CSSR for

Culturally Sensitive Social Robotics for Africa

D5.4.1 Cultural Knowledge Ontology & Culture Knowledge Base

Due date: **31/12/2024**
Submission Date: **30/12/2024**
Revision Date: **03/03/2025**

Start date of project: **01/07/2023** Duration: **36 months**

Lead organisation for this deliverable: **Carnegie Mellon University Africa (for Wits)**

Responsible Person: **D. Vernon** Revision: **1.3**

Project funded by the African Engineering and Technology Network (Afretec) Inclusive Digital Transformation Research Grant Programme		
Dissemination Level		
PU	Public	**PU**
PP	Restricted to other programme participants (including Afretec Administration)	
RE	Restricted to a group specified by the consortium (including Afretec Administration)	
CO	Confidential, only for members of the consortium (including Afretec Administration)	

Executive Summary

Deliverable D5.4.1 formalizes the Rwandan modes of social interaction documented in Deliverable D1.2: culturally sensitive behaviours, activities, actions, and motions, i.e., Rwandan cultural knowledge for polite and respectful interaction. It presents a cultural knowledge ontology and, based on this ontology, a simple representation of the cultural knowledge documented in D1.2, in the form of the cultural parameter values that can be used by the robot to emulate these polite and respectful behaviours, activities, actions, and motions.

The deliverable comprises three elements: (i) a cultural knowledge ontology, (ii) a related culture knowledge base file, and (iii) the documented software required to compile a C++ helper class to read the culture knowledge base file, store the knowledge, and make the knowledge accessible through a two public access methods.

In the work plan, this deliverable is assigned to the University of the Witswatersrand. However, the material in this version was developed and written by Carnegie Mellon University Africa. This was necessary because the ontology was needed to guide the preparation of the survey in Deliverable D1.2, and because a knowledge base was needed for integration in the system architecture described in Deliverable D3.1.

Contents

1 Introduction

Deliverable D5.4.1 formalizes the Rwandan modes of social interaction documented in Deliverable D1.2: culturally sensitive behaviours, activities, actions, and motions, i.e., Rwandan cultural knowledge for polite and respectful interaction. It presents a cultural knowledge ontology and, based on this ontology, a simple representation of the cultural knowledge documented in D1.2. This takes the form of the cultural key-value pairs that can be used by the robot to emulate polite and respectful behaviours, activities, actions, and motions.

The deliverable comprises three elements: (i) a cultural knowledge ontology, (ii) a related culture knowledge base file, and (iii) the documented software required to compile a C++ helper class to read the culture knowledge base file, store the knowledge, and make the knowledge accessible through two public access methods.

We begin in Section 2 by addressing the representation of cultural knowledge, identifying the different categories of knowledge, and we introduce a simple ontology of cultural knowledge in the form of key-value pairs. This ontology, which is shown in Fig. 2, was used to organize the questions in Part 3 of the survey described in Deliverable D1.2.

In Section 3, we map the Rwandan cultural knowledge to the knowledge ontology. To do this we first analyze the fifty-two consensus answers to the subset of the fifty-seven questions in the cultural knowledge survey documented in Deliverable D1.2, i.e., excluding questions 2-4, 2-5, 2-8, 3-28, and 3-30 for which no consensus emerged. The questions and consensus answers, reproduced from Deliverable D1.2 Rwandan Cultural Knowledge, are shown in Tables 1 – 4. The goal of this analysis is to map each of the fifty-two questions and consensus answers to ontology keys. This necessitated an extension to the original ontology shown in Fig. 2 to ensure that each item of knowledge has a corresponding entry in the ontology tree. This extended ontology is shown in Figs. 3 and 4.

Section 4 then lists each item of knowledge along with its corresponding key, derived from the ontology tree. It does so in two forms. First, we list keys and the corresponding items of knowledge, derived directly from the consensus answers. Second, we list keys and the corresponding items of knowledge in the form of numeric or symbolic values that can be used directly by the robot mission interpreter, i.e., the `behaviorController` ROS node. These key-value pairs are stored in the file that forms the culture knowledge base.

The implementation of software to support access to the culture knowledge base is described in Section 5. As noted above, this takes the form of a C++ helper class, and an example application, implemented in the `utilities` ROS package, to read the culture knowledge base file, store the knowledge, and make the knowledge accessible through two public access methods.

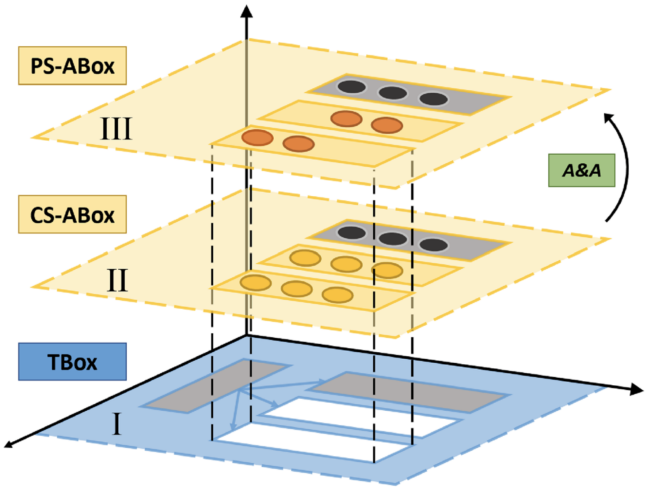

Figure 1: Knowledge representation architecture for a culturally competent robot. The bottom TBox layer (Layer I) defines the ontology for all knowledge, including domain-specific ontologies and upper ontologies that provide interoperability among domain-specific ontologies (grey boxes), and ontologies that model cultural-knowledge (white boxes). The middle CS-ABox layer (Layer II) is the culture-specific layer which includes instances of national-level cultural knowledge (yellow circles), as well as instances of knowledge from domain-specific ontologies and upper ontologies (grey circles). The top PS-ABox layer (Layer III) is the person-specific layer which includes instances of knowledge about the user (orange circles), as well as instances of knowledge from domain-specific ontologies and upper ontologies (grey circles). (From Bruno et al. 2019 [1].)

2 Representation of Cultural Knowledge

In the following, we summarize the knowledge representation architecture and knowledge classification suggested by Bruno et al. [1] and adopt elements of this classification to create a knowledge representation, i.e., a cultural knowledge ontology, that can be used in the CSSR4Africa system.

Section 3 describes the mapping of the answers to each question in the survey to the ontology, and the definition of a simple representation of the knowledge using key-value pairs, with keys derived from the ontology.

Section 4 then lists the knowledge associated with each key, and numeric or symbolic values that encapsulate this knowledge.

2.1 The Different Categories of Knowledge

Bruno et al. [1] propose a knowledge representation architecture for a culturally competent robot; see Fig. 1. This architecture has three layers, each capturing a different element of the knowledge specification. The bottom layer is a *terminological* box (TBox). This is where the ontology proper is specified. The middle and top layers are *assertional* boxes (ABox). This is where the culture-specific and person-specific knowledge (defined by the ontology) is stored.

In more detail, the three elements of the knowledge representation architecture are as follows.

A culture-generic knowledge ontology is captured in the bottom TBox layer (Layer I). This layer defines the ontology for all knowledge, including domain-specific ontologies and upper ontolo-

gies that provide interoperability among domain-specific ontologies (grey boxes), and ontologies that model cultural-knowledge (white boxes).

Culture-specific knowledge is captured in the middle CS-ABox layer (Layer II). Specifically, this layer includes instances of national-level cultural knowledge (yellow circles), as well as instances of knowledge from domain-specific ontologies and upper ontologies.

Person-specific knowledge is captured in the top PS-ABox layer (Layer III), including instances of knowledge about the user (orange circles), as well as instances of knowledge from domain-specific ontologies and upper ontologies

The culture-generic knowledge ontology captures eight types of knowledge, grouped in three categories, as follows.

1. Context knowledge.
   (a) Knowledge about the assisted person.
   (b) Knowledge about the environment.
2. Robot knowledge.
   (a) Knowledge about the actions that the robot can perform.
   (b) Knowledge about the parameters of these actions.
   (c) Knowledge about how actions can be combined into higher-level behaviours.
3. Core values knowledge.
   (a) Knowledge about the goals of the robot mission.
   (b) Knowledge about social norms; these can be considered additional culturally-grounded goals, i.e., constraints on goals, planning operators, action, and cultural parameters.
   (c) Knowledge about conversational subject matter.

Here, we are concerned with 2 (a) knowledge about the actions that the robot can perform, 2 (b) knowledge about the parameters of these actions, and 3 (b) knowledge about social norms. The values we use for the action and cultural parameters determine the culturally sensitive nature of the robot's actions. To quote Bruno et al. [1]:

> "Knowledge pertaining to the robot's sensorimotor and communication capabilities is required by the robot to know what it can do and how the user might prefer it to be done. This knowledge again includes static, *a priori* information (e.g., describing the set of commands allowing the robot to perform the Namaste greeting, the associated parameters and their preferable values) and dynamic information (e.g., describing the robot's current posture and values of related parameters)."

These values are then used by the various ROS nodes in the CSSR4Africa system when invoking actions through ROS service requests. The values themselves are derived from the consensus answers to the survey questions.

The actions that the robot can perform — 2 (a) — depend on the functionality of the system architecture, as described in Deliverable D3.1: animate behaviour, deictic, iconic, and symbolic gesture, overt attention, locomotion and navigation. As such, we do not encode this knowledge explicitly

in the CSSR4Africa knowledge base. Neither do we encode knowledge about how actions can be combined into higher-level behaviours — 2 (c) — explicitly in the CSSR4Africa knowledge base, although some of the knowledge that is revealed and made explicit by the consensus answers to the survey questions does address activities and behaviours. Thus, the cultural knowledge that is required for the CSSR4Africa project has two forms:

1. A compendium of culturally sensitive behaviours, activities, actions, and motions.
2. An knowledge ontology to categorize the behaviours, activities, actions, and motions that the Pepper robot can perform.
3. A mapping from the compendium of culturally sensitive behaviours, activities, actions, and motions to the ontology.
4. The action and cultural parameter values — 2 (b) and 3 (b) — that are derived from a subset of the consensus answers.

Item 1 comprises the consensus answers to the fifty-seven questions in the Rwandan cultural knowledge survey documented in Deliverable D1.2. The remaining items are documented in this deliverable. Note that the mapping from the compendium of culturally sensitive behaviours, activities, actions, and motions to the ontology is partial because there are some behaviours, activities, actions, and motions that the Pepper robot is incapable of performing.

2.2 A Knowledge Ontology for the Pepper Robot

As noted in Section 1, Deliverable D1.2 compiles the cultural knowledge required for culturally sensitive human robot interaction between robots and Rwandan people. To be effective, this knowledge must be organized in some manner. This organization is effectively a knowledge ontology that resulted from work carried out early in Task 5.4.1, and documented in this deliverable in Figure 2. This ontology was used to guide the preparation of the survey in Deliverable D1.2. As we will see in Section 3, this ontology had to be extended following an analysis of the consensus answers to the questions in the survey.

While Bruno et al. [1] use the OWL-2 language to define their ontology, we adopt a simpler approach here that represents the ontology as a tree of concepts, as shown in Figure 2. Note that the ontology is restricted to the actions that the Pepper robot can perform. It explicitly omits forms of non-verbal communication that are important in human-robot interaction, e.g., facial expressions, such as eyebrow and mouth gestures. This provides us with a straightforward way to specify the parameter values for each element in the ontology: we can represent the cultural knowledge with a simple list of key-value pairs, where a key is constructed from the name of a leaf nodes in the ontology tree and the name of its parent. The values can be either quantitative numeric values or qualitative symbolic values, which can then be interpreted by the ROS node that uses the key-value pair to produce culturally sensistive behaviour.

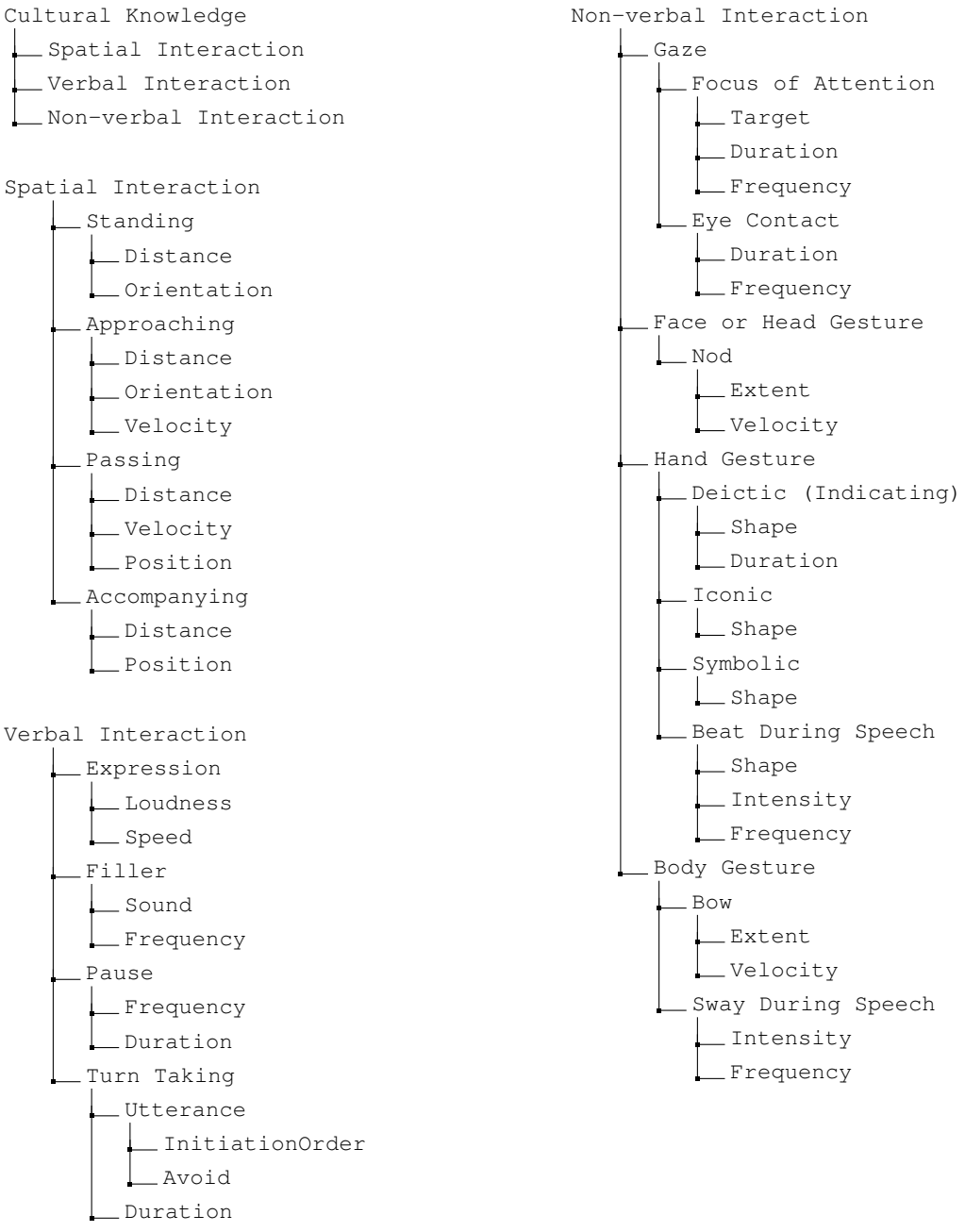

Figure 2: Original ontology of cultural knowledge. Note that the ontology is restricted to the actions that the Pepper robot can perform.

3 Mapping Rwandan Cultural Knowledge to the Knowledge Ontology Keys

By itself, an ontology is insufficient, since the knowledge base that is to be used by the robot must be populated by the knowledge that is derived from the survey and documented in Deliverable D1.2. Thus, we need to map the knowledge encapsulated in the consensus answers to each question in the survey to the ontology. For ease of reference, the survey questions and associated consensus answers in Deliverable D1.2 are reproduced here in Tables 1 – 4. Five of the fifty-seven questions in the cultural knowledge survey yielded no consensus: 2-4, 2-5, 2-8, 3-28, and 3-30, leaving fifty-two consensus answers to populate the culture knowledge base. This necessitated that the original ontology shown in Figure 2 be extended to ensure that each item of knowledge has a corresponding entry in the ontology tree. Specifically, if several survey answers map to a single leaf node in the ontology tree in Fig. 2, the ontology was extended by appending suffixes to the leaf labels, e.g., `SymbolicShapeRespect` instead of `SymbolicShape`. This results in an ontology tree with one leaf node for each consensus answer. The onology tree also has several leaf nodes for which there is no associated knowledge, either because no consensus answer emerged, or because no survey question addressed that element of the ontology. The extended ontology is shown in Figures 3 and 4.

Table 5 lists the keys derived from the original ontology tree in Figure 2, along with the questions in Parts 2 and 3 of the survey that reveal the associated cultural knowledge listed in Tables 3 and 4. Tables 6 - 8 list the keys derived from the extended ontology tree in Figures 3 and 4, along with each corresponding question in Parts 2 and 3 of the survey.

2-1	To show respect, one should lower gaze when greeting someone older.
2-2	One should suspend work or movements and pay attention when addressed.
2-3	One should keep intermittent eye contact; lack of eye contact depicts disrespect as it shows divided attention during the interaction.
2-4	One should not make persistent eye contact with an older person.
2-5	One should not make eye contact when being corrected by someone.
2-6	One should use an open palm of the hand to point to people and objects.
2-7	One should not point an upward facing palm of the hand at someone.
2-8	One should not use the left hand to point to anything.
2-9	To show respect, one should bow slightly when greeting someone older.
2-10	To show respect, one should raise both hands when greeting.
2-11	One should not wave at someone from a distance; one should move towards them to greet them.
2-12	One should not use the left hand to hand something to someone.
2-13	To show respect, one should hand over and accept gifts with two hands and do so from the front, facing the recipient.
2-14	To show respect, one should shake hands with the right hand and use the left arm to support the right forearm when doing so.
2-15	An appreciation of rhythmic sound and movement is valued.
2-16	To show respect, one should bow slightly and lower gaze when greeting someone older.
2-17	The younger interaction partner should bow when greeting an older person or when rendering a service.
2-18	All interactions should begin with a courteous greeting.
2-19	The younger interaction partner should enable a greeting to be initiated by an older person.
2-20	It is respectful to use local languages and they should be used for verbal interaction when possible.
2-21	One should use formal titles when addressing someone.
2-22	One should engage in a preamble before getting to the point, as being too forward may be regarded as disrespectful.
2-23	One should not interrupt or talk over someone when they are speaking.
2-24	One should not talk loudly to an older person.
2-25	Behaviours should focus on fostering social connections and relationships; they should not be purely functional.
2-26	One should not walk between two or more people who are conversing because it is considered rude to do so.
2-27	One should not walk far ahead of an older person, unless leading the person (in which case, one should walk slightly to the side).

Table 1: Survey questions – Part 2. Respondents who took the survey were asked whether these statements were correct or incorrect, or whether they were unsure.

3-1	What distance should you keep when passing someone?
3-2	How should you acknowledge someone when passing them?
3-3	How should you pass a group of two or more people?
3-4	When showing someone older than you the way, where should you position yourself?
3-5	When showing someone the same age as you the way, where should you position yourself?
3-6	When showing someone younger than you the way, where should you position yourself?
3-7	How should you address someone who is older than you and who you haven't met before?
3-8	How should you address someone who is the same age as you and who you haven't met before?
3-9	How should you address someone who is younger than you and who you haven't met before?
3-10	Should you pause before responding when someone asks you a question? If yes, for how long?
3-11	In an interaction where you and someone else take turns to speak, would you signal that you want to speak? If yes, how do you do that?
3-12	If you are explaining something to someone, what is your primary focus of attention, i.e., where do you direct your gaze?
3-13	If you are explaining something to someone, how often should you make eye contact?
3-14	If you are explaining something to someone, how often would you make eye contact if the person was older than you?
3-15	If you are explaining something to someone, how often would you make eye contact if the person was younger than you?
3-16	If someone is explaining something to you, what is your primary focus of attention, i.e., where do you direct your gaze?
3-17	If someone is explaining something to you, how often should you make eye contact?
3-18	If someone is explaining something to you, how often would you make eye contact if the person was older than you?
3-19	If someone is explaining something to you, how often would you make eye contact if the person was younger than you?
3-20	Would you use a face or head gesture to draw someone's attention to something? If yes, what would that gesture be?
3-21	Would you use a face, head, hand, or body gesture to express gratitude?
3-22	Would you use a face, head, hand, or body gesture to express agreement?
3-23	Would you use a face, head, hand, or body gesture to express respect?
3-24	Would you use a face, head, hand, or body gesture to express friendliness?
3-25	Would you use a face, head, hand, or body gesture to express confusion?
3-26	Would you use a face, head, hand, or body gesture to express comprehension?
3-27	Would you use a face, head, hand, or body gesture to express interest?
3-28	Is there a face head, hand, or body gesture you should not use?
3-29	Would you use a hand or body gesture while speaking to someone?
3-30	Would you use a hand or body gesture while listening to someone?

Table 2: Survey questions – Part 3. Respondents who took the survey were given a number of possible answers from which to choose.

Question	Consensus Cultural Knowledge
2-1	To show respect, one should lower gaze when greeting someone older.
2-2	One should suspend work or movements and pay attention when addressed.
2-3	One should keep intermittent eye contact; lack of eye contact depicts disrespect as it shows divided attention during the interaction
2-4	No consensus
2-5	No consensus
2-6	One should use an open palm of the hand to point to people and objects.
2-7	One should not point an upward facing palm of the hand at someone.
2-8	No consensus
2-9	To show respect, one should bow slightly when greeting someone older.
2-10	To show respect, one should raise both hands when greeting.
2-11	One should not wave at someone from a distance; one should move towards them to greet them
2-12	One should not use the left hand to hand something to someone
2-13	To show respect, one should hand over and accept gifts with two hands and do so from the front, facing the recipient
2-14	To show respect, one should shake hands with the right hand and use the left arm to support the right forearm when doing so.
2-15	An appreciation of rhythmic sound and movement is valued.
2-16	To show respect, one should bow slightly and lower gaze when greeting someone older
2-17	The younger interaction partner should bow when greeting an older person or when rendering a service
2-18	All interactions should begin with a courteous greeting.
2-19	The younger interaction partner should enable a greeting to be initiated by an older person.
2-20	It is respectful to use local languages and they should be used for verbal interaction when possible.
2-21	One should use formal titles when addressing someone.
2-22	One should engage in a preamble before getting to the point, as being too forward may be regarded as disrespectful.
2-23	One should not interrupt or talk over someone when they are speaking.
2-24	One should not talk loudly to an older person
2-25	Behaviours should focus on fostering social connections and relationships; they should not be purely functional.
2-26	One should not walk between two or more people who are conversing because it is considered rude to do so.
2-27	One should not walk far ahead of an older person, unless leading the person (in which case, one should walk slightly to the side).

Table 3: Consensus answers to the twenty-seven questions in Part 2 of the cultural knowledge survey.

Question	Consensus Cultural Knowledge
3-1	One should maintain a distance of one meter or less when passing someone.
3-2	One should say 'Hello' or 'Muraho' when acknowledging someone while passing them.
3-3	One should pass behind a group of two or more people.
3-4	One should position themselves beside someone older when showing them the way.
3-5	One should position themselves beside someone of the same age when showing them the way.
3-6	One should position themselves beside someone younger when showing them the way.
3-7	The preferred way to address someone older, and whom you haven't met before, is by saying 'Muraho' or 'Hello'.
3-8	The preferred way to address someone the same age, and whom you haven't met before, is by saying 'Muraho' or 'Hello'.
3-9	The preferred way to address someone younger, and whom you haven't met before, is by saying 'Muraho' or 'Hello'.
3-10	When asked a question, respondents should pause for a few seconds before answering.
3-11	In turn-based conversations, participants can raise their right hand to signal their desire to speak.
3-12	When explaining something to someone, you should direct your gaze equally between. the person and the object.
3-13	When explaining something to someone, you should make eye contact often.
3-14	You should make eye contact often when explaining something to someone older than you.
3-15	You should make eye contact often when explaining something to someone younger than you.
3-16	When someone is explaining something to you, you should direct your gaze equally between the person and the object.
3-17	When someone is explaining something to you, you should make eye contact often.
3-18	If someone is explaining something to you and they are older than you, you should make eye contact often.
3-19	If someone is explaining something to you and they are younger than you, you should make eye contact often.
3-20	To draw someone's attention to something, use a head-nodding gesture while looking at the object.
3-21	To express gratitude, common gestures include nodding, smiling, and bowing the head, using hand gestures like a thumbs up or clasped hands, and slight bowing of the body.
3-22	To express agreement, common gestures include nodding the head and giving a thumbs up with the right hand.
3-23	To show respect, common gestures include a slight bow of the head, a greeting or handshake using the right hand supported by the left, and bowing, which is the most frequent body gesture.
3-24	To express friendliness, people commonly use facial gestures like smiling, hand gestures such as a handshake using both hands or the right hand, and body gestures such as hugging.
3-25	When expressing confusion, individuals typically use facial gestures like wrinkling or frowning the brow or tilting the head, hand gestures such as raising both hands or the right hand, tand body movements that vary according to the situation.
3-26	When expressing comprehension, individuals typically use head gestures, such as nodding, hand gestures like a right-hand thumbs-up, and body gestures that vary by situation.
3-27	When expressing interest, nodding and smiling are the most common gestures. Hand gestures, such as giving a thumbs up with the right hand, and body gestures such as facing someone, are used less frequently.
3-28	No consensus.
3-29	One should use various body and hand gestures while speaking to someone. The most recommended gestures are slight body movement and slightly moving both hands.
3-30	No consensus.

Table 4: Consensus answers to the thirty questions in Part 3 of the cultural knowledge survey.

Key	Questions
Spatial Interaction	
StandingDistance	
StandingOrientation	
ApproachingDistance	
ApproachingOrientation	
ApproachingVelocity	
PassingDistance	3-1
PassingVelocity	
PassingPosition	2-26, 3-3
AccompanyingDistance	2-27, 3-4, 3-5, 3-6
AccompanyingPosition	
Verbal Interaction	
ExpressionLoudness	2-24
ExpressionSpeed	
FillerSound	
FillerFrequency	
PauseFrequency	
PauseDuration	3-10
TurnTakingUtterance	2-19, 2-23
TurnTakingDuration	
Non-Verbal Interraction	
Gaze	
FocusofAttentionTarget	2-1, 2-2, 2-16, 3-12, 3-16
FocusofAttentionDuration	
FocusofAttentionFrequency	
EyeContactDuration	2-3, 2-4, 2-5, 3-2
EyeContactFrequency	3-13, 3-14, 3-15, 3-17, 3-18, 3-19
Face or Head Gesture	
NodExtent	3-20, 3-21, 3-22, 3-23, 3-24, 3-26, 3-27
NodVelocity	
Hand Gesture	
DeicticShape	2-6, 2-7, 2-8
DeicticDuration	
IconicShape	3-29, 3-30
SymbolicShape	2-10, 2-13, 2-14, 3-11, 3-21, 3-22, 3-23, 3-24, 3-25, 3-26, 3-27
BeatShape	
BeatIntensity	
BeatFrequency	
Body Gesture	
BowExtent	2-9, 2-16, 2-17, 3-21, 3-22, 3-23, 3-24, 3-26, 3-27
BowVelocity	
SwayIntensity	3-21, 3-22, 3-23, 3-24, 3-25, 3-26, 3-27
SwayFrequency	

Table 5: Keys for specifying culturally sensitive actions and the corresponding questions in Parts 2 and 3 of the survey that reveal the associated cultural knowledge, prior to extending the ontology, i.e., using the ontology depicted in Fig. 2.

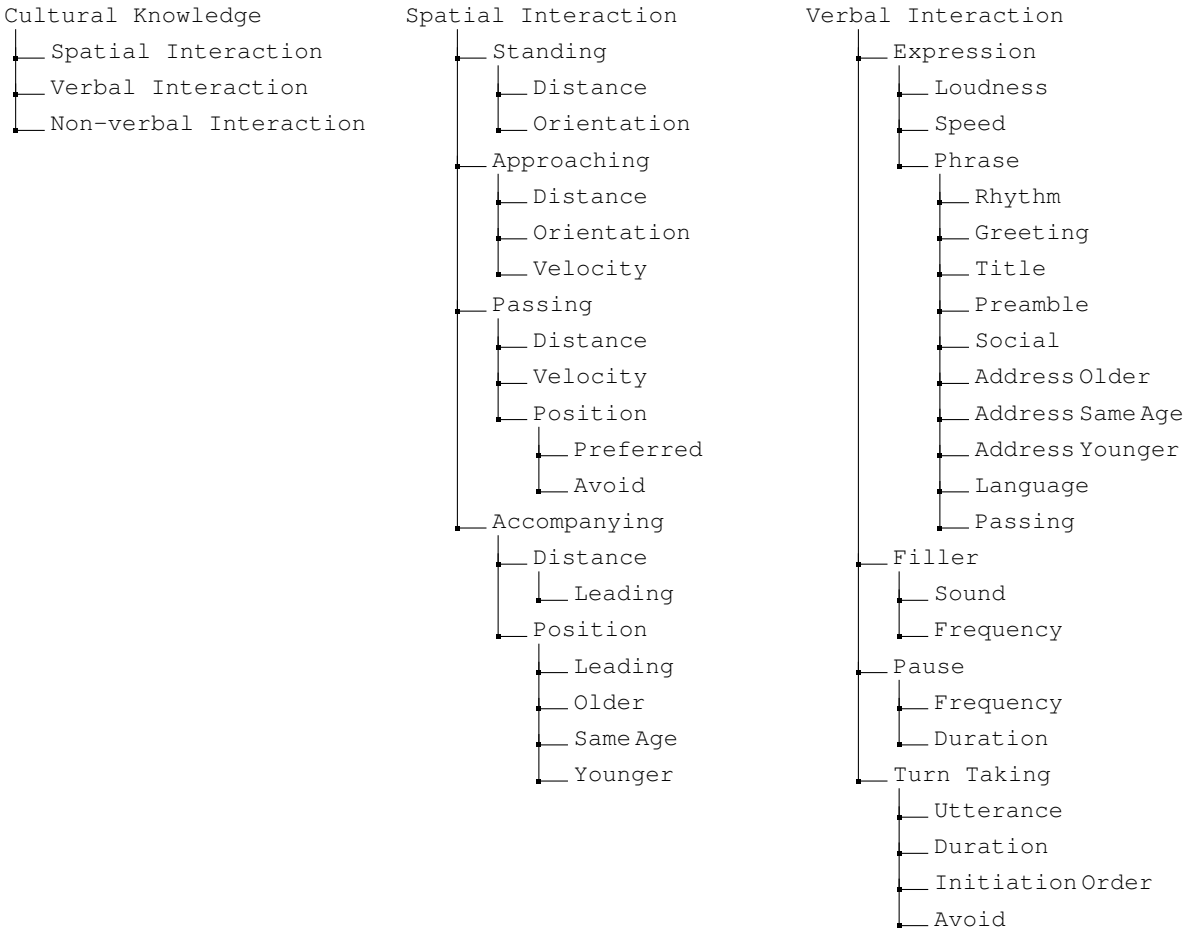

Figure 3: Extended ontology of cultural knowledge after mapping Rwandan cultural knowledge to the knowledge ontology.

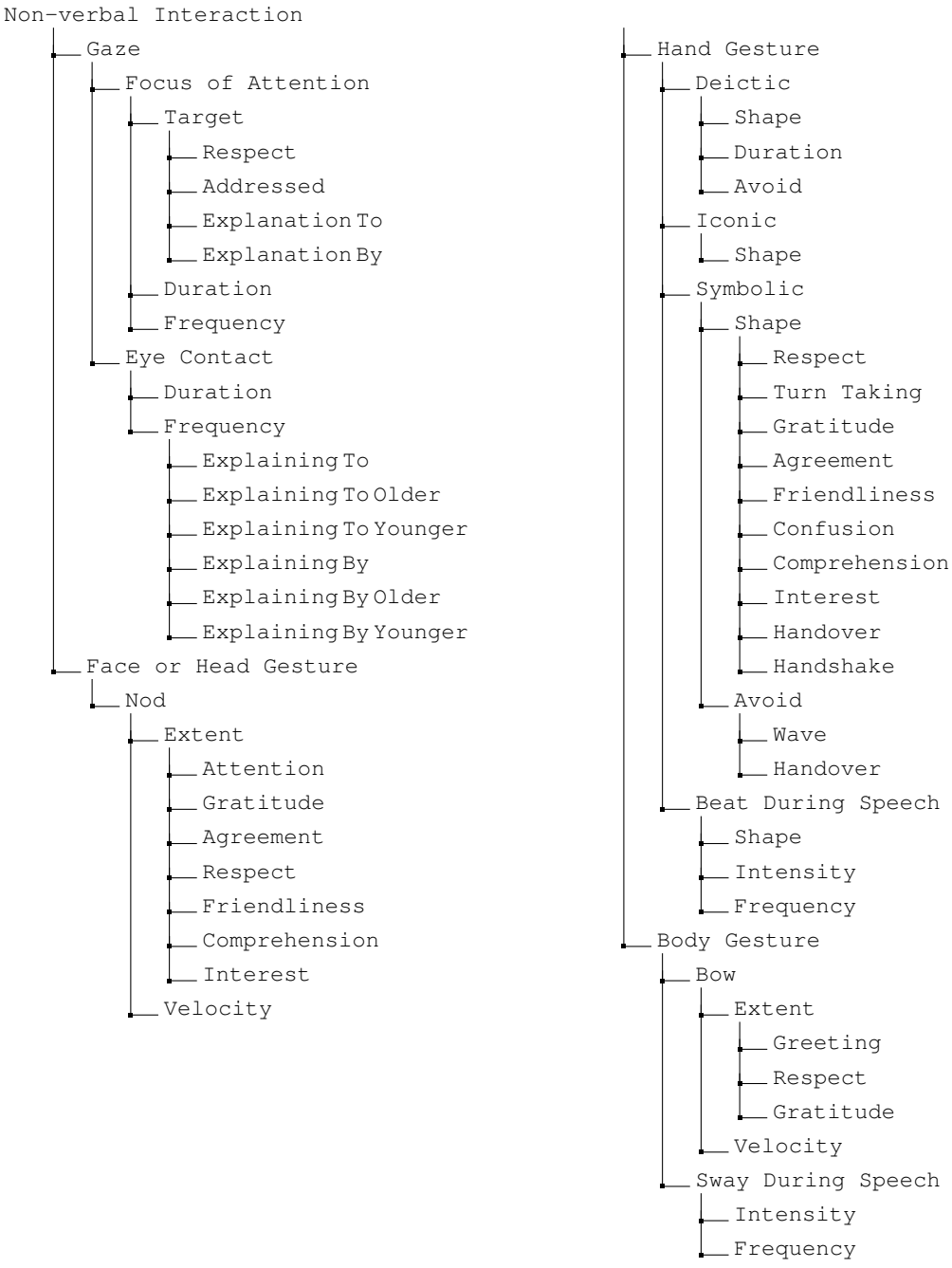

Figure 4: Extended ontology of cultural knowledge after mapping Rwandan cultural knowledge to the knowledge ontology.

Spatial Interaction	
Key	Questions
StandingDistance	
StandingOrientation	
ApproachingDistance	
ApproachingOrientation	
ApproachingVelocity	
PassingDistance	3-1
PassingVelocity	
PassingPositionPreferred	2-26
PassingPositionAvoid	3-3
AccompanyingDistanceLeading	2-27
AccompanyingPositionLeading	2-27
AccompanyingPositionOlder	3-4
AccompanyingPositionSameAge	3-5
AccompanyingPositionYounger	3-6

Table 6: Keys for specifying culturally sensitive spatial interaction and the corresponding questions in Parts 2 and 3 of the survey that reveal the associated cultural knowledge, after extending the ontology, i.e., using the ontology depicted in Figs. 3 and 4.

Verbal Interaction	
Key	Questions
ExpressionLoudness	2-24
ExpressionSpeed	
ExpressionPhraseRhythm	2-15
ExpressionPhraseGreeting	2-18
ExpressionPhraseTitle	2-21
ExpressionPhrasePreamble	2-22
ExpressionPhraseSocial	2-25
ExpressionPhraseAddressOlder	3-7
ExpressionPhraseAddressSameAge	3-8
ExpressionPhraseAddressYounger	3-9
ExpressionPhraseLanguage	2-20
ExpressionPhrasePassing	3-2
FillerSound	
FillerFrequency	
PauseFrequency	
PauseDuration	3-10
TurnTakingUtterance	
TurnTakingDuration	
TurnTakingInitiationOrder	2-19
TurnTakingAvoid	2-23

Table 7: Keys for specifying culturally sensitive verbal interaction and the corresponding questions in Parts 2 and 3 of the survey that reveal the associated cultural knowledge, after extending the ontology, i.e., using the ontology depicted in Figs. 3 and 4.

Non-verbal Interaction	
Key	Questions
Gaze	
FocusofAttentionTargetRespect	2-1, 2-16
FocusofAttentionTargetAddressed	2-2
FocusofAttentionTargetExplanationTo	3-12
FocusofAttentionTargetExplanationBy	3-16
FocusofAttentionDuration	
FocusofAttentionFrequency	
EyeContactDuration	2-3
EyeContactFrequencyExplainingTo	3-13
EyeContactFrequencyExplainingToOlder	3-14
EyeContactFrequencyExplainingToYounger	3-15
EyeContactFrequencyExplainingBy	3-17
EyeContactFrequencyExplainingByOlder	3-18
EyeContactFrequencyExplainingByYounger	3-19
Face or Head Gesture	
NodExtentAttention	3-20
NodExtentGratitude	3-21
NodExtentAgreement	3-22
NodExtentRespect	3-23
NodExtentFriendliness	3-24
NodExtentComprehension	3-26
NodExtentInterest	3-27
NodVelocity	
Hand Gesture	
DeicticShape	2-6
DeicticDuration	
DeicticAvoid	2-7
IconicShape	3-29
SymbolicShapeRespect	2-10, 3-23
SymbolicShapeTurnTaking	3-11
SymbolicShapeGratitude	3-21
SymbolicShapeAgreement	3-22
SymbolicShapeFriendliness	3-24
SymbolicShapeConfusion	3-25
SymbolicShapeComprehension	3-26
SymbolicShapeInterest	3-27
SymbolicShapeHandover	2-13
SymbolicShapeHandshake	2-14
SymbolicAvoidWave	2-11
SymbolicAvoidHandover	2-12
BeatShape	
BeatIntensity	
BeatFrequency	
Body Gesture	
BowExtentGreeting	2-9
BowExtentRespect	2-16, 2-17, 3-23
BowExtentGratitude	3-21
BowVelocity	
SwayIntensity	
SwayFrequency	3-29

Table 8: Keys for specifying culturally sensitive non-verbal interaction and the corresponding questions in Parts 2 and 3 of the survey that reveal the associated cultural knowledge, after extending the ontology, i.e., using the ontology depicted in Figs. 3 and 4.

4 Culture Knowledge Base

Tables 9–11 lists the knowledge associated with each key in the extended ontology tree: spatial interaction, verbal interaction, and non-verbal interaction. Tables 12 –14 list the key-value pairs, i.e., with numeric or symbolic values that encapsulate the cultural knowledge. These numeric or symbolic values can then be used directly in the robot mission interpreter, i.e., the `behaviorController` ROS node, and passed as arguments in the service requests it issues to the nodes in the system architecture to achieve culturally sensitive behavior.

The key-value pairs are stored in a file `cultureKnowledgeBaseInput.dat`. This file is read and the value-pairs are accessed using a helper class `CultureKnowledgeBase` described in Section 5.

Spatial Interaction	
Key	Values
`StandingDistance`	
`StandingOrientation`	
`ApproachingDistance`	
`ApproachingOrientation`	
`ApproachingVelocity`	
`PassingDistance`	1m or less
`PassingVelocity`	
`PassingPositionPreferred`	Pass behind a group of two or more people
`PassingPositionAvoid`	Walk between two or more people who are conversing
`AccompanyingDistanceLeading`	Do not walk far ahead of an older person
`AccompanyingPositionLeading`	Walk slightly ahead and to the side
`AccompanyingPositionOlder`	Walk to the side
`AccompanyingPositionSameAge`	Walk to the side
`AccompanyingPositionYounger`	Walk to the side

Table 9: Spatial interaction: key-knowledge pairs for specifying culturally sensitive actions after extending the ontology, i.e., using the ontology depicted in Figs. 3 and 4.

Verbal Interaction	
Key	Values
ExpressionLoudness	Do not talk loudly to an older person
ExpressionSpeed	
ExpressionPhraseRhythm	Say 'Nice'
ExpressionPhraseGreeting	Say 'Muraho' or 'Greetings'
ExpressionPhraseTitle	Say 'Mr', 'Ms', 'Dr', 'Prof', 'Your Excellency'
ExpressionPhrasePreamble	Say 'It's nice to meet you. How are you?'
ExpressionPhraseSocial	Say 'I hope you are well'
ExpressionPhraseAddressOlder	Say 'Muraho' or 'Hello'
ExpressionPhraseAddressSameAge	Say 'Muraho' or 'Hello'
ExpressionPhraseAddressYounger	Say 'Muraho' or 'Hello'
ExpressionPhraseLanguage	Kinyarwanda, isiZulu
ExpressionPhrasePassing	Say 'Muraho' or 'Hello' to acknowledge someone when passing them
FillerSound	
FillerFrequency	
PauseFrequency	
PauseDuration	A few seconds
TurnTakingUtterance	
TurnTakingDuration	
TurnTakingInitiationOrder	Pause until the older person speaks
TurnTakingAvoid	Do not talk over someone when they are speaking

Table 10: Verbal interaction: key-knowledge pairs for specifying culturally sensitive actions after extending the ontology, i.e., using the ontology depicted in Figs. 3 and 4.

Non-Verbal Interaction	
Key	Values
	Gaze
FocusofAttentionTargetRespect	Lower your gaze and bow slightly
FocusofAttentionTargetAddressed	When addressed, stop what you are doing and give your full attention
FocusofAttentionTargetExplanationTo	Direct your gaze equally between the person and the object
FocusofAttentionTargetExplanationBy	Direct your gaze equally between the person and the object
FocusofAttentionDuration	
FocusofAttentionFrequency	
EyeContactDuration	Make intermittent eye contact
EyeContactFrequencyExplainingTo	Make eye contact often
EyeContactFrequencyExplainingToOlder	Make eye contact often
EyeContactFrequencyExplainingToYounger	Make eye contact often
EyeContactFrequencyExplainingBy	Make eye contact often
EyeContactFrequencyExplainingByOlder	Make eye contact often
EyeContactFrequencyExplainingByYounger	Make eye contact often
	Face or Head Gesture
NodExtentAttention	Use a head-nodding gesture while looking at the object
NodExtentGratitude	Use a head-nodding gesture while looking at the person
NodExtentAgreement	Use a head-nodding gesture while looking at the person
NodExtentRespect	Use a head-nodding gesture while looking at the person
NodExtentFriendliness	Use a head-nodding gesture while looking at the person
NodExtentComprehension	Use a head-nodding gesture while looking at the person
NodExtentInterest	Use a head-nodding gesture while looking at the person
NodVelocity	
	Hand Gesture
DeicticShape	Use an open palm to point at people and objects
DeicticDuration	
DeicticAvoid	Do not point an upward-facing palm at someone
IconicShape	Make slight body and hand movements
SymbolicShapeRespect	Raise both hands when greeting
SymbolicShapeTurnTaking	Raise the right hand to signal desire to speak
SymbolicShapeGratitude	Give thumbs-up with the right hand or clasp hands
SymbolicShapeAgreement	Give thumbs-up with the right hand
SymbolicShapeFriendliness	Handshake with both hands or just the right hand
SymbolicShapeConfusion	Raise both hands or just the right hand
SymbolicShapeComprehension	Give thumbs-up with the right hand
SymbolicShapeInterest	Give thumbs-up with the right hand
SymbolicShapeHandover	Use both hands
SymbolicShapeHandShake	Handshake with the right hand, supported by the left, while bowing
SymbolicAvoidWave	Do not wave at someone from a distance
SymbolicAvoidHandover	Do no use the left hand without using the right hand
BeatShape	
BeatIntensity	
BeatFrequency	
	Body Gesture
BowExtentGreeting	Bow slightly when greeting someone older.
BowExtentRespect	Bow slightly
BowExtentGratitude	Bow slightly
BowVelocity	
SwayIntensity	Make slight body movements
SwayFrequency	

Table 11: Non-verbal interaction: key-knowledge pairs for specifying culturally sensitive actions after extending the ontology, i.e., using the ontology depicted in Figs. 3 and 4.

Spatial Interaction		
Key	Values	Units
StandingDistance		
StandingOrientation		
ApproachingDistance		
ApproachingOrientation		
ApproachingVelocity		
PassingDistance	100	centimetres
PassingVelocity		
PassingPositionPreferred	BEHIND	
PassingPositionAvoid	BETWEEN	
AccompanyingDistanceLeading	100	centimetres
AccompanyingPositionLeading	FRONTSIDE	
AccompanyingPositionOlder	SIDE	
AccompanyingPositionSameAge	SIDE	
AccompanyingPositionYounger	SIDE	

Table 12: Spatial interaction: key-value pairs for specifying culturally sensitive actions after extending the ontology, i.e., using the ontology depicted in Figs. 3 and 4.

Verbal Interaction		
Key	Values	Units
ExpressionLoudness	40	% of max. volume
ExpressionSpeed		
ExpressionPhraseRhythm	Nice	
ExpressionPhraseGreeting	Muraho | Greetings	
ExpressionPhraseTitle	Mr | Ms | Dr | Prof | Excellency	
ExpressionPhrasePreamble	It is nice to meet you. How are you?	
ExpressionPhraseSocial	I hope you are well	
ExpressionPhraseAddressOlder	Muraho | Hello	
ExpressionPhraseAddressSameAge	Muraho | Hello	
ExpressionPhraseAddressYounger	Muraho | Hello	
ExpressionPhraseLanguage	Kinyarwanda | isiZulu | English	
ExpressionPhrasePassing	Muraho | Hello	
FillerSound		
FillerFrequency		
PauseFrequency		
PauseDuration	2	seconds
TurnTakingUtterance		
TurnTakingDuration		
TurnTakingInitiationOrder	WAIT	
TurnTakingAvoid	INTERRUPT	

Table 13: Verbal interaction: key-value pairs for specifying culturally sensitive actions after extending the ontology, i.e., using the ontology depicted in Figs. 3 and 4.

Non-Verbal Interaction		
Key	Values	Units
Gaze		
FocusofAttentionTargetRespect	TORSO	
FocusofAttentionTargetAddressed	FACE	
FocusofAttentionTargetExplanationTo	FACE_AND_OBJECT	
FocusofAttentionTargetExplanationBy	FACE_AND_OBJECT	
FocusofAttentionDuration		
FocusofAttentionFrequency		
EyeContactDuration	2	seconds
EyeContactFrequencyExplainingTo	5	seconds
EyeContactFrequencyExplainingToOlder	5	seconds
EyeContactFrequencyExplainingToYounger	5	seconds
EyeContactFrequencyExplainingBy	5	seconds
EyeContactFrequencyExplainingByOlder	5	seconds
EyeContactFrequencyExplainingByYounger	5	seconds
Face or Head Gesture		
NodExtentAttention	30	degrees
NodExtentGratitude	30	degrees
NodExtentAgreement	30	degrees
NodExtentRespect	30	degrees
NodExtentFriendliness	30	degrees
NodExtentComprehension	30	degrees
NodExtentInterest	30	degrees
NodVelocity		
Hand Gesture		
DeicticShape	PALM_UPWARDS	
DeicticDuration		
DeicticAvoid	PALM_FORWARDS	
IconicShape	ANIMATE_BEHAVIOR	
SymbolicShapeRespect	BOTH_HANDS_UP	
SymbolicShapeTurnTaking	RIGHT_HAND_UP	
SymbolicShapeGratitude	CLASP_HANDS	
SymbolicShapeAgreement	RIGHT_THUMB_UP	
SymbolicShapeFriendliness	HANDSHAKE	
SymbolicShapeConfusion	RIGHT_HAND_UP	
SymbolicShapeComprehension	RIGHT_THUMB_UP	
SymbolicShapeInterest	RIGHT_THUMB_UP	
SymbolicShapeHandover	BOTH_HANDS	
SymbolicShapeHandShake	POLITE_HANDSHAKE	
SymbolicAvoidWave	30	centimetres
SymbolicAvoidHandover	LEFT_HAND	
BeatShape		
BeatIntensity		
BeatFrequency		
Body Gesture		
BowExtentGreeting	10	degrees
BowExtentRespect	10	degrees
BowExtentGratitude	10	degrees
BowVelocity		
SwayIntensity	ANIMATE_BEHAVIOR	
SwayFrequency		

Table 14: Non-verbal interaction: key-value pairs for specifying culturally sensitive actions after extending the ontology, i.e., using the ontology depicted in Figs. 3 and 4.

5 Implementation

The key-value pairs listed in Tables 12 –14, comprising an alphanumeric key and associated numeric or symbolic values that encapsulate the cultural knowledge, are stored in a file named `cultureKnowledgeBaseInput.dat`. This file is accessed using a C++ helper class `CultureKnowledgeBase` described in this section. Specifically, a C++ object instantiation of the helper class reads the culture knowledge base file, store the knowledge, and make the knowledge accessible through two public access methods. The remainder of this section details the implementation of this C++ helper class.

5.1 File Organization

The files for the culture knowledge base helper class are located in the `utilities` subdirectory, as shown in Figure 5. The constituent files are organized is several subdirectores as shown in Figure 6. There are three C++ source code files: `cultureKnowledgeBaseApplication.cpp`, `cultureKnowledgeBaseImplementation.cpp`, and `cultureKnowledge.h`. The implementation file contains the helper class definition. The interface file contains the helper class declaration. The application file is essentially a unit test to illustrate how the helper class is used and to verify that it works correctly. It instantiates a C++ helper class object which reads the culture knowledge base file, and uses the access method to retrieve values in the culture knowledge base, implemented using a binary search tree dictionary data structure, write them to the terminal.

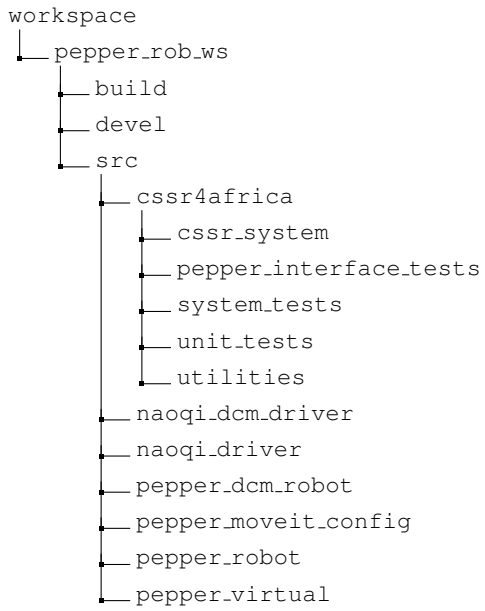

Figure 5: Directory structure for the CSSR4Africa software repository.

```
utilities
├── config
│   └── cultureKnowledgeBaseConfiguration.ini
├── data
│   ├── cultureKnowledgeValueTypes.dat
│   └── cultureKnowledgeBaseInput.dat
├── include
│   └── utilities
│       └── cultureKnowledgeBaseInterface.h
├── src
│   ├── cultureKnowledgeBaseApplication.cpp
│   └── cultureKnowledgeBaseImplementation.cpp
├── launch
│   └── cultureKnowledgeBaseExample.launch
├── README.md
└── CMakeLists.txt
```

Figure 6: Directory Structure for the `CultureKnowledgeBase` C++ helper class.

5.2 Configuration File

The population of the knowledge base is determined by the contents of a configuration file `cultureKnowledgeBase.ini` that contain a list of key-value pairs, as shown below in Table 15.

The configuration file is named `cultureKnowledgeBaseConfiguration.ini`.

Table 15: Configuration file for the `CultureKnowledgeBase` helper class.

Key	Value	Description
`valueTypes`	`cultureKnowledgeValueTypesInput.dat`	Specifies the filename of the file in which the type of the values in each cultural knowledge key-value pair is specified.
`knowledgeBase`	`cultureKnowledgeBaseInput.dat`	Specifies the filename of the file in which the cultural knowledge key-value pairs are stored.
`verboseMode`	`true` or `false`	Specifies whether diagnostic data is to be printed to the terminal.

5.3 Culture Knowledge Value Types

The types of each value are listed in Figure 16. There are four types, one to represent an integer value, one to represent a single word, and one to represent a multi-word string. Key-value pairs without values in Tables 12 -14 have been assigned a fourth value type of `UNDEFINED`. The types are represented with a numeric value, as follows.

```
#define UNDEFINED  0
#define NUMBER     1  // integer type
#define WORD       2  // single word
#define PHRASE     3  // multi-word phrase
```

Table 16: Value types listed in the file `cultureKnowledgeValueTypesInput.dat`. Key-value pairs without values in Tables 12 -14 have type `UNDEFINED`.

StandingDistance	UNDEFINED
StandingOrientation	UNDEFINED
ApproachingDistance	UNDEFINED
ApproachingOrientation	UNDEFINED
ApproachingVelocity	UNDEFINED
PassingDistance	NUMBER
PassingVelocity	UNDEFINED
PassingPositionPreferred	WORD
PassingPositionAvoid	WORD
AccompanyingDistanceLeading	NUMBER
AccompanyingPositionLeading	WORD
AccompanyingPositionOlder	WORD
AccompanyingPositionSameAge	WORD
AccompanyingPositionYounger	WORD
ExpressionLoudness	NUMBER
ExpressionSpeed	UNDEFINED
ExpressionPhraseRhythm	WORD
ExpressionPhraseGreeting	WORD
ExpressionPhraseTitle	WORD
ExpressionPhrasePreamble	PHRASE
ExpressionPhraseSocial	PHRASE
ExpressionPhraseAddressOlder	WORD
ExpressionPhraseAddressSameAge	WORD
ExpressionPhraseAddressYounger	WORD
ExpressionPhraseLanguage	WORD
ExpressionPhrasePassing	WORD
FillerSound	UNDEFINED
FillerFrequency	UNDEFINED
PauseFrequency	UNDEFINED
PauseDuration	NUMBER
TurnTakingUtterance	UNDEFINED
TurnTakingDuration	UNDEFINED
TurnTakingInitiationOrder	WORD
TurnTakingAvoid	WORD
FocusofAttentionTargetRespect	WORD
FocusofAttentionTargetAddressed	WORD
FocusofAttentionTargetExplanationTo	WORD
FocusofAttentionTargetExplanationBy	WORD
FocusofAttentionDuration	UNDEFINED
FocusofAttentionFrequency	UNDEFINED
EyeContactDuration	NUMBER
EyeContactFrequencyExplainingTo	NUMBER
EyeContactFrequencyExplainingToOlder	NUMBER
EyeContactFrequencyExplainingToYounger	NUMBER
EyeContactFrequencyExplainingBy	NUMBER
EyeContactFrequencyExplainingByOlder	NUMBER
EyeContactFrequencyExplainingByYounger	NUMBER
NodExtentAttention	NUMBER
NodExtentGratitude	NUMBER
NodExtentAgreement	NUMBER
NodExtentRespect	NUMBER
NodExtentFriendliness	NUMBER
NodExtentComprehension	NUMBER
NodExtentInterest	NUMBER
NodVelocity	UNDEFINED
DeicticShape	WORD
DeicticAvoid	WORD
IconicShape	WORD
SymbolicShapeRespect	WORD
SymbolicShapeTurnTaking	WORD
SymbolicShapeGratitude	WORD
SymbolicShapeAgreement	WORD
SymbolicShapeFriendliness	WORD
SymbolicShapeConfusion	WORD
SymbolicShapeComprehension	WORD
SymbolicShapeInterest	WORD
SymbolicShapeHandover	WORD
SymbolicShapeHandShake	WORD
SymbolicAvoidWave	NUMBER
SymbolicAvoidHandover	WORD
BeatShape	UNDEFINED
BeatIntensity	UNDEFINED
BeatFrequency	UNDEFINED
BowExtentGreeting	NUMBER
BowExtentRespect	NUMBER
BowExtentGratitude	NUMBER
SwayIntensity	WORD
SwayFrequency	UNDEFINED

5.4 Culture Knowlege Base

The culture knowledge base file comprises a list of key-value pairs as shown in Figure 17. Key-value pairs without values in Tables 12 -14 have been omitted.

Table 17: Key-value pairs listed in the knowledge base file `cultureKnowledgeBaseInput.dat`.

PassingDistance	100
PassingPositionPreferred	BEHIND
PassingPositionAvoid	BETWEEN
AccompanyingDistanceLeading	100
AccompanyingPositionLeading	FRONTSIDE
AccompanyingPositionOlder	SIDE
AccompanyingPositionSameAge	SIDE
AccompanyingPositionYounger	SIDE
ExpressionLoudness	40
ExpressionPhraseRhythm	Nice
ExpressionPhraseGreeting	Muraho | Greetings
ExpressionPhraseTitle	Mr | Ms | Dr | Prof | Excellency
ExpressionPhrasePreamble	It is nice to meet you. How are you?
ExpressionPhraseSocial	I hope you are well
ExpressionPhraseAddressOlder	Muraho | Hello
ExpressionPhraseAddressSameAge	Muraho | Hello
ExpressionPhraseAddressYounger	Muraho | Hello
ExpressionPhraseLanguage	Kinyarwanda | isiZulu | English
ExpressionPhrasePassing	Muraho | Hello
PauseDuration	2
TurnTakingInitiationOrder	WAIT
TurnTakingAvoid	INTERRUPT
FocusofAttentionTargetRespect	TORSO
FocusofAttentionTargetAddressed	FACE
FocusofAttentionTargetExplanationTo	FACE_AND_OBJECT
FocusofAttentionTargetExplanationBy	FACE_AND_OBJECT
EyeContactDuration	2
EyeContactFrequencyExplainingTo	5
EyeContactFrequencyExplainingToOlder	5
EyeContactFrequencyExplainingToYounger	5
EyeContactFrequencyExplainingBy	5
EyeContactFrequencyExplainingByOlder	5
EyeContactFrequencyExplainingByYounger	5
NodExtentAttention	30
NodExtentGratitude	30
NodExtentAgreement	30
NodExtentRespect	30
NodExtentFriendliness	30
NodExtentComprehension	30
NodExtentInterest	30
DeicticShape	PALM_UPWARDS
DeicticAvoid	PALM_FORWARDS
IconicShape	ANIMATE_BEHAVIOR
SymbolicShapeRespect	BOTH_HANDS_UP
SymbolicShapeTurnTaking	RIGHT_HAND_UP
SymbolicShapeGratitude	CLASP_HANDS
SymbolicShapeAgreement	RIGHT_THUMB_UP
SymbolicShapeFriendliness	HANDSHAKE
SymbolicShapeConfusion	RIGHT_HAND_UP
SymbolicShapeComprehension	RIGHT_THUMB_UP
SymbolicShapeInterest	RIGHT_THUMB_UP
SymbolicShapeHandover	BOTH_HANDS
SymbolicShapeHandShake	POLITE_HANDSHAKE
SymbolicAvoidWave	30
SymbolicAvoidHandover	LEFT_HAND
BowExtentGreeting	10
BowExtentRespect	10
BowExtentGratitude	10
SwayIntensity	ANIMATE_BEHAVIOR

5.5 Output Data File

There is no output data file for the culture knowledge base helper class.

5.6 Class Definition

Instantiating the `CultureKnowledgeBase` class as a C++ object causes the contents of the culture knowledge base file to be read and stored in private dictionary data structure. Diagnostic messages are printed on the screen, depending on the value of `verboseMode` key in the configuration file. The contents of the dictionary are accessed using the keys. Appendix A provides the full definition of the `CultureKnowledgeBase` class.

5.6.1 Constructor

The `CultureKnowledgeBase()` constructor reads the configuration file to determine the mode of operation, the name of the knowledge base value types file, and the name of the knowledge base file. It sets a private data member flag with the mode of operation, initializes the private dictionary data structure with the key-value pairs read from the knowledge base file. If operating in verbose mode, it echoes the keys and values to the terminal.

5.6.2 Destructor

The `~CultureKnowledgeBase()` destructor deletes the dictionary data structure and write a diagnostic message if in verbose mode.

5.6.3 Private Data

The dictionary is implemented using a binary search tree with an element of type `struct KeyValueType`, with four fields.

```
typedef  struct {
    char *key;
    union {
        int   integerValue;
        char* alphanumericValue;
    };
    int valueType;
    bool initialized;
} KeyValueType;
```

The first field is the `key`. The data type is a C-string, i.e., a null-terminated array of characters.

The second field is the value associated with each key. The data type is dependent of the key: either an integer or a C-string, with the values specified in Tables 12 -14. This is accommodated by using a union data type, with two fields `integerValue` and `alphanumericValue`, an integer and a C-string respectively, the latter implemented as a null-terminated array of characters.

The is also an integer flag `valueType` to denote the type of the value so that it can be processed correctly, i.e., so that one knows which element of the union has been used for that key. It has one of four values — `NUMBER`, `WORD`, `PHRASE`, and `UNDEFINED` — as noted above.

Finally, there is a Boolean flag `initialized` that is set to `true` if the key has been initialized with a valid value. This flag is set to `false` if the value type of the key is `UNDEFINED`.

There are also a small number of other private utility data fields to store the configuration filename, the configuration data, a keyValue, and the verbose mode flag.

5.6.4 Public Access Methods

There are two public methods, one to print the knowledge base to the screen and one to retrieve a key-value pair.. These are `printToScreen()` and `getValue()`, respectively.

The `printToScreen()` method does not have any parameters.

This `getValue()` method has two parameters: a key and a value, as follows.

```
bool getValue(char *key, KeyValueType *keyValue);
```

The method returns `true` if the key value was successfully retrieved from the dictionary, `false` otherwise.

6 Example Application

The example application in `cultureKnowledgeBaseApplication.cpp` illustrates the use of the class to read the culture knowledge base file and print each key-value pair, along with its value type and initialization flag. It also provides four examples of how to retrieve the values of keys, each one with a different value type, and write the associated values to the terminal.

```
#include <utilities/cultureKnowledgeBase.h>

int main() {

   KeyValueType keyValue; // structure with key, value, value type, initialization flag
   Keyword      key;      // string
   int          i;        // counter

   /* instantiate the cultural knowledge base object                          */
   /* this reads the knowledge value types file and the knowledge base file  */
   /* as specified in the culturalKnowledgeBaseConfiguration.ini file        */

   CultureKnowledgeBase knowledgebase;

   /* verify that the knowledge base was read correctly */

   knowledgebase.print_to_screen();

   /* query the contents off the knowledge base */
   /* retrieve example of the four value types  */

   for (i=0; i<4; i++) {
      switch (i) {
         case UNDEFINED: strcpy(key,"ExpressionSpeed");
                         break;
         case NUMBER:    strcpy(key,"PassingDistance");
                         break;
         case WORD:      strcpy(key,"PassingPositionAvoid");
                         break;
         case PHRASE:    strcpy(key,"ExpressionPhrasePreamble");
                   break;
      }

      if (knowledgebase.getValue(key, &keyValue) == true) {

         switch (keyValue.valueType) {

            case UNDEFINED: printf("main: attempting to retrieve an UNDEFINED value type\n");
                            break;
            case NUMBER:    printf("main: %-40s %-40d %-20s %-20s\n", keyValue.key,
                                        keyValue.integerValue,
                                        valueType2Alphanumeric(keyValue.valueType),
                                        initialized2Alphanumeric(keyValue.initialized));
                            break;
            case WORD:      printf("main: %-40s %-40s %-20s %-20s\n", keyValue.key,
                                        keyValue.alphanumericValue,
                                        valueType2Alphanumeric(keyValue.valueType),
                                        initialized2Alphanumeric(keyValue.initialized));
                            break;
            case PHRASE:    printf("main: %-40s %-40s %-20s %-20s\n", keyValue.key,
                                        keyValue.alphanumericValue,
                                        valueType2Alphanumeric(keyValue.valueType),
                                        initialized2Alphanumeric(keyValue.initialized));
                            break;
            default:        printf("main: invalid value type %d\n",   keyValue.valueType);
                            break;
         }
      }
   }
}
```

Note the use of two utility functions, `valueType2Alphanumeric` and `initialized2Alphanumeric`, which convert the integer value type `valueType` and the Boolean `initialized` value to their equivalent alphanumeric strings for ease of interpretation when reporting them on screen.

Run the application by entering the following command:

```
rosrun utilities cultureKnowledgeBaseExample
```

This assumes the existence of a `utilities` package, as shown in Figures 5 and 6, and that the package has been installed, as described in Deliverable D3.3 Software Installation Manual.
A screenshot of the output of running this application is shown in Figure 7.

```
dvernon@ubuntu:~/workspace/ros$ rosrun utilities cultureKnowledgeBaseExample
AccompanyingDistanceLeading             100                                     NUMBER              Initilaized Value
AccompanyingPositionLeading             FRONTSIDE                               WORD                Initilaized Value
AccompanyingPositionOlder               SIDE                                    WORD                Initilaized Value
AccompanyingPositionSameAge             SIDE                                    WORD                Initilaized Value
AccompanyingPositionYounger             SIDE                                    WORD                Initilaized Value
ApproachingDistance                     (null)                                  UNDEFINED           Uninitialized Value
ApproachingOrientation                  (null)                                  UNDEFINED           Uninitialized Value
ApproachingVelocity                     (null)                                  UNDEFINED           Uninitialized Value
BeatFrequency                           (null)                                  UNDEFINED           Uninitialized Value
BeatIntensity                           (null)                                  UNDEFINED           Uninitialized Value
BeatShape                               (null)                                  UNDEFINED           Uninitialized Value
BowExtentGratitude                      10                                      NUMBER              Initilaized Value
BowExtentGreeting                       10                                      NUMBER              Initilaized Value
BowExtentRespect                        10                                      NUMBER              Initilaized Value
BowVelocity                             (null)                                  UNDEFINED           Uninitialized Value
DeicticAvoid                            PALM_FORWARDS                           WORD                Initilaized Value
DeicticShape                            PALM_UPWARDS                            WORD                Initilaized Value
ExpressionLoudness                      40                                      NUMBER              Initilaized Value
ExpressionPhraseAddressOlder            Muraho                                  WORD                Initilaized Value
ExpressionPhraseAddressSameAge          Muraho                                  WORD                Initilaized Value
ExpressionPhraseAddressYounger          Muraho                                  WORD                Initilaized Value
ExpressionPhraseGreeting                Muraho                                  WORD                Initilaized Value
ExpressionPhraseLanguage                Kinyarwanda                             WORD                Initilaized Value
ExpressionPhrasePassing                 Muraho                                  WORD                Initilaized Value
ExpressionPhrasePreamble                It is nice to meet you. How are you?    PHRASE              Initilaized Value
ExpressionPhraseRhythm                  Nice                                    WORD                Initilaized Value
ExpressionPhraseSocial                  I hope you are well                     PHRASE              Initilaized Value
ExpressionPhraseTitle                   Dr                                      WORD                Initilaized Value
ExpressionSpeed                         (null)                                  UNDEFINED           Uninitialized Value
EyeContactDuration                      2                                       NUMBER              Initilaized Value
EyeContactFrequencyExplainingBy         5                                       NUMBER              Initilaized Value
EyeContactFrequencyExplainingByOlder    5                                       NUMBER              Initilaized Value
EyeContactFrequencyExplainingByYounger  5                                       NUMBER              Initilaized Value
EyeContactFrequencyExplainingTo         5                                       NUMBER              Initilaized Value
EyeContactFrequencyExplainingToOlder    5                                       NUMBER              Initilaized Value
EyeContactFrequencyExplainingToYounger  5                                       NUMBER              Initilaized Value
FillerFrequency                         (null)                                  UNDEFINED           Uninitialized Value
FillerSound                             (null)                                  UNDEFINED           Uninitialized Value
FocusofAttentionDuration                (null)                                  UNDEFINED           Uninitialized Value
FocusofAttentionFrequency               (null)                                  UNDEFINED           Uninitialized Value
FocusofAttentionTargetAddressed         FACE                                    WORD                Initilaized Value
FocusofAttentionTargetExplanationBy     FACE_AND_OBJECT                         WORD                Initilaized Value
FocusofAttentionTargetExplanationTo     FACE_AND_OBJECT                         WORD                Initilaized Value
FocusofAttentionTargetRespect           TORSO                                   WORD                Initilaized Value
IconicShape                             ANIMATE_BEHAVIOR                        WORD                Initilaized Value
NodExtentAgreement                      30                                      NUMBER              Initilaized Value
NodExtentAttention                      30                                      NUMBER              Initilaized Value
NodExtentComprehension                  30                                      NUMBER              Initilaized Value
NodExtentFriendliness                   30                                      NUMBER              Initilaized Value
NodExtentGratitude                      30                                      NUMBER              Initilaized Value
NodExtentInterest                       30                                      NUMBER              Initilaized Value
NodExtentRespect                        30                                      NUMBER              Initilaized Value
NodVelocity                             (null)                                  UNDEFINED           Uninitialized Value
PassingDistance                         100                                     NUMBER              Initilaized Value
PassingPositionAvoid                    BETWEEN                                 WORD                Initilaized Value
PassingPositionPreferred                BEHIND                                  WORD                Initilaized Value
PassingVelocity                         (null)                                  UNDEFINED           Uninitialized Value
PauseDuration                           2                                       NUMBER              Initilaized Value
PauseFrequency                          (null)                                  UNDEFINED           Uninitialized Value
StandingDistance                        (null)                                  UNDEFINED           Uninitialized Value
StandingOrientation                     (null)                                  UNDEFINED           Uninitialized Value
SwayFrequency                           (null)                                  UNDEFINED           Uninitialized Value
SwayIntensity                           ANIMATE_BEHAVIOR                        WORD                Initilaized Value
SymbolicAvoidHandover                   LEFT_HAND                               WORD                Initilaized Value
SymbolicAvoidWave                       30                                      NUMBER              Initilaized Value
SymbolicShapeAgreement                  RIGHT_THUMB_UP                          WORD                Initilaized Value
SymbolicShapeComprehension              RIGHT_THUMB_UP                          WORD                Initilaized Value
SymbolicShapeConfusion                  RIGHT_HAND_UP                           WORD                Initilaized Value
SymbolicShapeFriendliness               HANDSHAKE                               WORD                Initilaized Value
SymbolicShapeGratitude                  CLASP_HANDS                             WORD                Initilaized Value
SymbolicShapeHandShake                  POLITE_HANDSHAKE                        WORD                Initilaized Value
SymbolicShapeHandover                   BOTH_HANDS                              WORD                Initilaized Value
SymbolicShapeInterest                   RIGHT_THUMB_UP                          WORD                Initilaized Value
SymbolicShapeRespect                    BOTH_HANDS_UP                           WORD                Initilaized Value
SymbolicShapeTurnTaking                 RIGHT_HAND_UP                           WORD                Initilaized Value
TurnTakingAvoid                         INTERRUPT                               WORD                Initilaized Value
TurnTakingDuration                      (null)                                  UNDEFINED           Uninitialized Value
TurnTakingInitiationOrder               WAIT                                    WORD                Initilaized Value
TurnTakingUtterance                     (null)                                  UNDEFINED           Uninitialized Value

main: attempting to retrieve a value for a key-value pair with an UNDEFINED value type
main: PassingDistance                   100                                     NUMBER              Initilaized Value
main: PassingPositionAvoid              BETWEEN                                 WORD                Initilaized Value
main: ExpressionPhrasePreamble          It is nice to meet you. How are you?    PHRASE              Initilaized Value
dvernon@ubuntu:~/workspace/ros$
```

Figure 7: Screenshot of the output of running the example application.

Appendix A The CultureKnowledgeBase Class

Note: documentation comments for the private methods have been removed due to space constraints but are retained in the source file.

```
#define NUMBER_OF_CONFIGURATION_KEYS   3
#define NUMBER_OF_VALUE_TYPES          4

/* constant definitions for valueType flag to identify which element of the union is to be used */

#define UNDEFINED 0 // value hasn't been initialized
#define NUMBER    1 // value is integer
#define WORD      2 // value is alphanumeric but just one word
#define PHRASE    3 // value is alphanumeric but several words

typedef char Keyword[KEY_LENGTH];

typedef struct {
   char knowledgeBase[MAX_FILENAME_LENGTH];
   char valueTypes[MAX_FILENAME_LENGTH];
   bool verboseMode;
} ConfigurationDataType;

typedef  struct {
    char *key;
    union {
        int   integerValue;
        char* alphanumericValue;
    };
    int valueType;
    bool initialized;
} KeyValueType;

typedef struct node *NodeType;

typedef struct node {
           KeyValueType keyValue;
           NodeType left, right;
        } Node;

typedef NodeType BinaryTreeType;

typedef BinaryTreeType WindowType;

class CultureKnowledgeBase {
public:
   CultureKnowledgeBase();
   ~CultureKnowledgeBase();
   bool                  getValue(char *key, KeyValueType *keyValue);
   void                  print_to_screen();

private:

   /* data members */
   KeyValueType          keyValue;
   BinaryTreeType        tree = NULL;
   ConfigurationDataType configurationData;
   char                  configuration_filename[MAX_STRING_LENGTH] = "cultureKnowledgeBaseConfiguration.ini";

   /* methods */
   void                  assign_key_attributes(KeyValueType *keyValue, char key[],  int valueType, bool operational);
   void                  assign_key_value(KeyValueType *keyValue, int integerValue, bool operational);
   void                  assign_key_value(KeyValueType *keyValue, char *alphanumericValue, bool operational);
   KeyValueType          delete_min(BinaryTreeType *tree);
   BinaryTreeType        *delete_element(KeyValueType keyValue, BinaryTreeType *tree);
   bool                  exists(char *key);
   bool                  exists(KeyValueType keyValue, BinaryTreeType *tree);
   int                   height();
   int                   height(BinaryTreeType tree, int n);
   int                   getValueType(char *key);
   bool                  getValue(char *key, KeyValueType *keyValue, BinaryTreeType *tree);
   void                  initialize(BinaryTreeType *tree);
   BinaryTreeType        *insert(KeyValueType keyValue, BinaryTreeType *tree, bool update);
   int                   inorder_print_to_screen(BinaryTreeType tree, int n);
   int                   inorder_print_to_file(BinaryTreeType tree, int n, FILE *fp_out);
   int                   postorder_delete_nodes(BinaryTreeType tree);
   int                   print_to_file(FILE *fp_out);
   int                   print_to_file(BinaryTreeType tree, FILE *fp_out);
   int                   print_to_screen(BinaryTreeType tree);
   int                   size(BinaryTreeType tree);
   void                  readConfigurationData();
   void                  readKnowledgeBase();
   void                  readKnowledgeBaseValueTypes();
   int                   size();
   int                   total_number_of_probes();
   int                   total_number_of_probes(BinaryTreeType tree, int n);
};
```

References

[1] B. Bruno, C. T. Recchiuto, I. Papadopoulos, A. Saffiotti, C. Koulouglioti, R. Menicatti, F. Mastrogiovanni, R. Zaccaria, and A. Sgorbissa. Knowledge representation for culturally competent personal robots: requirements, design principles, implementation, and assessment. *International Journal of Social Robotics*, 11(3):515–538, 2019.

Principal Contributors

The main authors of this deliverable are as follows (in alphabetical order).

Eyerusalem Birhan, Carnegie Mellon University Africa.
Muhirwa Richard, Carnegie Mellon University Africa.
David Vernon, Carnegie Mellon University Africa.

Document History

Version 1.0

First version created by moving and reorganizing Section 2 Representation of Cultural Knowledge and Section 3.2 Action and Cultural Parameter Values fom Deliverable D1.2.
David Vernon.
30 December 2024.

Version 1.1

Changed ontology category from Words to Expression.
Introduced an extended ontology tree in which there is at least one leaf node for each consensus answer; see Fig. 3 .
Added Tables 6 - 14.
Added Section 5 Implementation.
David Vernon.
25 February 2025.

Version 1.2

Added launch subdirectory and launch file to the directory structure in Figure 6.
David Vernon.
28 February 2025.

Version 1.3

Changed `cultureKnowledgeBase.dat` and `cultureKnowledgeBaseValueTypes.dat` to `cultureKnowledgeBaseInput.dat` and `cultureKnowledgeBaseInput.dat` to be compliant with the software engineering standards.
David Vernon.
3 March 2025.