# Creating and Capturing Artificial Emotions in Autonomous Robots and Software Agents

Claus Hoffmann[1][0000−0002−9195−2917] and Maria-Esther Vidal[2][0000−0003−1160−8727]

[1] Research Group Robots and Software Agents with Emotions, Sankt Augustin, Germany,
hoffmann.claus@web.de
[2] TIB Leibniz Information Centre for Science and Technology, Hannover, Germany,
Maria.Vidal@tib.eu

**Abstract.** This paper presents ARTEMIS, a control system for autonomous robots or software agents. ARTEMIS is able to create and capture artificial emotions during interactions with its environment, and we describe the underlying mechanisms for this. The control system also realizes the capturing of knowledge about its past artificial emotions. A specific interpretation of a knowledge graph, called an Agent Knowledge Graph, represents these artificial emotions. For this, we devise a formalism which enriches the traditional factual knowledge in knowledge graphs with the representation of artificial emotions. As proof of concept, we realize a concrete software agent based on the ARTEMIS control system. This software agent acts as a user assistant and executes the user's orders. The environment of this user assistant consists of autonomous service agents. The execution of user's orders requires interaction with these autonomous service agents. These interactions lead to artificial emotions within the assistant. The first experiments show that it is possible to realize an autonomous agent with plausible artificial emotions with ARTEMIS and to record these artificial emotions in its Agent Knowledge Graph. In this way, autonomous agents based on ARTEMIS can capture essential knowledge that supports successful planning and decision making in complex dynamic environments and surpass emotionless agents.

**Keywords:** Autonomous Agents · Artificial Emotions · Agent Knowledge Graphs.

## 1 Introduction

Data-driven technologies in conjunction with smart infrastructures for management and analytics, increasingly offer huge opportunities for improving quality of life and industrial competitiveness. Semantic data models like RDF and OWL, have been proposed to represent knowledge in data-driven systems. Albeit expressive, the aim of these models is to represent entities, and the meaning of their features and relations. **The Problem and Proposed Solution.** Our research is guided by the following questions: i) how to create artificial emotions and ii) how to capture these emotions in a knowledge graph of an agent. For the creation of artificial emotions, we developed the ARTEMIS robot or software agent control system. The Component Process Model (CPM) of the emotion psychologist Scherer [24] and the Psi theory of the cognitive psychologist Dörner [8] provide the theoretical basis for ARTEMIS. Thus, ARTEMIS relies on a solid theoretical background, which we briefly introduce in section 3. Knowledge bases are essential components of autonomous robots or software agents. They are the cornerstone for their planning and decision-making. There are several ways to realize such knowledge bases. We suggest for this purpose a particular version and interpretation of knowledge graphs or Agent Knowledge Graph. This particular version and interpretation of knowledge graphs are designed to capture relevant knowledge for autonomous agents. It allows autonomous robots or software agents to mimic human "problem solving" in complex environments to a certain extent.

Knowledge graphs [10] are in general, becoming more and more important: Large companies such as Google, Facebook, or Microsoft now all operate their interpretations of knowledge graphs. The interpretation of Google's knowledge graph is optimized to enrich search results with semantic information. The interpretation of Facebook is designed to map social relationships. Our interpretation of knowledge graphs, we call Agent Knowledge Graph, is intended to support autonomous robots and software agents in planning and decision making in complex environments.

**Our Contributions.** We present the design of our robot or software agent control system ARTEMIS. The control system is capable of creating and capturing artificial emotions. In this paper, the basis for the creation of artificial emotions is the appraisal of interactions of the agent with other autonomous agents. Both cognitive processes and need processes are involved in realizing these appraisals. We demonstrate how ARTEMIS implements both types of processes. It is also vital that the ARTEMIS control system contains an Agent Knowledge Graph which stores the emotions and makes them available for later planning and decision-making processes. We further discuss the semantic and the episodic part of the Agent Knowledge Graph and provide a formalism to store artificial emotions. We have conducted a user study, and the observed results suggest that human test subjects consider the artificial emotions generated by ARTEMIS plausible. Further experiments evidence that knowledge about past artificial emotions contained in the Agent Knowledge Graph helps autonomous agents to successfully plan and make decisions in complex dynamic environments, outperforming, thus, emotion-free agents.

We have organized the paper as follows. In section 2, we will give an overview of possible application areas of ARTEMIS, using a motivating example. In section 3, we look at the foundations of how emotions are created and stored in an Agent Knowledge Graph. In section 4, we devise an Agent Knowledge Graph able to model the problem presented in the motivating example. In section 5, we discuss related approaches and their relevance to the ARTEMIS control system. In

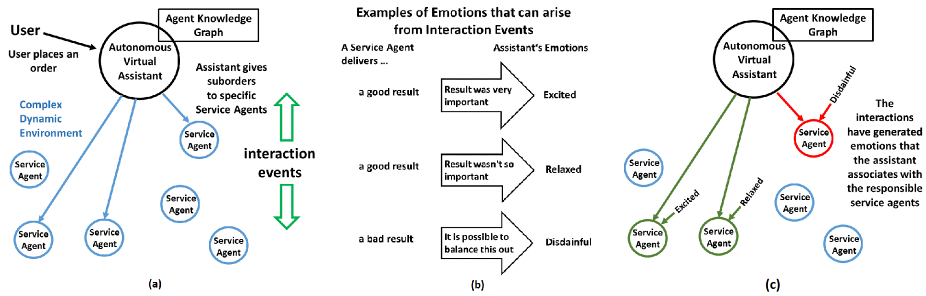


**Fig. 1. Motivating Example**. (a) An autonomous virtual assistant executes a user's order in a complex environment. For this purpose, the virtual assistant selects the most suitable service agents, which in turn are also autonomous. Several interactions take place between the user assistant and the service agents. (b) The autonomous virtual assistant appraises these interactions. These appraisals create artificial emotions. For reasons of comprehensibility, we present the created emotions in this figure as text. In truth, they are encoded as points in a space that correspond to these text representations. We will discuss this later in this paper. (c) The resulting artificial emotions are associated in the Agent Knowledge Graph with the corresponding interaction events and with the causative service providers. The Assistant thus gains an attitude towards the Service Agents over time, which provides useful information for its future selections of cooperation partners.

section 6, we present our experimental prototype and describe our experimental results. In Section 7, we discuss our conclusions and our future work.

## 2 Motivating Example

We motivate our approach using a typical situation that may be present in a wide variety of data-driven scenarios. Examples of application scenarios include the selection of a) machines in future 'Smart Factories', b) means of transport in 'Supply Chains 4.0', and c) information sources by an autonomous information broker in a 'decentral dataspace' like an 'Industrial Data Space'. The exemplary application scenario moves within the context of the so-called Service Web (see [9]). With this exemplary scenario, we can study principal problems of service selection without getting lost in the details of concrete application areas. The process of the exemplary application scenario is as follows (see Figure 1a). An autonomous agent takes on the role of an autonomous virtual assistant for its user. The autonomous virtual assistant accepts the orders of its human user. To execute an order, the autonomous virtual assistant searches its knowledge base for a suitable plan. A plan defines a list of steps. For each plan step, the autonomous virtual assistant has to find a suitable service agent that performs the step. Autonomous service agents offer their services at different prices and are differently trustworthy. The autonomous virtual assistant has to decide which service agent fits best with the current situation. The following conditions form the basis for the exemplary application scenario:

1. In complex dynamic environments (e.g., 'Industry 4.0' applications) conditions for cooperation with autonomous service agents, can change from time to time. Present cooperation partners may leave the environment of the autonomous virtual assistant, and others may arrive. As a result, the search for suitable cooperation partners becomes a permanent task.
2. The cooperation partners of the autonomous virtual assistant are autonomous themselves and try to maximize their outcomes. Therefore, the results of cooperation are often uncertain. The autonomous virtual assistant always has to expect that cooperation partners do not meet the agreements and provide results that do not fulfill expectations. This violation of expectations can have numerous reasons. One reason could be that cooperation partners are not capable of delivering their promised services. Another reason could be that they did not understand their mandate correctly. It is also possible that they deliberately do not execute the job correctly in order to gain an advantage for themselves.

These conditions provide the basis for a complex interaction between the assistant and the service agents. Appraisals of these interactions create corresponding emotions in the virtual assistant. For example, 'Excited' when something goes well in contrast to expectations (and the result was very important) and 'Disdainful' when a cooperation partner performs poorly (and it is possible to balance this out) (see Figure 1b). Through numerous interactions with the service agent, the assistant gains experience on the reliability of cooperation partners. Emotions are created and stored in the Agent Knowledge Graph of the assistant (see Figure 1c). Over time, the assistant gains essential knowledge that helps for future effective planning and decision-making. Conventional approaches without artificial emotions would only determine whether an interaction was successful or not. The emotion-based approach, on the other hand, is much more differentiated. Emotions summarize the agent's assessment of the entire underlying situation. An essential function of emotions is to adapt the planning and decision-making of an autonomous actor to a particular situation. Scherer [24] describes this as follows: Emotions are mechanisms that enable the individual to adapt to constantly and complexly changing environmental conditions" (from [24]). This applies to both current and remembered emotions.

## 3 Creating and Capturing Artificial Emotions

We first discuss how artificial emotions are created and their meaning. Then, we define the problem of capturing artificial emotions in Agent Knowledge Graphs.

### 3.1 Preliminaries

We present the ARTEMIS control system for creating artificial emotions. An Agent Knowledge Graph stores these created emotions. ARTEMIS resorts to the theoretical basis of the Scherer's [24] Component Process Model (CPM)

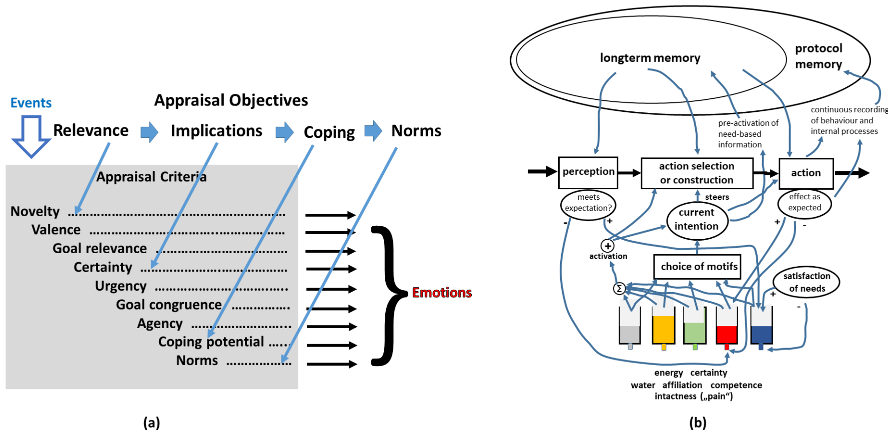


**Fig. 2. Fundamentals for the creation of emotions.** (a) Scherer's appraisal pattern [24] defines steps that have to be taken in order for emotions to form. For this purpose, the appraisal pattern defines appraisal objectives and appraisal criteria. The appraisal criteria subdivide the appraisal objectives. Emotions are created by applying the appraisal criteria to analyze events. However, Scherer's model does not describe any concrete mechanism on how appraisals should take place. (Eva Hudlicka inspired this picture, see [15]). (b) In ARTEMIS Dörner's Psi theory is the basis to realize Scherer's appraisal pattern. The Figure shows an overview of the structure of Psi (cut-out and own translation from [5]). The Psi theory defines an architecture of autonomous agents. This Figure primarily describes the interaction of need processes and cognitive processes. As this Figure shows, Dörner uses the concepts of motive and intention. We cannot discuss here in detail what exactly is meant by this. For the sake of simplicity, these terms could be replaced for the moment by the term goal.

and the Dörner's [8] Psi theory. In the Component Process Model, Scherer [24] defines an appraisal pattern for events. The Scheer's and Dörner's approaches are presented next.

**Scherer's Appraisal Pattern for Events** Scherer (e.g., [22–24]) describes the objectives of the appraisal process of events: "There are four major appraisal objectives that an organism needs to adaptively react to a salient event: (1) How relevant is this event? Does it directly affect me or my social reference group? (relevance); (2) What are the implications or consequences of this event and how do they affect my well-being and my immediate or long-term goals? (implications); (3) How well can I cope with or adjust to these consequences? (coping potential); (4) What is the significance of this event for my self-concept and for social norms and values? (normative significance)." (Scherer [24, p. 50]) Scherer subdivides the four appraisal objectives into more detailed appraisal criteria (see Figure 2a). The appraisal criteria include novelty, valence, goal relevance, urgency, goal congruence, responsible agent, coping potential, and norms (see Scherer [24, p. 51]). With his proposal, Scherer presents a theoretically sound appraisal pattern. However, he does not give any precise information on how to realize it. Scherer, however, gives hints on the boundary conditions for implementation; he also emphasizes the existence of needs and goals as essential prerequisites for the appraisals of events. Further, Scherer's criteria point out that computational agents who have no needs or goals cannot have real emotions (Scherer [24, p. 52]).

### 3.2 An Outline of Dörner's Psi theory

The Psi theory defines an architecture of autonomous systems (see, e.g., [6–8]). For experimental purposes, Dörner and his research team have realized the Psi theory as a computer simulation. The following scenario is the basis of this computer simulation: a virtual robot must protect its life on an island and at the same time, fulfill a task. The robot can alternatively be controlled by human test subjects or by Dörner's system. Dörner demonstrates that in a simulated environment, the realization of the Psi theory exhibits a similar behavior as the human test subjects.

Dörner's Psi theory shows that it is advantageous whenever needs are the basis of a control system of autonomous agents. For the realization of needs, Dörner proposes a simple model. Dörner models the 'need processes' using tanks that can have varying filling levels. If 'needs' are satisfied, the corresponding tanks are full. Each tank possesses an inlet and an outlet. Successes, reported by efficiency signals, raise the fill level of the tanks. Failures, reported by inefficiency signals, lower the level. Figure 2b shows needs represented by tanks. The actual fill level of these tanks (and thereby the actual strength of the needs) influences, for example, the arousal level as well as different behavior tendencies of the agent.

In the Psi theory, needs such as energy, integrity, or belonging activate goals that an agent must achieve in order to meet the needs. Sometimes some of these goals compete with each other and cannot be achieved at the same time. In this case, the control system must select a goal. The basis for a selection is how strongly the goals are activated and how difficult it is to achieve them. Further, there is a selection threshold that regulates the change of goals; it prevents an agent from switching between targets too quickly.

### 3.3 Creation of Artificial Emotions in ARTEMIS

We expand the knowledge of events that have taken place to include knowledge about the artificial emotions associated with them. To create emotions, we devise the ARTEMIS control system for autonomous agents (see Figure 3). Artificial emotions are the result of the agent's appraisals of events (see Figure 4a). First, we discuss the components of the ARTEMIS control system. As a starting point, we present the need system which is the basis of the ARTEMIS approach. This need system generates values for the parameters 'Pleasure', 'Arousal', and 'Dominance'. The parameter values generated by the need system are then mapped to the PAD cube of emotions and define emotions there (Figure 4b). Then, we

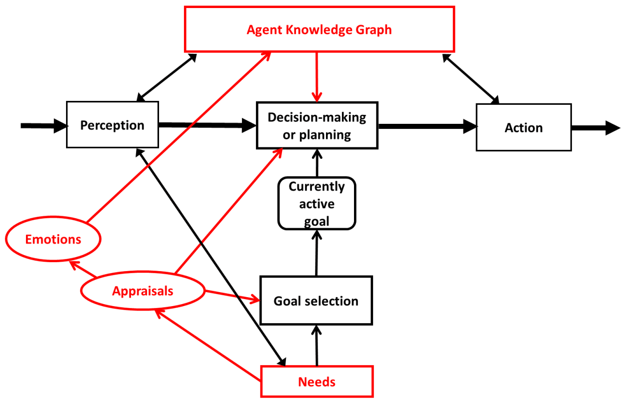


**Fig. 3. The ARTEMIS Architecture.** An overview of the structure of the ARTEMIS control system. The Dörner‘s Psi theory represents the basis for many essential components of this architecture (shown in the black parts of the Figure). In ARTEMIS, an Agent Knowledge Graph realizes parts of the long term and protocol memory of the Dörner's Psi theory (see Figure 2b). In contrast to the Dörner‘s approach, ARTEMIS has a specific appraisal component which appraises events based on both cognitive processes and need processes. Based on these appraisals, ARTEMIS creates artificial emotions. The results of these appraisals influence the 'decision-making/planning', or 'goal selection' components of the control system and modulate their effects (red arrows). An Agent Knowledge Graph captures these artificial emotions. (the red parts of the Figure are ARTEMIS specific realizations)

discuss how the Dörner's Psi theory can realize the appraisal pattern defined in the Scherer's theory.

**The Need System of ARTEMIS** Here, we present seven needs captured in the ARTEMIS control system. Why does our control system work with these seven needs as opposed to the Psi architecture? The answer is: Dörner uses the needs to be found in Figure 2b in the context of psychological research questions. We do not conduct psychological research but build autonomous agents within the scope of artificial intelligence applications. So, we have adapted the needs of the Dörner's Psi theory to ARTEMIS. The chosen needs are better suited to our research questions; they are as follows:

1. Preserve existence (e.g., execute orders, make sure that services can be paid),
2. Avoid pain (for robots it could mean to avoid structural damages, for software agents it could mean not spending too much money),
3. Be agile (change methods and maybe partners from time to time, get neither bored nor boring),
4. Affiliation (the need for robust social integration and a good relationship with others),

5. Certainty (being knowledgeable about the environment. Certainty results from the ability to explain and predict events based on knowledge about the environment),
6. Competence (effectiveness and the ability to deal with real-world problems),
7. Avoid damages (for robots, it means maintaining machines or buildings and not overloading machines; for software agents, it represents the ability of not making decisions that endanger the environment).

The ARTEMIS emotion model uses a dimensional theory of emotions [19]. Different emotions are characterized in terms of the three dimensions of a PAD cube (see Figure 4b). The three dimensions are described by the parameters: "Pleasure", "Arousal", and "Dominance". The values for these parameters are provided by the need system in the following way.

– Pleasure - Rising and falling of the strength of needs determine the level of pleasure.
– Arousal - A combination of the strengths of all needs determines the level of arousal.
– Dominance - The levels in the tank of the need for certainty and the need for competence determine the dominance of the agent.

Let *Eff*, *Ineff*, *Cert*, *Comp* be efficiency, inefficiency, certainty, and competence, respectively. Then, the former parameter values are defined as follows:

```
for (i= 1 to NumberNeeds)
  L(Need[i]):= W(Eff)*Signal(Eff) - W(Ineff)*Signal(Ineff)
  L(Need[i]):= Max(0,Min(1,L(Need[i]))
  Need[i]:=ln(1+L(Need[i]))
  Arousal := ln(1+(Need[i]) * W[i])
  Pleasure := W(Eff)*Signal(Eff)  - W(Ineff)*Signal(Ineff)
  Dominance := Need[Cert] * (1 - Need[Comp])
```

The strength of the needs depends on the corresponding levels (represented with the variable L) of the associated need tanks. The levels of the need tanks are calculated continuously. The level can only take values between 0 and 1. The efficiency and inefficiency signals have a weight W, which models the strength of their impacts (see [6]).

**Artificial Emotions Based on PAD Parameters** The combination of PAD parameters forms a cube, as shown in Figure 4b. The values of the parameters correspond to points in this cube. In the literature there are different proposals for mappings the points of the PAD cube to emotions. For our approach, we lean on the emotion mapping from Mehrabian [18,19]. Mehrabian considers only octants (subcubes) of the PAD cube. However, it makes perfect sense to name the extreme points of the PAD cube after these octants. So-called dimensional approaches make it possible to define vague boundaries of emotion categories. In our approach, the intensity of emotions increases from the center to the edges

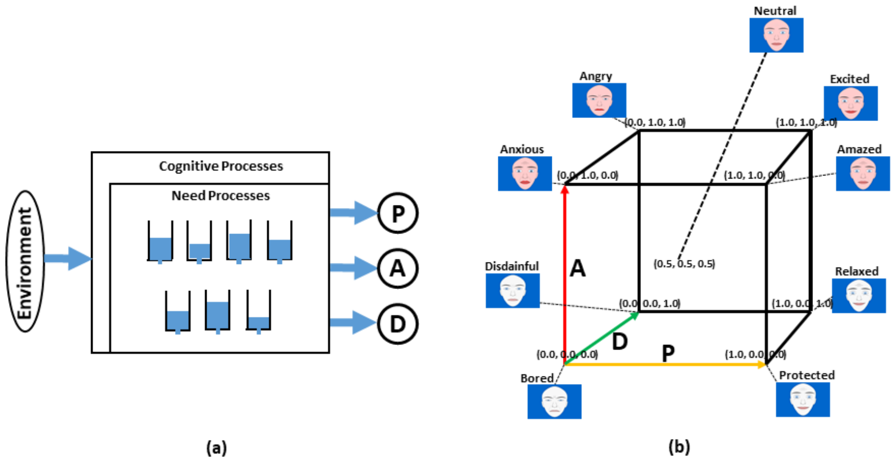


**Fig. 4. Creation of Artificial Emotions in ARTEMIS.** (a) Dörner's Psi theory is the basis for cognitive and need-based evaluations of events. In this way, ARTEMIS realizes the Scherer's appraisal pattern. The appraisal processes generate values for the parameters (P)leasure, (A)rousal, and (D)ominance. (b) The values of these parameters determine points on or in a cube. ARTEMIS maps these points to artificial emotions. Mehrabian's dimensional emotion theory [18, 19] inspires this mapping.

of the cube. According to Peter Gaerdenfors [11, p. 48], the PAD cube equips emotions with meaning. The PAD parameters receive their values through need processes; as a consequence, the artificial emotions defined by ARTEMIS, finally, receive their meaning through need processes. An essential aspect of our approach is that the representations of the artificial emotions of ARTEMIS are not meaningless strings, but are grounded in the corresponding need processes.

**Scherer's appraisal pattern in ARTEMIS** Some appraisal steps that Scherer defined (see Figure 2a) can be realized directly or indirectly by the needs of ARTEMIS. Cognitive processes are the basis for these appraisal steps.

- 'Certainty' can be realized directly based on the 'need for certainty'. The filling level of the 'certainty tank' provides the necessary information for this purpose.
- The feature 'Coping potential' follows from the filling levels of the tanks 'certainty' and 'competence'.
- The filling level of the 'Preserve Existence' tank can be used to deduce the parameter 'Urgency'.
- Novelty is a comparison of the current event with the agent's expectations. Dörner's Psi achieves this by pre-activating need-based information in long-term memory.

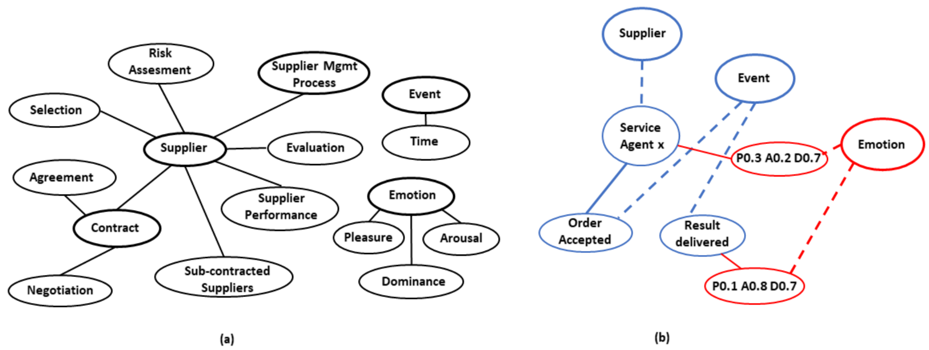


**Fig. 5. Realizing the Agent Knowledge Graph.** The Agent Knowledge Graph contains both semantic and episodic information. The semantic part of a Knowledge Graph contains general knowledge about the environment. The episodic part of the Agent Knowledge Graph contains information about specific entities and events that have occurred and the artificial emotions associated with it. (a) The semantic knowledge of the virtual assistant for the example presented in the motivating example (section 2) looks like this. The template for this knowledge comes from Graupner [12]. It shows an Agent Knowledge Graph for the process scenario of supplier management. We added the concepts 'Event' and 'Emotion'. (b) Information about instances, actual events that occurred, and associated artificial emotions are represented in this part of the Agent Knowledge Graph.

– A determination of valence, goal congruence, goal relevance, agency, and norms require cognitive evaluations. Due to a lack of space, this is not explained in this paper.

## 4 The Agent Knowledge Graph

Previously, the process followed by ARTEMIS to create artificial emotions was described. In this section, we describe how ARTEMIS captures artificial emotions in an Agent Knowledge Graph. The Agent Knowledge Graph has a semantic and an episodic part. The semantic part of the Agent Knowledge Graph serves to classify information. A protocol of the events that take place represents the basis for the episodic part, such as interactions between the virtual assistant and the service agents are established (see the motivating example in Section 2). An Agent Knowledge Graph is an essential part of the ARTEMIS architecture (see Figure 3). In Dörner's theory, a self-defined type of neural network is defined to realize the memory of Psi. However, for practical reasons, we have decided that ARTEMIS uses established methods of knowledge graphs for this purpose.

### 4.1 Realizing Semantics in an Agent Knowledge Graph

Knowledge specified by Graupner [12] forms the basis for the assistant's semantic part of an Agent Knowledge Graph. This knowledge provides the necessary conceptual information for the assistant about the problem area. We show an example of the organization of the semantic part of the Knowledge Graph in Figure 5a. Since the Graupner's example is a supplier management system, the focus is on the supplier concept. Suppliers are related to many other concepts such as 'Risk Assessment', 'Evaluation', 'Supplier Management Process', 'Selection', 'Supplier Performance', 'Sub-Contracted Suppliers', and 'Contract'. The 'Contract' concept in turn is related to the 'Negotiation' and 'Agreement' concepts (see Figure 5a). In ARTEMIS, we extend this model with the concepts 'Event' and 'Emotion'.

### 4.2 Episodic knowledge in an Agent Knowledge Graph

While semantic knowledge specifies what the environment of an autonomous agent consists of, episodic knowledge describes what is going on in its world. In addition to the abstract semantic knowledge, a virtual assistant possesses episodic knowledge, such as knowledge about specific service providers, events, or artificial emotions (see Figure 5b). The interactions of the assistant with the service providers create episodic knowledge; this episodic knowledge is enriched with emotional information. Over time this emotional information leads the assistant to develop a subjective attitude towards the service providers in its environment. This subjective attitude supports the assistant in future problem situations and enables the selection of appropriate cooperation partners in this dynamically complex environment. In our running example, the information stored in the episodic part refers to the abstract concepts supplier, event, and emotion. Information about a specific service provider (here, Service Agent x) is recorded. Events assigned to the specific service agents are 'Order accepted', and 'Result delivered'. Mainly the Events 'Service delivered' are predestined to generate artificial emotions, which are then also stored in the episodic part of an Agent Knowledge Graph.

## 5 Related Work

Research in the field of computers and emotions currently focuses on the recognition of user emotions. For example, one tries to recognize emotions in texts (emotional analysis), human faces, or the language (see [13]). This research direction has already achieved significant results. Emotion analysis will be essential for machines to react appropriately to their human users' emotions. Such analyses are, therefore, crucial for the next step in human-computer interaction (HCI). Here, however, the approach presented in this paper is not the recognition of human emotions. Instead, the focus is on creating and memorizing artificial emotions in autonomous agents. These artificial emotions help to adapt autonomous robots and software agents' behavior to the respective environment. It is also crucial that the communication of these artificial emotions (e.g., by face, voice, or gestures) can help users understand the system's decisions and actions. The basis of this understanding is that human users can often imagine that they probably would have had similar emotions in similar situations and that they would have acted or decided similarly on this foundation. The approach presented in this paper, therefore, has two results. On the one hand, it serves to improve the performance of autonomous agents. On the other hand, it is also a contribution to the research area of HCI.

There are diverse approaches to create artificial emotions. Marsella et al. [17] give an overview of this. So far, most approaches for agents with artificial emotions use the model from Ortony, Clore, and Collins (OCC) [20]. However, the OCC approach relies on a purely cognitive assessment of events. On the other hand, we rely on the approach of the emotion psychologist Scherer and the cognitive psychologist Dörner. Scherer's research has shown that realistic judgments must both consider cognitive processes and need processes. According to Scherer, this is more promising for creating realistic artificial emotions for autonomous actors. Dörner's theory can be the basis to realize the appraisal pattern required by Scherer. With ARTEMIS, we now present a model that realizes these vital requirements from these two researchers.

The problem of capturing emotions has been tackled in the literature as a data analytics problem, and different formalisms have been proposed for knowledge representation to solve this problem effectively (e.g., [1, 14]). Additionally, Chekol and Stuckenschmidt [2] present a formalism to represent temporally in probabilistic knowledge graphs. Albeit expressive for event representation or for performing data analytics, none of these approaches can naturally represent the semantics encoded in the emotions. In our approach, the artificial emotions of an autonomous agent have a meaning.

## 6 Experimental Study

We implemented a prototype of the virtual assistant to assess the performance of ARTEMIS. We aim to answer the following research questions (RQ): **RQ1)** Can a virtual assistant generate artificial emotions that are plausible for human test subjects? **RQ2)** Can captured artificial emotions make the virtual assistant more efficient? The experimental configuration is as follows:

**A Synthetic Virtual Assistant** We implemented a synthetic scenario to evaluate the feasibility and behavior of ARTEMIS. A virtual assistant is created, which can call 100 service agents. In this scenario, 50 of these service agents are somewhat reliable, and 50 are rather unreliable without the virtual agent having any information about them. The virtual assistant selects its cooperative partners from this pool. It initially selects its cooperation partners at random following a uniform distribution. With a large number of interactions, it can use the artificial emotions generated during individual interactions and captured in its Knowledge Graph. The assistant executed 300 test runs.

**Implementation** We realize the virtual assistant by a dynamic system based on difference equations; the system is implemented in Python 3.5.3. An Agent

Knowledge Graph is modeled as an RDF graph using RDFLib [21]; in order to realize the episodic part of the Agent Knowledge Graph, events are described based on the ontology LODE [16].

**Evaluation metrics:** We measure the performance of ARTEMIS in terms of time; it represents to the elapsed time between the submission of an order to the virtual assistant and the completion of the order; it corresponds to the absolute wall-clock system time reported by the Python `time.time()` function.

**User Evaluation:** We conducted an evaluation where 30 human test subjects evaluated the virtual assistant in the above described synthetic scenario. We asked the participants to asses the plausibility of the artificial emotions that the virtual assistant created when fulfilling an order of the user. We presented nine scenarios to each test subjects. The basis for the scenarios is the motivational example (compare section 2). The test subjects were asked to assess the plausibility of the created artificial emotions within the scenarios. The artificial emotions were presented to the test subjects in both pictorial and textual form.

### 6.1 Results of the User Study

All the users answered the questionnaires independently and evaluated the presented artificial emotions; 270 evaluations were thus available. The test subjects stated in seven evaluations that they could not understand the artificial emotions presented "very well" or "well". In a later optional interview, five of them stated that in one of the scenarios, they would tend to the emotion "indifferent" rather than to the emotion "disdainful". In nine evaluations, the test persons indicated that they could not decide. In 254 assessments, subjects indicated that they could understand the artificial emotions presented well or very well and that they could imagine having similar emotions in similar situations. Additionally, the performance of the virtual assistant was evaluated in terms of time; the behavior of the virtual assistant was observed without and with remembered emotions. The virtual assistant was executed for 300 runs. As a result, we observed that the effectiveness– in terms of average time– was enhanced by up to 40% whenever the virtual assistant was able to fall back on remembered emotions from earlier test runs for its decision-making process.

**Table 1. Results of the User Evaluation.** Artificial emotions are evaluated in a user study; they are represented both as text and as images. In 53.33 % of the cases, the users understand very well the emotions while 40.74 % just understand them well.

| User Question | Positive Answers | Percentage Positive Answers % |
|---|---:|---:|
| I fail to understand at all | 3 | 1.11 % |
| I fail to understand | 4 | 1.48 % |
| I cannot decide | 9 | 3.33 % |
| I can understand well | 110 | **40.74** % |
| I can understand very well | 144 | **53.33** % |

**Discussion** As far as we have been able to investigate this, the proposed approach opens up productive and promising research and application fields. These initial results suggest that the approach implemented in ARTEMIS works and enables autonomous agents to reach their goals faster. It turns out that remembered artificial emotions are helpful for successful agent planning and decision making in complex environments. Furthermore, the results of the experiments show that the approach can help to make decisions of a computer system more plausible for users. The system can thus make clear its internal situation on which it grounds its decision making. However, further experiments considering different scenarios and types of goals are required to thoroughly asses the pros and cons of a model able to create and capture artificial emotions.

## 7 Conclusion and Future Work

We have tackled the problem of creating and capturing knowledge about artificial emotions. To generate artificial emotions, a suitable model, as well as a system that implements this is required. For this purpose, we have developed the ARTEMIS control system for autonomous agents with artificial emotions. The Psi theory of the cognitive psychologist Dietrich Dörner is the basis of essential components of the ARTEMIS control system. We added a specific appraisal and an emotional component. In ARTEMIS, event appraisals create artificial emotions. The appraisal pattern described by the emotion psychologist Klaus Scherer is the basis for this. However, Scherer does not provide any information on how to realize this appraisal pattern. In ARTEMIS, we use Dörner's Psi theory to implement Scherer's appraisal pattern.

For capturing knowledge about artificial emotions, we developed the concept of an Agent Knowledge Graph as a formalism for empowering autonomous robots and software agents with this knowledge. In addition to knowledge about facts, Agent Knowledge Graphs also represent subjective knowledge of individual autonomous agents. Captured artificial emotions form this subjective knowledge. Artificial emotions are collected together with other information (e.g., point in time) about events in Agent Knowledge Graphs. As time goes by, the captured artificial emotions form a subjective world view of the agents. This subjective world view helps agents to plan and decide successfully in complex dynamic environments. Artificial emotions of autonomous robots or software agents based on ARTEMIS have a meaning. This meaning can be derived as follows. According to Peter Gaerdenfors [9, p. 48], the PAD cube equips emotions with meaning. The PAD parameters receive their values through need processes; as a consequence, the artificial emotions defined by ARTEMIS finally get their meaning from the underlying need processes.

We empirically investigated the behavior of ARTEMIS in a synthetic scenario in which a virtual assistant had to select suitable cooperation partners from a pool of 100 service agents. In three hundred interactions, the virtual assistant developed an emotional attitude to many of these service providers. We have evaluated the feasibility of the artificial emotions the assistant created by a group of thirty human test subjects. The test subjects confirmed that most of the artificial emotions generated by the virtual assistant were comprehensible to them. Furthermore, we measured the execution time of the virtual assistant in a setting with and without remembered artificial emotions. The results of the evaluation reveal that the virtual assistant could reach their objective on average in 40% less time than the configuration without remembered artificial emotions. The observed results reveal the potential of the proposed approach. Nevertheless, we recognize that this formalism is still in an initial phase and that further studies are required to provide a general approach that can represent artificial emotions in any scenario. The development of general approaches able to capture artificial emotions while manage conflicts that may arise in different agent interactions are part of our future work.

**Acknowledgments** We thank Christoph Lange from Fraunhofer Institute FIT for his valuable comments regarding our work. This work has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 822404 (QualiChain).

## References

1. Candrlic, S., Katic, M. A., Pavlic, M.: A system for transformation of sentences from the enriched formalized Node of Knowledge record into relational database. In: Expert Syst. Appl., 115: 442–464 (2019)
2. Chekol, M. W., Stuckenschmidt, H.: Towards Probabilistic Bitemporal Knowledge Graphs. In: Companion of the The Web Conference 2018 on The Web Conference 2018, WWW 2018, Lyon, France, April 23-27, 2018, p. 1757-1762 (2018)
3. Declarative Memory, ScienceBeta, https://sciencebeta.com/declarative-memory/. Last accessed 2020-01-14
4. RDFLib Homepage, https://rdflib.readthedocs.io/en/stable/. Last accessed 2020-01-14
5. Dörner, D., Schaub, H., Detje, F: Das Leben von PSI. Über das Zusammenspiel von Kognition, Emotion und Motivation - oder: Eine einfache Theorie komplexer Verhaltensweisen. In: von Lüde, R., Moldt, D. Valk R. (eds.) Sozionik aktuell, volume 2, Informatik Universität Hamburg, Hamburg (2001)
6. Dörner, D.: The Mathematics of Emotion. In The Logic of Cognitive Systems. Fifth International Conference on Cognitive Modelling, 2003-04-10/2003-04-12, Universitätsverlag Bamberg, Bamberg (2003)
7. Dörner, D., Gerdes, J.: Motivation, Emotion, Intelligence. ICSAI, Yantai China (2012)
8. Dörner, D., Güss, C. D.: PSI: A computational architecture of cognition, motivation, and emotion. In: Review of General Psychology, volume 17, number 3, pp. 297–317 (2013)
9. Dominique, J., Fensel, D., Davies, J., González-Cabero, R., Pedrinaci, C.: The service web: a web of billions of services. In: Tselentis, G., Domingue, J., Galis, A., Gavras, A., Hausheer, D., Krco, S., Lotz, V., Zahariadis, T. (eds.). Towards a Future Internet: a European Research Perspective, pp. 203–216, IOS Press, Amsterdam (2009)

10. Ehrlinger, L., Woess, W. Towards a Definition of Knowledge Graphs. SEMANTICS 2016. Posters and Demos Track 2016
11. Gaerdenfors, P.: The Geometry of Meaning: Semantics based on Conceptual Spaces. The MIT Press, Cambridge, Massachusetts, London (2014)
12. Graupner, S., Netzad, H. R. M., Singhal. S.: Making Processes from Best Practice Frameworks Actionable. 2009 13th Enterprise Distributed Object Computing Conference Workshops, (2009)
13. Emotion Analysis: A Survey: Hakak, N. M., Mohd, M. Kirmani, M., Mohd, M.: 2017 International Conference on Computer, Communications and Electronics (Comptelix), Manipal University Jaipur, Malaviya, (2017)
14. He, S., Liu, K., Ji, G., Zhao, J.: Learning to Represent Knowledge Graphs with Gaussian Embedding. In Proceedings of the 24th ACM International Conference on Information and Knowledge Management, CIKM 2015, Melbourne, VIC, Australia, October 19 - 23, 2015, pp. 623-632 (2015)
15. Hudlicka. Alternative Theoretical Perspectives on Emotion Representation and Modeling. https://www.slideshare.net/hudlicka, Last accessed: 2020-01-14
16. LODE: An ontology for Linking Open Descriptions of Events. http://linkedevents.org/ontology/2010-10-07/, Last accessed: 2020-01-14
17. Marsella, S., Gratch, J., Petta, P.: Computational Models of Emotion. In: Scherer, K.R., Bänziger, T., Roesch, E. (eds.) A Blueprint for Affective Computing: A Sourcebook and Manual, OXFORD UNIV PR (2010)
18. Mehrabian, A.: Framework for a comprehensive description and measurement of emotional states. In: Genetic, Social, and General Psychology Monographs, 121, 339-361 (1995)
19. Mehrabian, A.: Pleasure-Arousal-Dominance: A General Framework for Describing and Measuring Individual Differences in Temperament. In: Current Psychology: Developmental Learning Personality Social Winter, volume 14, number 4, pp 261-292 (1996)
20. Ortony, A., Clore, G. L., Collins, A.: The Cognitive Structure of Emotions. NY: Cambridge (1988)
21. RDFLib Homepage, https://rdflib.readthedocs.io/en/stable/. Last accessed 2020-01-14
22. Sander, D., Grandjean, D., Scherer. K. R.: A systems approach to appraisal mechanisms in emotion. In: Neural networks, volume 18, number 4, pp. 317-352, Elsevier (2005)
23. Scherer, K. R.: Appraisal considered as a process of multilevel sequential checking. in Appraisal processes in emotion: Theory, methods, research. volume 92, number 120, pages 57, New York (2001)
24. Scherer, K. R.: The component process model: Architecture for a comprehensive computational model of emergent emotion. In: Scherer K. R., Bänziger, T., Roesch E. B.(eds.), Blueprint for Affective Computing: A Sourcebook, OXFORD, pp. 47-70, OXFORD UNIVERSITY PRESS (2010)