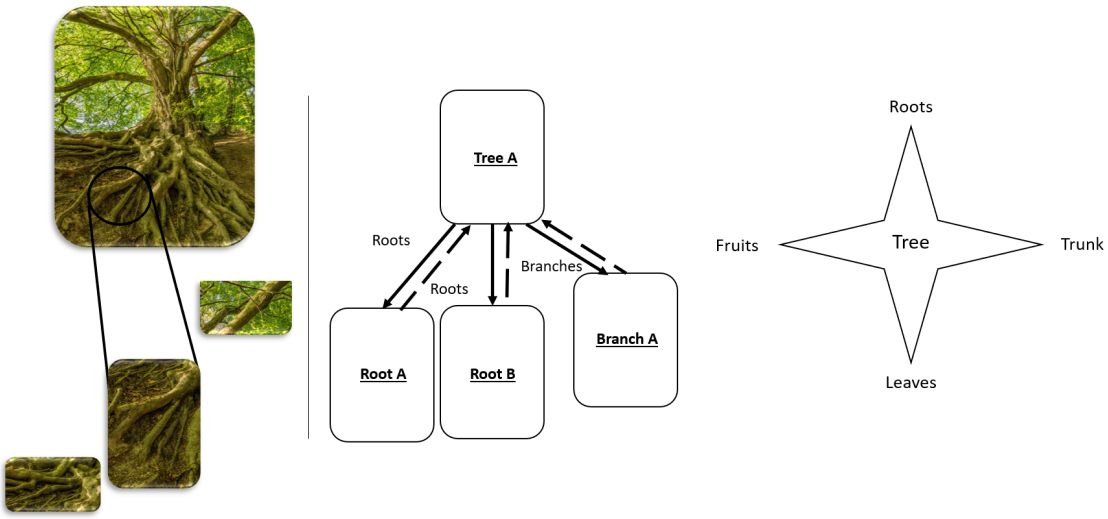

Figure 5.13: Use of sphere-connection-diagram of the tree-class on object-diagram of specific tree

whereabouts in the Ether are raised again, but more specifically: whether they are part of the world-tree or not. Well, they are not.

In pillar 5, the four elements are still part of the Ether, but are from now one considered to be separate from the structures discussed in the pillars of the ether-teaching. That means, they are not part of the world-tree. The latter may seem like an obvious contradiction, because they are part of the Ether and therefore probably also part of the structures of the Ether and because they can be experienced through ether-object or their physical representations, but it is assumed nevertheless for reasons of balance. When one assumes that the elements are not part of the structures in the Ether, it means that they cannot be accessed through them with the access to all things. This makes sure that one cannot, for example, access the fire-element through calling a bonfire and then use its powers theoretically from anywhere, which would be too powerful (the consequences of this are further discussed in section 5.2.5 of this pillar).

Another possibility is that the elements are considered as part of the structures in the Ether and can be accessed through them by calling but using their powers via calling requires so much energy that it is practically useless. This possibility, however, is not considered in this book.

5.1.7 Summary

In Figure 5.18 one can see all types of ether-structures summarized. In Figure 5.19 the relations through which they were derived are shown. All structure types, which have two arrows pointing on them were created by applying two other structures on one another, while structure types with only one arrow pointing on them were created by connecting structures of the structure type from which the arrow originates. And

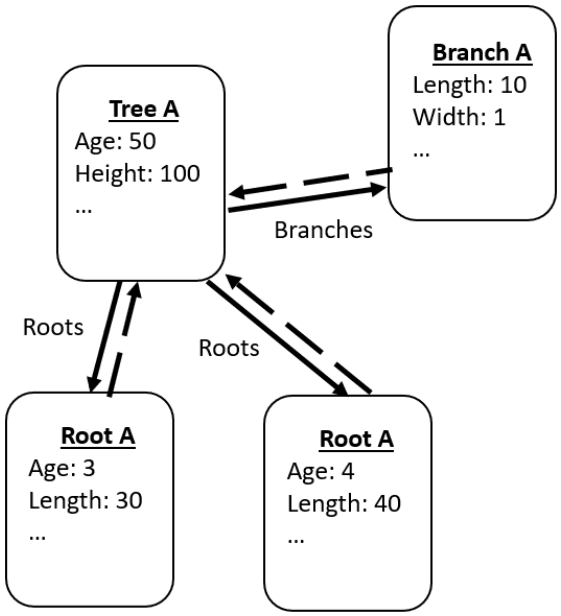

Figure 5.14: Object-Diagram with information about class-typical sphere-connections

finally, Figure 5.20, which of the pillars of the ether-teaching covered the respective types of ether-structures. One could argue that, because pillar 5 introduces more structure types than any other pillar, this pillar should be split up in more pillars. On the other hand, since two of the four structures were only scratched on the surface and have little practical relevance for the medial and worldly layer, the number of pillars could also stay the way it is now.

5.2 Medial Layer

This section shows how any thing in the Ether can be called via the sphere-connections of ether-objects. This form of calling objects via their spheres is called spherical calling and enables to call arbitrary things in the universe regardless of where they are thus achieving the goal of the „access to all things“.

5.2.1 Idea of Spherical Calling

The problem that spherical calling solves is the following: how can one, depending on the position one is in on the world tree, interact with a certain other thing on the world tree? This problem is first described in the next section and, building on this, how the target object can be called via the sphere-connections of the source-object.

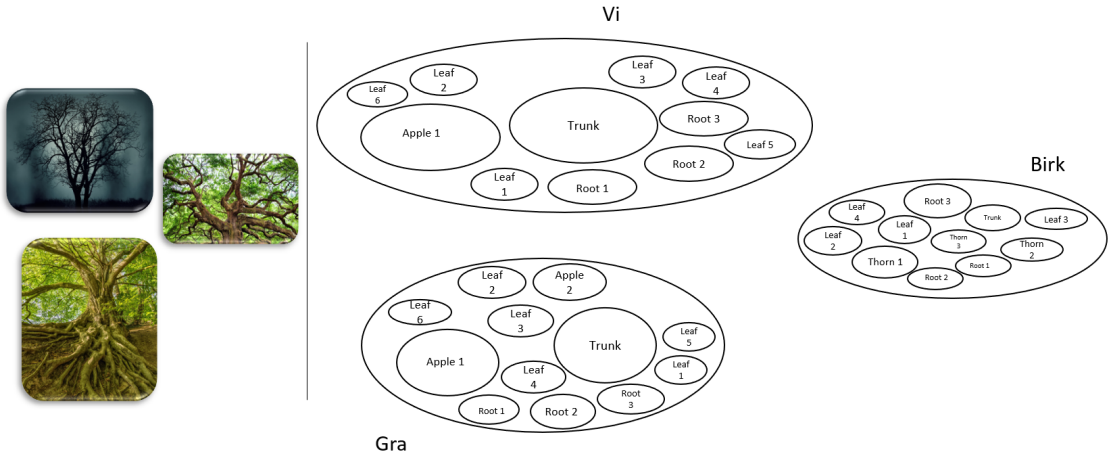

Figure 5.15: Sphere-Diagrams of three tress (sphere-notation)

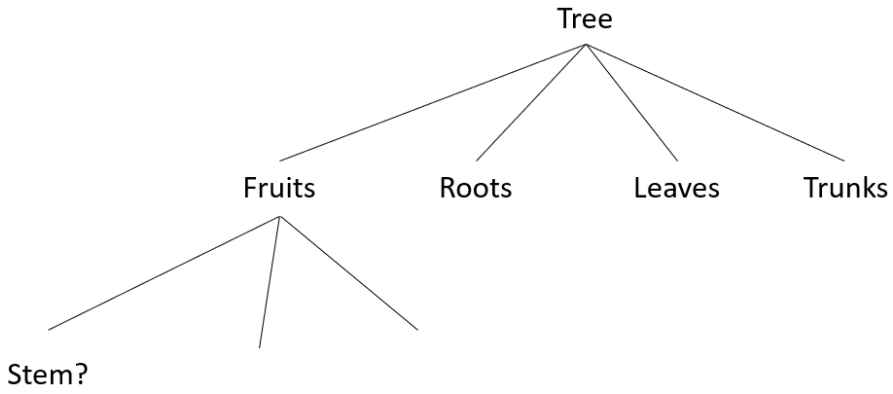

Figure 5.16: Failed attempt to create a sphere-class for the tree-sphere

Description of the Problem

As callers, we are also an object somewhere on the world tree and now want to call any other object. For example, we stand next to a tree and want to call the last apple to let it fall from the tree. With direct calling, this would be possible simply by touching the apple, but what if we cannot reach it?

The task of spherical calling is now to reach the apple object via its sphere-connections from the point of standing next to the tree. The sphere-diagram of the tree is shown in Figure 5.21, where the object to be called is marked in red and the position of the caller is marked with a smiley (next to the trunk).

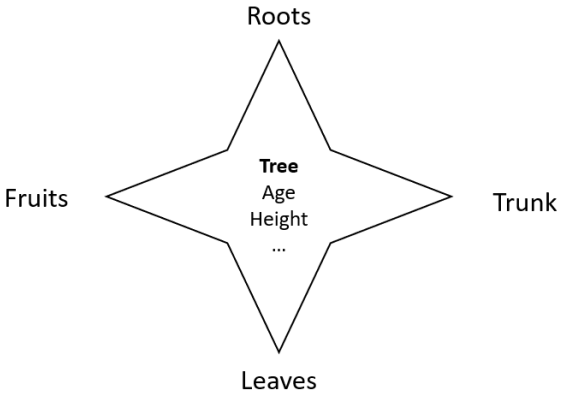

Figure 5.17: Integration of class-typical-sphere-connections in class-diagrams

Reaching the Target Object From the Caller-Object via Sphere-Connections

The idea of spherical calling is now to reach the target object via successive usage of the caller-object's sphere-connections.

Possible paths are shown in the figure 5.22. On the left side one can also see the previous form of access by touch. However, this does not work in this case because one cannot reach the apple. On the right side one sees the access via the sphere-connections of the trunk, which one touch and directly call, to the tree-object and down again to the apple, which is a child of it. In this way, one can not only reach the apple of a tree but any object in the universe.

It should be noted that every spherical call is preceded by a direct call to an initial object and only then the spherical path is described starting from this initial object. This initial object can also be one's own body.

The path through the world tree that is taken in spherical calling is always unambiguous, and therefore, for the first time, objects in the Ether can be unambiguously determined not only by their connection to their physical representation, but also by their position in the structure of the ether.

How does the Described Structure of Spheres in the Ether help in calling Arbitrary Things?

You can see that the introduction of spheres (and with them sphere-connections) is central for enabling the accessing of arbitrary objects in the Ether. Because due to the fact that all objects are connected to one another, there can always be described a unique path from one object of the structure to another. However, for the practical use of spherical calling, the hierarchy in which they are arranged is just as important. Without the right hierarchy of sphere-connections, it would not be as easy to call arbitrary things through spherical calling as shown above. The current hierarchy, called the world-tree, is arranged in the form of a tree and enables an easy to describe path over parent-connections to child-connections of ether-objects. This benefits the ease to call objects

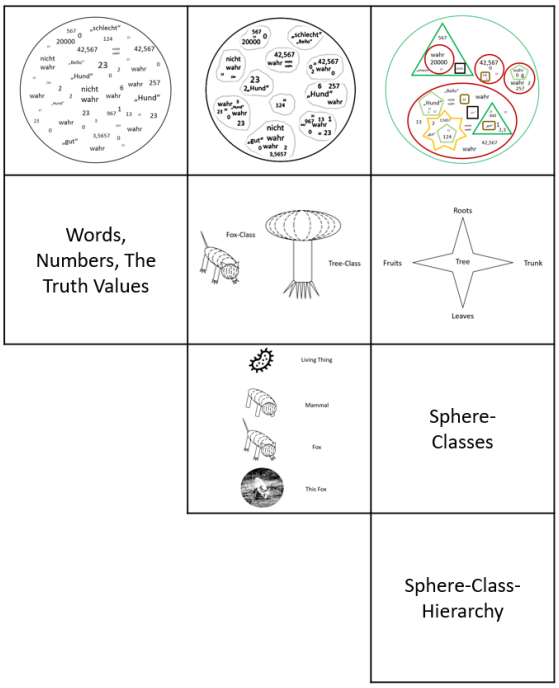

Figure 5.18: Visual Summary of all types of ether-structures

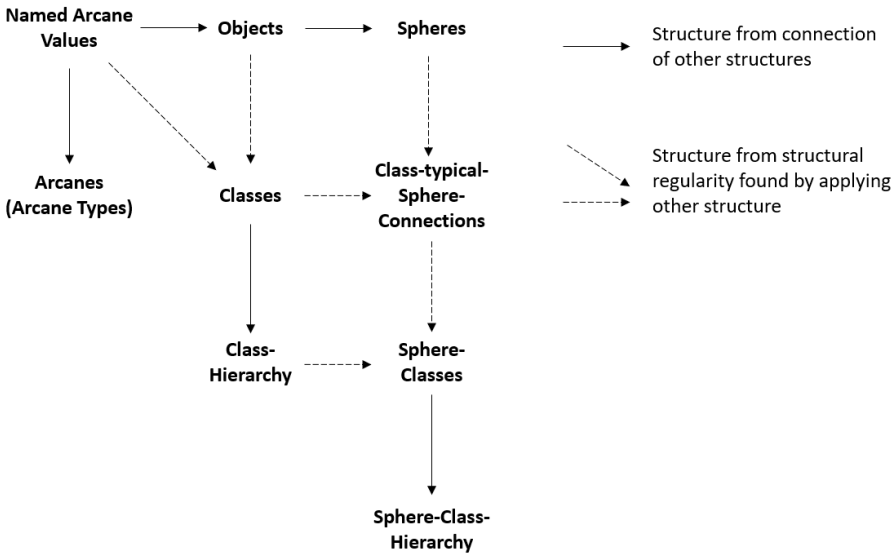

Figure 5.19: Relationships between all types ether-structures

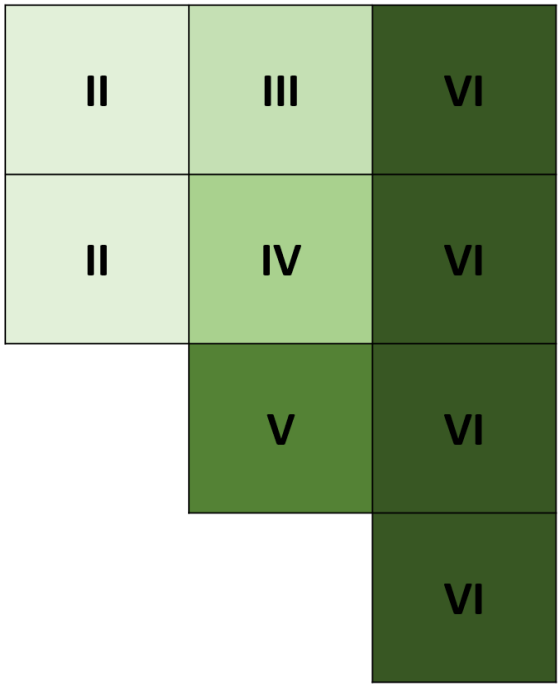

Figure 5.20: Covering of all the types of ether-structures by the pillars of the ether-teaching

through etherical calling immensely, because these kind of paths can be described with very short commands. Also, it is somewhat intuitive to go from the small to the big, back again to the small when trying to find a path through the Ether. If the structure would be a net instead of a tree, it would be harder to call the connections, one wants to take through this net, simply because with less structure there is less information from them, and the path-finding would be also much harder, because without a given structure it is very hard to even imagine the structure of the ether-objects for a given situation.

Thus, the spherical calling in the simplicity in which it was shown, is made possible by the introduction of spheres but more specifically, by the structuring of them in the form of the world-tree.

5.2.2 Spells for the Usage of Sphere-Connections

The following sections will now go into more detail on how the sphere-connections can be used to call any object. For this purpose, it must first be explained how child objects of an object can be called and how parent objects can be called. In the following, one will get to know the corresponding spells in order to use sphere-connections.

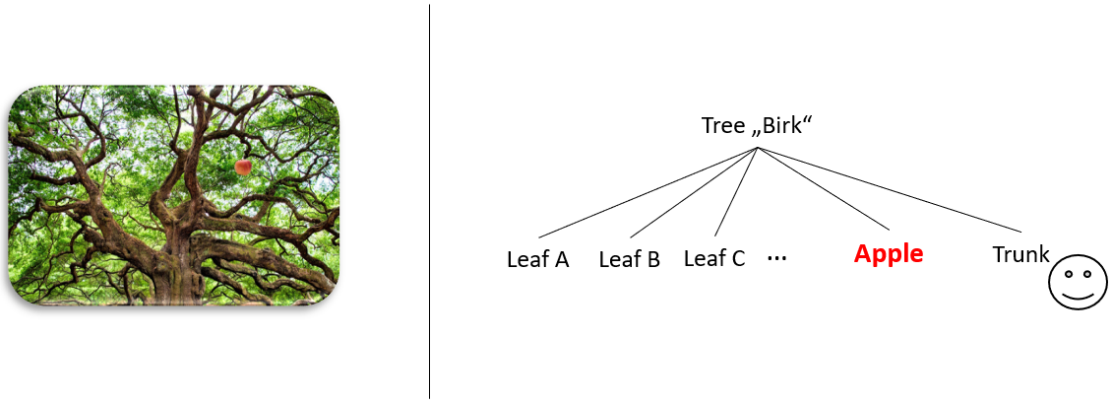

Figure 5.21: Sphere-Diagram (tree notation) of the tree „Birk“ with marked caller- and target-object

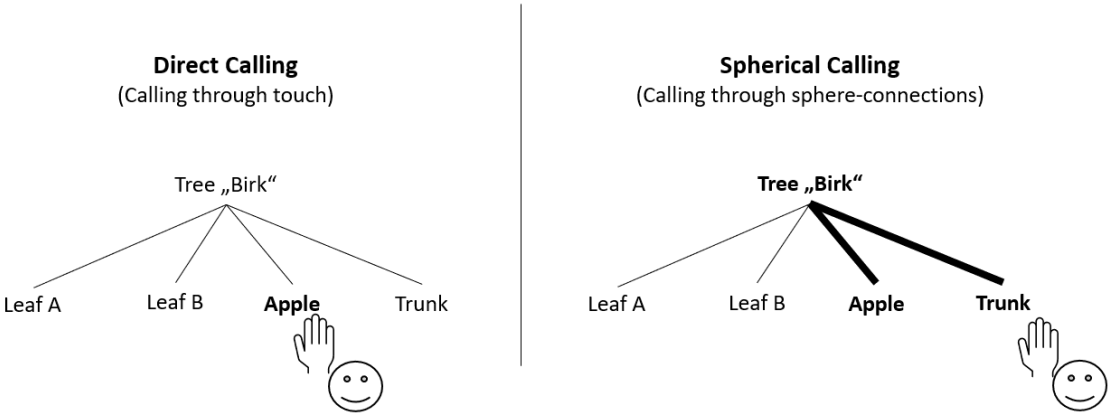

Figure 5.22: Difference between direct and spherical calling of an ether-object

From Single Spherical Calls To Spherical Calling-Paths

Basically, spherical calling is only a successive calling of child-/parent-objects through their sphere-connections. That means, that it is basically only necessary to know how to call a child-/parent-objects of an object through their sphere-connections and then use this knowledge to describe paths through the Ether. These kind of single usages of sphere-connections (also called the calling of child/-parent-objects) are however still considered as spherical calling.

Calling of Child-Sphere-Objects

Calling child-spheres is as simple as saying „children of the object“ and focusing on the children. One then gets all the child objects of the object.

In Figure 5.23 one can see that all children of the sphere can be called via the word

children. In principle, other words can also be used, one just has to concentrate on the children when pronouncing them. It should also be noted that the child word is used to call all children and not necessarily only children of one class, e.g. all roots. This means that it could be difficult to find out the relevant ones among all the children. How children of a certain class can be called easily is discussed in section 5.2.3. For one, the called children would need to be searched through, e.g. in an ether-dance, to find the object one is looking for.

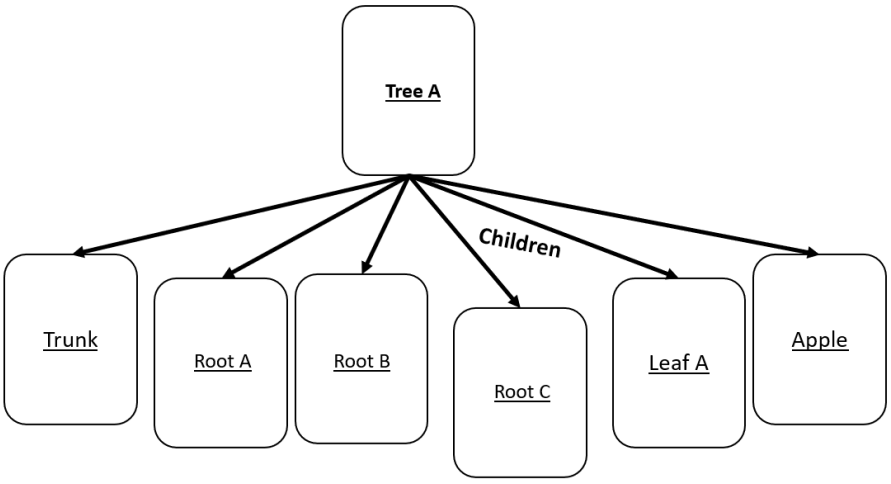

Figure 5.23: Available sphere-connections to call the child objects of tree A, from the tree A object

Calling of Parent-Sphere-Objects

Calling parent objects works just as easily as calling child-spheres. In these, for example, the word „parent“ is used, as in Figure 5.24. Unlike the child-spheres, which are usually multiple Ether-Objects, the parent Ether-Object is always only one Ether-Object. In the case of the all-sphere, it is the nothing-value. It is thus very easy, to chain calls to parent-objects together.

Practical Application

One could now call an object directly by touching it, for example the trunk. From this, calling the tree via its parent object and then calling its children, among whom the apple would be. The final difficulty is to find out among these children which one is the apple. Besides the fact that there could be several apples, there are also the roots, the trunk and all the leaf objects among the children. One would therefore have to search all the objects called for the apple object, for example with the loop of an Ether-Dance.

In the next section, another way to deal with this problem is described; there, it is described how, with the help of class-diagrams, the calling of the child-spheres can be

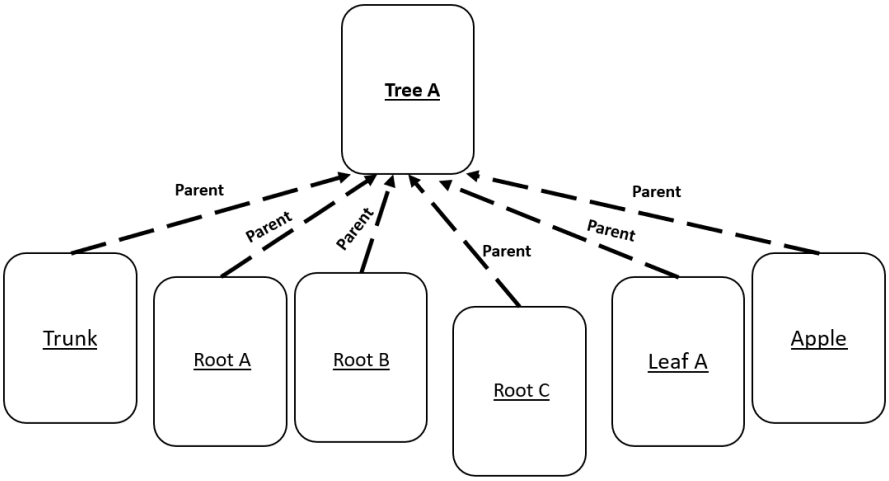

Figure 5.24: Available sphere-connections to call the parent object, from the respective objects

limited in such a way that only the objects of one class are called, for example, all apples. Calling the parent object does not have to be considered, as this is always only one object.

5.2.3 Use of Class-Typical-Sphere-Connections to better call Children of Specific Classes

While the last section already explained spherical calling thus enabling one to call all things, the problem remains that when calling the children of an object, one always receives access to all child objects. It would be more convenient to be able to limit the children received to objects of a certain class.

This can be done with the help of the class-typical-sphere-connections learned in the cosmic layer and their respective diagrams. From these, the typical sphere-connections of an object can be read and thus it can be found out to which classes of objects one can limit the response via sphere-connections. For example, in the case of a tree, one could limit it to fruits (based on Figure 5.10). Knowing that it is an apple tree, one would have to refer to the diagram for apple trees and could call all apples. Calling is done as described in the last section, except that one now call also the class into ones head and instead of all child objects one now only gets access to the child objects of the class used.

In Figure 5.25 one can now see how the sphere-connections of the tree object can be restricted with the knowledge about the class-typical sphere-connections in order to be able to call objects of a certain class exclusively.

It should also be noted that in certain situations even the child can be called directly. Since there are an infinite number of classes for each object, and a class can also describe exactly this one object, one can thus call exactly one of all children if one already has

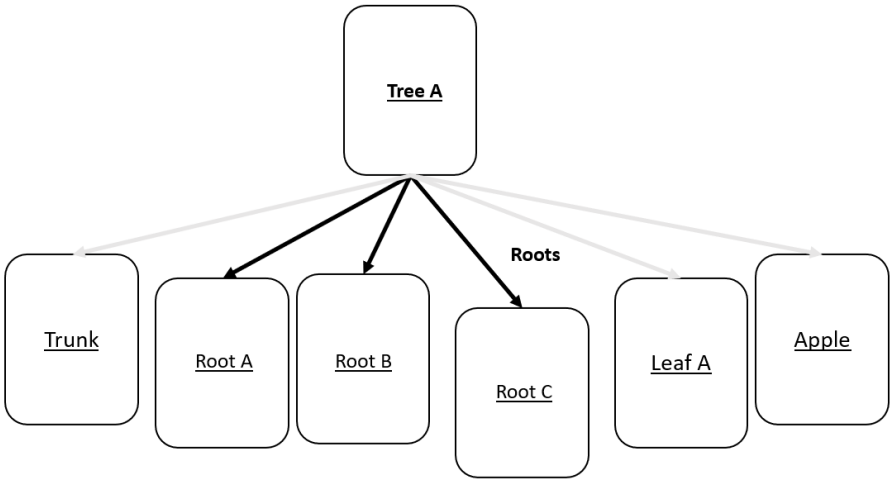

Figure 5.25: Use of the class-typical child-sphere-connections to call child objects of a specific class

a fixed image of it in one's head. One could therefore probably call the apple that one already sees with „this apple“ and call this apple to mind. The situation is different if, for example, one does not see the apple and only knows that it exists, because another person has told one so, or if one doesn't even know which specific apple it is, for example, if one wants to call the most beautiful apple of a apple tree at the other end of a meadow, which one knows exists, but which one doesn't know. In both cases, a direct narrowing down of the children spheres to this apple would not be possible. However, one could call the apple spheres and reach the apple in the first case this way, and in the second case, search for the most beautiful apple among them according to a fixed criterion.

This way, except for the two described cases, no Ether-Dance is strictly necessary to perform the most basic spherical callings in cases, in which one sees, or knows the object to be called. For example, the apple on the tree could simply be dropped in this way, even if one doesn't see it but knows which one it is; however, it quickly reaches its limits if, for example, one wants to call an object whose properties one does not yet know very well. In these cases, one has to search the child objects and this can only be done with the help of the language constructs of the ether-dances.

It should be noted that the class-typical-sphere-connections usually correspond to intuition and that in many situations one does not necessarily need to refer to diagrams; however, as often described before, it is sometimes useful to have sources to check and support this intuition.

5.2.4 Direct Calling and Spheres, or: The Sphere-Vortex

Until now, it has always been assumed that when calling directly, one simply has to touch a thing and can then call it easily. However, with the introduction of spheres, this principle has to be looked at again, as there are potentially an infinite number of

spheres converging at the place one touches, which one could call. In the case of touching a trunk, for example, one could call the trunk, but just as well the tree to which the trunk belongs, or the forest in which the tree is located. These possibilities that open up when touching any point in the universe are called the sphere-vortex. They will open up a new possibility to simpler call parent objects (seemingly directly even with multiple generations at the same time) and shed new light on the central way of calling of the last pillars, the direct calling.

For example, if you touch a tree, you can access all objects of the spheres converging at that point (to be seen in Figure 5.26). You can only be at one point in the sphere-vortex at a time, which means that you can only access one sphere-object at a time, as is usual in all the last pillars. For this, you have to concentrate on the corresponding object, e.g. on the particular forest you want to call when you touch the trunk. This does not change the usual way of direct calling, you just get more possibilities when you touch an object. Effectively, it also means that calling parent objects is not absolutely necessary if you only start spherical calling at a sufficiently „high“ (in the spherical sense) position.

This is complicated by the fact that it is harder to call spheres in both directions. It takes effort to call larger things, that is, things with a larger amount of objects in their sphere, but theoretically you can also call the universe-object if you concentrate hard enough. With concentration, one can reach the spheres into the small or large, e.g. from the tree to the forest to the universe, or in the other direction: from the tree, to the trunk, to the stem piece, to the cell, to the atom, etc. Concentration increases in both directions. The initial point is the object that intuitively comes to mind when we touch something. This object is called the „normal sphere-level“, „sphere-vortex-starting-point“, „sphere-pointer“, „sphere-vortex-handle“ or simply „sphere-handle“. In the case of a tree, this is probably usually the trunk. When you touch the door of a house it is the door and not the house or the wood that the door is made of, etc.

Towards the top, the spheres become larger, i.e. they contain more objects but there is always exactly one parent object; moving downwards is usually not that simple, because there are many objects to choose in between. Therefore, moving downward from the sphere-handle simply by concentration is not considered possible; it must be done by regularly spherical calling and searching trough the different levels of child-objects.

For a better understanding, Figure 5.27 shows the sphere-vortex within the objects of the world tree. One can see that the sphere-vortex is a line from the handle upwards and can be described by a series of clearly consecutive parent objects, while it splits downwards into more and more branching child objects.

The sphere-vortex is a sometimes intimidating as well as difficult concept to understand. Nevertheless with the fifth Pillar it becomes a natural part in the practical use of direct calling. As a metaphor, one can think of the sphere-vortex as a pond that one can swell and drain with concentration. It has a level that is „normal“ from the human point of view, but with concentration it can be moved above the normal level, or down, to create an ocean, or a puddle, or a molecule of water, or a universe full of water. The possibility of moving up the sphere-vortex, makes it possible to seemingly directly access a whole continent, planet, or: the universe itself.

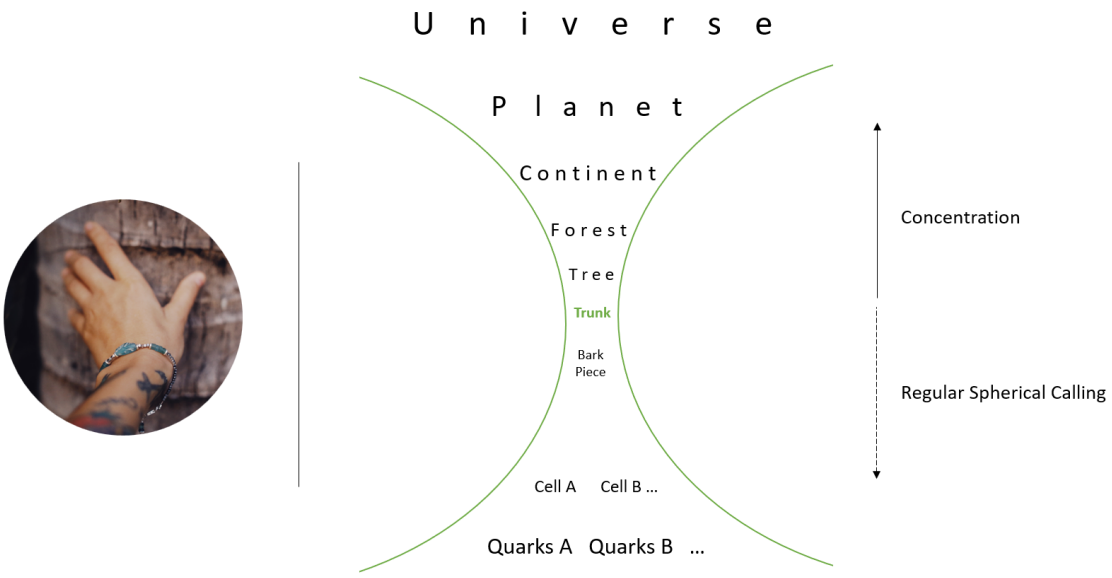

Figure 5.26: The sphere-vortex using the example of a tree trunk / when touching a tree trunk

In conclusion, the sphere-vortex puts the central way of calling of the last pillars in a different light, even though it might have practically always worked, because one always mentally concentrated on the vortex handle anyways and therefore it worked. However, some parts of last pillars calls can now not anymore considered as being direct calls, because they involved moving up the sphere-vortex, for example when calling the forest when touching a tree. This is from now on not considered direct calling anymore, because the usage of the sphere-vortex is not considered direct calling anymore, because it uses sphere-connections. Therefore, only the calling of the sphere-handle is from now on considered direct calling. Moving up the sphere-vortex is considered as being a part of spherical calling. It just makes the part of calling parent objects simpler. Thus, it is a simplified form of spherical calling of parent-objects.

The new definition of direct calling is therefore: the calling via touch while concentrating on the sphere-handle (which is the most intuitive way anyways, because it involves the least concentration of all spheres at that point). Moving up the sphere-vortex at the same time on the other hand is considered a direct call with a subsequent spherical call (potentially involving multiple single spherical calls).

5.2.5 Spheres and the Four Elements

The introduction of spheres has consequence for accessing the four elements on the one hand due to the revaluation to their position in the Ether, as already described in the cosmic layer and, because the question arises how the powers of them are performed on spheres.

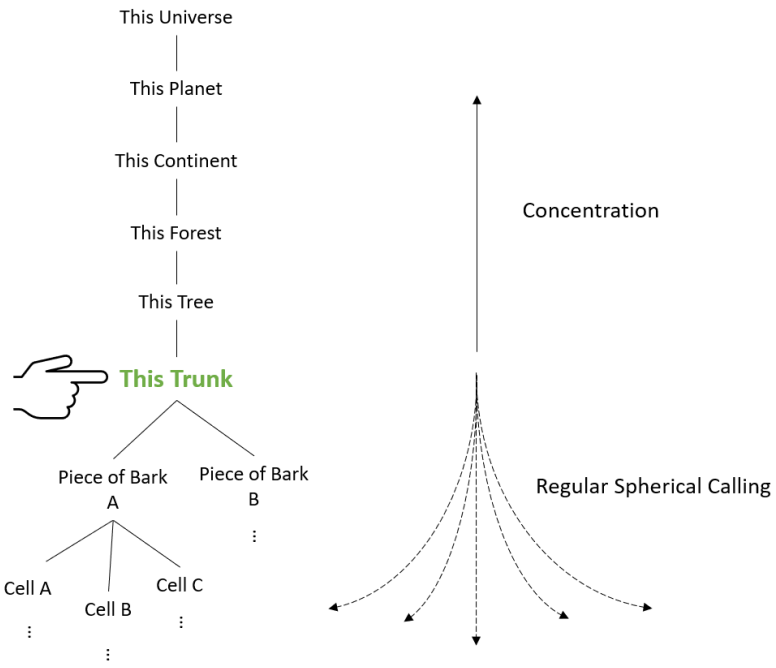

Figure 5.27: Ether-Vortex in the world tree

Active Manipulation in the Spherical Ether-Model

An element action always affects all children and childrens-children of a sphere and the sphere itself. For this reason, casting powers on large things (usually equivalent to spheres with many objects), costs significantly more energy than for small things (usually equivalent to spheres with fewer objects).

Accessing the Four Elements

Even though the elements are not considered as being part of the objects in the Ether, they are, because of their duality still part of the Ether. But they are also not everywhere in the Ether and cannot simply be accessed like arcanes. Instead, as defined in pillar 3, they exist in both worlds at once. They can be thought of as being attached to things in the form of states of them and cannot be called through them or any way at all. But, because of their duality, they can be accessed by touch. When one touches a thing, in which an element is present and one touches it, one directly also touches them in the Ether and is directly connecting to them mentally and is able to access them as if one had a connection to them via calling or as if they were part of one's own object.

The statement of the third pillar has therefore now to be taken back (or at least partly), because elements from now on cannot be called, but are accessed by touch, which is considered a sub-form of accessing/retrieving/recalling, or as something being somewhere between calling and accessing.

5.2.6 Spheres and Energy

Switching from spheres one touches to parent spheres to children consumes more energy, as longer distance must be covered than if one were to call an object directly. That means, the longer the sphere-connection-path one uses to call an object, the more energy is needed for spells that use that connection. This problem cannot be circumvented by using the sphere-vortex, in fact, there is technically no difference between „manually“ moving up parent-objects and to start at a „high“ position of the sphere-vortex. Using the sphere-vortex is just sub-form of spherical calling that is more convenient to use in terms of the calling steps to follow. Those are only one when using the sphere-vortex but several when using conventional spherical calling.

5.2.7 The Teaching of the World-Net

In the cosmic layer it was mentioned that objects can sometimes be in several spheres at the same time. This means, for example, that there are children who are actually no longer physically in the sphere, but are theoretically still available via sphere-connections, just as objects have two parent objects. Effectively, however, this makes no difference, since corresponding objects do not appear when calling the Ether and are thus non-existent from a practical point of view. This means that when the parent sphere is spoken of in this pillar, the parent sphere with the stronger energetic connection is always meant. All other connections are perceived as quasi-non-existent. The same applies to child-spheres. Only spheres are called child-spheres and are also called if these child-spheres also see the sphere under consideration as a parent sphere. This means that the world tree is more of an all-network in which every object is connected to every other object, only to some more and some less. Practically, however, this does not change the understanding of the Ether as a tree, which is why the concept of the all-net will not be pursued further here. So, in general, one can assume that an object is part of exactly one other sphere.

However, strictly speaking, all objects in the universe are directly connected with each other, only via energetic connections of varying strength. Thus the world tree can rather be understood as a world net. However, as humans naturally seek the effective paths of these energetic connections and these are reflected exactly in the structure that has already been described as a world tree, one can just as well stay with the idea. However, knowledge of the World-Net has a few practical advantages. If an element is removed from one sphere and transferred to another, it will still exist for a short time in both spheres. That is, a piece of the trunk of a tree could be stowed in the pocket of a pair of trousers and then used to do an ether-dance with it and drop the apple. It is not necessary to keep the hand on the trunk the whole time. After some time, however, the piece of the tree would have completely detached itself from the sphere of the tree and would no longer exist in it. This would also mean that it would no longer be possible to reach for the tree.

As the existence of an object becomes weaker from the sphere from which it was removed, the energy cost of accessing the parent sphere through that element also in-

creases. In fact, an object never completely disappears from a sphere it was once in, it just can't practically be accessed anymore because the energy would be too great. Thus the world tree is only an approximation. In reality, all ether-objects are connected to all ether-objects, but some with stronger and some with weaker connections to each other. One realisation should also be that what happens with direct naming is that the object, for example the apple is temporary also in ones own sphere for a short time.

Since it has practically no effect, it is of course still assumed/said that the order in the Ether is therefore still a (world)-tree, but it is useful to know about the world-net, for the practical advantages mentioned when calling things directly.

5.2.8 Summary

In summary, in Figure 5.28 one can see all the access distances of concepts in the Ether that have been learned in the last five pillars.

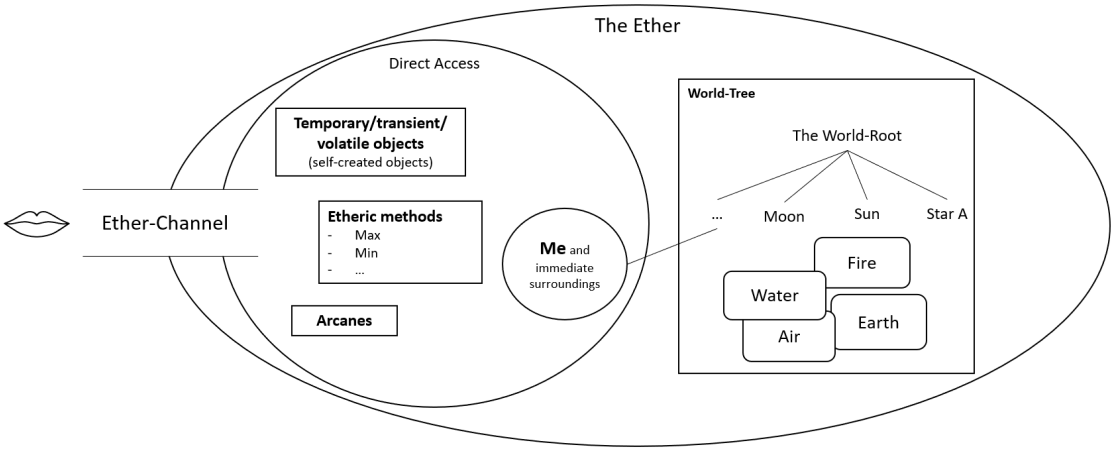

Figure 5.28: Access distances via the ether-channel

5.3 Worldly Layer

Since the new form of calling shown in the last layer is used in ether-dances the same way as direct calling, there is not much in this worldly layer. However, how loops can be used to search throw children is addressed and it also covers how spheres affect the creation of objects and classes.

5.3.1 Iteration through Children, Lists/Chains

Now it becomes clear why it was possible to iterate over all trees in the forest: because someone knew (presumably from a class-diagram) that the forest sphere has trees as typical child-spheres. Called Child-Spheres occur as the result of ether-commands in the

form of so-called object chains or object lists. That is, instead of one object, a chain of objects is returned. A loop does nothing but go through these objects one by one.

Chains can also be created artificially. As seen in ether-dance-protocol 5.1, chains of numbers can be created. Chains of objects can also be created. All artificial chains can be iterated over in the same way as the list of children of a sphere-object.

```
Syoombraaya

Let numberchain be a variable.

Let numberChain be a string of numbers 1 to 1000.

Let sum be a variable.

For each number in numberChain {
    Add number to sum.
}

Air: say sum.
```

Ether-Dance-Protocol 5.1: Using chains to add the numbers 1 to 1000

5.3.2 Creation of own Spheres

Artificial spheres can be created by connecting artificial ether-objects with each other. Natural objects cannot be connected with each other without creational powers.

What about creating natural spheres? Is that even possible. Sure. That's how the universe was created. Though as the same as for classes and objects, it is not usually possible for humans as there are creational powers needed for it.

5.3.3 Effects on the Creation of own Objects/Classes

What changes now when creating your own objects or classes?

Objects

Sphere-connections to other objects can be created. These can be treated like natural sphere-connections. Sphere-connections can even be created to real objects. However, the natural object will remain unaffected, i.e. the artificial object could not be called from it.

Classes

Class-typical sphere-connections can be created. These can be used again as if they were natural class-typical sphere-connections.

Creating Objects/Classes

In this pillar it becomes clear why it is virtually impossible to make artificial objects real, i.e. to create them, since that actually means that a sphere of them would need to be created (because every natural object must also have a sphere) and they contain an infinite number of other spheres and the energy required for this is also infinite.

6 Sixth Pillar

The sixth pillar contains no predetermined content and denotes the free discovery of the Ether and its possibilities on one's own, leading over to the study of actual "hard" computer science knowledge.